SKIN CANCER DETECTION APP USING DEEP LEARNING

AI19511 – MOBILE APPLICATION DEVELOPMENT LABORATORY FOR ML AND DL APPLICATIONS

A PROJECT REPORT

Submitted by

JAYASHRINIDHI V (221501053)

PRAJEIN C K (221501189)

in partial fulfillment for the award of the

degree of

BACHELOR OF TECHNOLOGY

in

ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

RAJALAKSHMI ENGINEERING COLLEGE

PROGRESS THROUGH KNOWLEDGE

RAJALAKSHMI ENGINEERING COLLEGE, KANCHIPURAM ANNA UNIVERSITY : CHENNAI 600 025

DECEMBER 2024

RAJALAKSHMI
ENGINEERING COLLEGE

BONAFIDE CERTIFICATE

NAME ………………………………………………………………………………….….…………

ACADEMIC YEAR……………….………SEMESTER…………BRANCH………………

UNIVERSITY REGISTER No.

Certified that this is the bonafide record of work done by the above students in the Mini Project titled **SKIN CANCER DETECTION APP USING DEEP LEARNING** in the subject **AI19511 – MOBILE APPLICATION DEVELOPMENT LABORATORY FOR ML AND DL APPLICATIONS** during the year **2024 - 2025.**

Signature of Faculty – in – Charge

Submitted for the Practical Examination held on __________________

INTERNAL EXAMINER **EXTERNAL EXAMINER**

ABSTRACT

Cancer remains one of the leading causes of mortality worldwide, making early detection critical for effective treatment and improved survival rates. This project presents a deep learning-driven mobile application designed to aid in the early diagnosis of cancer, combining Convolutional Neural Networks (CNNs) for image-based classification with React Native Expo for seamless cross-platform development. The application processes medical images, identifying potential cancerous regions with high precision and speed. By leveraging state-of-the-art machine learning algorithms, the system offers robust performance in detecting abnormalities across various cancer types.

The React Native framework ensures compatibility across Android and iOS platforms, providing a user-friendly and accessible interface for both healthcare professionals and patients. The app also features a secure backend for managing sensitive user data, ensuring compliance with privacy regulations. Performance evaluation based on metrics like accuracy, sensitivity, specificity, and F1-score validates the reliability of the system. This work demonstrates the transformative potential of artificial intelligence in healthcare, aiming to bridge the gap between cutting-edge diagnostic technology and its accessibility in resource-limited settings. The project is a step toward democratizing healthcare, empowering individuals with a portable, affordable, and efficient diagnostic solution.

ACKNOWLEDGEMENT

Initially I thank the Almighty for being with us through every walk of my life and showering his blessings through the endeavor to put forth this report.

My sincere thanks to our Chairman **Mr. S. MEGANATHAN, M.E., F.I.E.,** and our Chairperson **Dr. (Mrs.)THANGAM MEGANATHAN, M.E.,Ph.D.,** for providing me with the requisite infrastructure and sincere endeavoring educating me in their premier institution.

My sincere thanks to **Dr. S.N. MURUGESAN, M.E., Ph.D.,** our beloved Principal for his kind support and facilities provided to complete our work in time.

I express my sincere thanks to **Dr. K.SEKAR, M.E., Ph.D.,** Head of the Department of Artificial Intelligence and Machine Learning and Artificial Intelligence and Data Science for his guidance and encouragement throughout the project work. I convey my sincere and deep gratitude to our internal guide, **Dr. PAVITHRA GURU (Ph.D).,** Associate Professor, Department of Artificial Intelligence and Machine Learning, Rajalakshmi Engineering College for his valuable guidance throughout the course of the project.

Finally I express my gratitude to my parents and classmates for their moral support and valuable suggestions during the course of the project.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1
INTRODUCTION

Globally, cancer is a major health issue, cutting across different ethnic groups and areas with millions of people diagnosed with it every year. The World Health Organization estimates that cancer will be the leading cause of death globally sooner than cardiovascular diseases. Early diagnosis is key to improving treatment outcomes and sustaining life in cases of severe types of cancer such as melanoma, breast cancer, and lung cancer. The data has over and over shown that if cancer is detected at an early stage, the chances of successful treatment and long-term survival are high. Technology has brought about mobile apps that are now very important tools in enabling early cancer detection and in encouraging patient participation. Among these technologies, deep learning (DL) is the one which is most remarkable due to its high-performance capability in analyzing complicated medical images with high accuracy.

The planned mobile app will use deep learning algorithms to recognize skin cancer through image analysis. By leveraging convolutional neural networks (CNNs) which are basically used in image classification tasks, the app will analyze the skin lesions to figure out whether they are benign or malignant. The users can take photographs of their skin lesions with their smartphones, and the app will give them instant feedback as well as give suggestions for further action such as visiting a healthcare professional. This tactic aside from helping people to stay in control of their own health also makes the work of healthcare workers easier as they only take it from there by doing an initial assessment and thus excess people from clinics come in due to minor issues.

This mobile app's potential to democratize access to cancer screening tools makes it a very important link in the healthcare system. For instance, a lot of people have no alternative healthcare services due to the area they live in, financial constraints, or ignorance about cancer symptoms. The user-friendly interface of the mobile phone that lets a person test cancer danger from skin with the data collected by the smartphone will help these problems of access and awareness.

Cancer is one of the major public health issues worldwide, with millions being diagnosed every year in different age groups and geographical regions. According to the World Health Organization cancer is expected to overtake cardiovascular diseases as the leading cause of death globally.

Numerous pieces of research have indicated that early detection of cancer leads to a higher probability of successful treatment and longer survival periods. Through technological progress, mobile applications have become highly effective instruments in identifying cancer at an early stage and engaging the patients in the process. Among these technologies, deep learning (DL) is particularly remarkable due to its ability to examine complex medical data, especially images, with precision.

The primary characteristics of the application are deep learning integration that uses the latest CNN architectures (ResNet50v2 and Inception v3 as examples) trained on large-scale dermoscopic image datasets to make sure that the classification is correct.

These algorithms can detect image patterns that are invisible to the human eye, thus, improving the accuracy of the diagnosis. On one hand, the application uses on-device processing capabilities which perform the analysis in real-time without the necessity for internet connection thus, the user data is constantly private and accessible. Besides, it contains tools for user engagement like symptom tracking, educational content about skin cancer prevention and treatment options, as well as regular check-up reminders. During data training, data augmentation is also carried out to eliminate data imbalance problems thus, model robustness is increased by artificially enlarging the training dataset with variations of existing images.

The application has not only the technical aspects of it but it also educates and raises awareness, which is seen as fundamental for cancer prevention. A variety of information concerning skin cancer risk factors, early warning cues, and guidelines for the conduct of self-examinations will be provided to the users. A well-informed user base would be the means by which the app would not only help in early detection but also promote proactive health management behaviors among users.

This project comes as an innovative breakthrough in mobile technology and deep learning in cancer diagnosis by means of deep learning, thus, a mobile application will be the main product. Using this tool of diagnosis, which is available on your smartphone, you can get the help of professionals and increase the chance of healing while others get educations on skin cancer threats. Applications of this sort will only be the harbinger of the new age of how people will get their information on their health and respond to that information in the years to come. Here mobile becomes one of the valuable weapons, which would enhance massively the breadth of actions on behalf of users leading them to detect cancer in its early stages and subsequently take informed treatment decisions.

The cancer detection mobile app is incorporated with sophisticated artificial intelligence (AI) and machine learning (ML) algorithms that self-learn from data fed by the users on an ongoing basis. Gathering data begins with acquiring suitable information through user interactions, image uploads, and other sources. As the users interact with the app—for example, by submitting images of the skin lesions—the data is collected and pruned to make it appropriate for the model's training. This step consists of cleaning and transforming the data into a polished format ready for the analysis and ensuring that the data is coherent and accurate.

When the data has been gathered, the app utilizes machine learning to teach the algorithms. The basic process of this model is based on using labeled data where skin cancer images are classified into benign or malignant categories. Artificial examples permit the model to grasp the differences between the categories. At the stage of training, the deep learning model, specifically the convolutional neural network (CNN), which interestingly, you might be familiar with among other things, has a prowess of detecting and classifying images, is provided with the data. The input data process includes the adjustment of model's internal parameters, gradually expanding its ability to describe new images.

As new data become accessible—be the new data user uploads or additional datasets—the app can undertake the process of continuous learning. This is to say, the model can be updated with new examples to better its predictions. For instance, if a user shares a skin image with a doctor that then is verified as malignant, this data can be used to increase the model's accuracy in detecting similar cases in the future. Furthermore, the continuous process of learning makes it possible for the app not only to adapt to the new pattern but also to become better at diagnosing as time goes by.

In a broad sense, the combination of artificial intelligence and machine learning in mobile cancer detecting applications puts forward an interactive system that can evolve through the new information. The app will be consistently receiving input from users and their feedback and according become more and more precise in the evaluation of skin lesions. The latter function could effectively be the way of the patients' engagement in the whole healing process and also the discovery of possible cancer at an early stage. On the other hand, it is also an imperative in a field such as oncology, where the correct diagnosis in time could determine the course of treatment and the survival prospects.

CHAPTER 2

LITERATURE SURVEY

Different types of cancer detection and classification using machine assistance have opened up a new research area for early cancer detection, which has shown the ability to reduce manual system impairments. This survey presents several sections on state of art techniques, analysis and comparisons on benchmark datasets for the brain tumor, breast cancer, lung cancer, liver tumor, leukemia and skin lesion detection respectively from F-measure, sensitivity, specificity, accuracy, precision points of view.

Nie et al. reported a complete automatic 3D-CNN brain tumor segmentation systems utilizing both T1, MR test (MRI) and Diffusion Tensor Imaging (DTI). Different pre-processing strategies were measured using T1, DTI tensor and fMRI-specific BLD (blood oxygen-dependent) variance rate for increasing MR modality i.e. strength standardization. The extractor function was 3D-CNN and the final prediction was carried out by SVM with 89.9% (accuracy), 92.19% (sensitivity), 88.22% (specificity). However, the proposed system is computationally expensive and is unsuitable for large dataset.

Arikan et al. proposed a semi-automated, collaborative seed selection-based SVM approach for the fragmentation of brain tumors. They used anisotropic diffusion filter in the MR images during the pre-processing stage to eliminate noise. Random seeds would then be chosen for the SVM classification from the previously processed MR images. For success evaluation, a freely accessible BRATS 2015 dataset was used. To test the outcomes of the new procedure they picked four patients from the MICCIA BRATS-2015 data collection. Their solution obtained an average Dice Similarity (DS) of approximately 81% relative to the basic reality. Ellwaa et al. proposed a fully automatic segmentation approach for MRI-based brain tumors utilizing the iterative random tree. The accuracy of the patient with specific details in the data set used for random forest classification was enhanced iteratively. The technique tested on BRATS-2016. Selection requirements for the individual with the strongest knowledge lead to positive outcomes. However, no justification was provided about the selection requirements and no accuracy results reported. Abbasi and Tajeripour [71] proposed a 3D automated interface for brain tumor detection and segmentation on BRATS 2013 dataset. For pre-processing, bias field correction, histogram communication, ROIs (Region of Interests) and are isolated from the FLAIR image context. For learning tasks, HOG and LBP features were fed to random forest classifier. However, they used synthetic data for experiments and reported an accuracy of 93%.

Mehmood et al. proposed an effective system for brain imaging and simulation using MR images. In the beginning, there is an immersive, semi-automated 3D segmentation technique that essentially separates the brain and tumor regions from the MR parts using SVM 95.53% accuracy, 99.49% sensitivity, 99.0% precision and 0.09 mean square error. However, experiments performed on self-generated datasets. Das et al. detected normal and abnormal tissue samples by using texture-based features. 80 images of abnormal and normal tissue were used obtained from GMCH, Guwahati Hospitals. Five descriptors fusion (LBP, Tamura, HOG, GRLN, and GLCM) extracted features from the images and formulated 172 features vector set. Additionally, six classifiers i.e. SVM, K-NN, Logistic Regression, Quadratic Discriminant, Linear Discriminant used to evaluate the performance of each feature set both in the group and individually.

The experimental results demonstrated 100% accuracy using the complete feature set as compared to individual sets of features and it enhanced the average classification accuracy to 98.6%. They concluded that the computer-aided system could lead to better diagnosis as child-hood brain lesions are very critical. The use of several classifiers makes the system slow and computationally expensive.

Iqbal et al. proposed a deep learning model for reliable brain tumor delineation from Medical Benchmarks, utilizing short-term memory (LSTM) and coevolutionary neural networks (ConvNet). The two different models ConvNet and LSTM are equipped using the same data set and create a group to maximize the performance. The two separate models are merged.

A data collection consisting of MRI images in four modalities T1, T2, T1c and FLAIR is widely accessible for this reason. MICCAI BRATS 2015 is accessible. To boost image clarity, several variations are designed and optimal output variations are added to the preprocessing approaches including noise reduction, histogram equalization and edge enhancement. Class weighting is used in the current models to address the issue of class inequality. Validation details from the same image collection are checked on the learned model, and findings from each experiment are recorded. ConvNet as a single score (exactitude) of 75% while 80% is produced by an LSTM-based network, with a total fusion of 82.29%.

The suggested Grab cut approach is implemented to specifically segment real losses symptoms while the Visual Geometry Transfer Learning System (VGG-19) is completed to produce features that are then assembled (shape and texture) by series. Such technologies are designed by entropy to reliably and easily identify and to transmit fused vectors to classification units.

The described model is tested in the MICCAI challenge databases including multi-modal brain tumor segmentation (BRATS) 2015, 2016 and 2017 respectively, utilizing top scientific image processing and computer-assisted intervention. The study findings with a coefficient of dice similarity (DSC) hit 0.99 using 2015 BRATS, 1.00 on 2016 BRATS and 0.99 using 2017 BRATS. However, they did not use other classifiers or their fusion to verify the viability of their technique.

Most recently, Ramzan et al. segmented multiple brain regions using 3DCNN with residual learning and extensive convolution to effectively apprise end-to- end mapping between MRI volumes & voxel-level brain segments. Mean 0.879 and 0.914 dice scores were obtained for 3 and 9 brain regions using data from 3 different sources. In comparison, for 8 brain regions with MRBrains18 data collection, a mean dice score of 0.903 is higher than 0.876 obtained in the last study. Similiary, to detect brain abnormalities from MR Images analysis, Nayak et al. proposed CNN model composed of five layers (four Convolutional layers and one fully-connected layer) with learnable parameters. To check the viability of their model two benchmark multi-class brain MRI datasets namely, MD-1 and MD-2 employed. Authors claimed 100% and 97.50% classification accuracy on MD-1 and MD-2 datasets respectively.

Lu et al. proposed a deep learning CNN ResNet based on the pyramid dilated convolution for Gliomas classification. Experiments were performed on a local clinical dataset and 80.11% glioma classification accuracy attained. However, they did not use benchmark dataset for experiments. presents a summary of existing methods for brain tumor detection, methodologies adopted, dataset and results

Smartphone-based Application for Skin Cancer Classification

This study presents a smartphone application designed to assist clinicians with low dermatological experience in skin cancer detection. The app utilizes a Convolutional Neural Network (CNN) trained on clinical images and patient demographics collected via smartphones. The authors propose a data balancing approach using Differential Evolution (DE) to handle imbalanced datasets, achieving a balanced accuracy of 85% and a recall of 96%. The application is developed using React Native and Expo, allowing it to send images to a server for analysis, which then returns diagnostic predictions.

Cancer Detection App Using React Native and TensorFlow

This project involves a cancer detection app that employs React Native for the frontend and TensorFlow for the backend. It allows users to capture photos of moles and receive real-time analysis regarding their benign or malignant nature. The app's architecture supports efficient image processing through TensorFlow, showcasing the integration of deep learning within mobile applications built React Native.

CHAPTER 3
PROPOSED METHOD

3.1 System Overview:

The primary objective of this project was to design and implement a mobile application for cancer detection, leveraging the power of deep learning and cross-platform mobile development. The app integrates a Convolutional Neural Network (CNN) model to classify medical images and identify potential cancerous regions with high accuracy. The system provides an accessible diagnostic tool for patients and healthcare providers, focusing on improving early detection and intervention. React Native Expo is employed for frontend development, ensuring a seamless user experience across Android and iOS platforms, while the backend is powered by Python, utilizing FastAPI to enable efficient communication between the app and the machine learning model. The methodology encompasses various phases, including dataset preprocessing, model training, backend development, and seamless integration with the mobile application. The system is designed to handle sensitive medical data securely, delivering accurate diagnostic results through an intuitive interface. This project bridges the gap between cutting-edge medical image analysis and accessible healthcare solutions, emphasizing usability, performance, and data security.

3. 2 Data Collection and Dataset Preparation:

The success of any deep learning-based application hinges on the quality and diversity of the dataset. For this project, the dataset used comprises publicly available medical imaging data from reputable sources such as The Cancer Imaging Archive (TCIA), Kaggle datasets, and other medical research databases. The dataset includes various imaging modalities, such as histopathological slides, X-rays, or CT scans, depending on the specific type of cancer being targeted. These images are labeled by domain experts, categorizing them into cancerous and non-cancerous classes, as well as specifying cancer subtypes where applicable.

Before feeding the data into the Convolutional Neural Network (CNN), an extensive preprocessing pipeline was applied to enhance the quality, consistency, and utility of the images for training. This pipeline includes:

- **Image Resizing**: All images were resized to a uniform dimension suitable for the CNN architecture. This standardization ensures computational efficiency and consistency during training.

- **Normalization**: Pixel intensity values were normalized to a range of [0, 1] to reduce variations caused by lighting conditions or scanning devices, ensuring better convergence during training.
- **Data Augmentation**: To address the challenges of imbalanced datasets and improve model generalizability, data augmentation techniques were applied. These include random rotations, flips, zooming, cropping, and brightness adjustments, which artificially expand the dataset and help the model learn robust features.
- **Noise Removal and Smoothing**: Filters were applied to remove noise and enhance important features in the images. Techniques such as Gaussian blur were used to smooth the data, aiding in the extraction of meaningful patterns.
- **Segmentation (if applicable)**: For certain cancers like skin cancer, segmentation was performed to isolate the region of interest (e.g., lesion areas) from the background. This step helps the model focus on relevant features and reduces the influence of irrelevant data.
- **Class Balancing**: If the dataset was imbalanced (i.e., there were significantly more samples in one class than the other), techniques such as Synthetic Minority Over-sampling Technique (SMOTE) or undersampling were used to balance the data.
- **Splitting the Dataset**: The dataset was divided into training, validation, and test sets in an 80-10-10 ratio. This ensures the model is trained on a diverse subset, validated on unseen data for hyperparameter tuning, and tested on a completely separate set to evaluate performance metrics.

The preprocessing steps ensure that the dataset is not only of high quality but also diverse and balanced, allowing the CNN to learn effectively and generalize well to unseen data. By addressing common challenges such as class imbalance and image variability, the preprocessing pipeline significantly contributes to the robustness and reliability of the cancer detection model.

3.3 Model Development:

The core of this project is the deep learning model, which employs a Convolutional Neural Network (CNN) to detect cancer from medical images. CNNs are particularly well-suited for image analysis tasks due to their ability to learn spatial hierarchies of features through layered architectures. The model development process involved careful consideration of the architecture, hyperparameters, and optimization strategies to ensure high accuracy, robustness, and computational efficiency.

Architecture Design:

The CNN architecture was tailored to handle medical images, emphasizing both depth and efficiency. It consisted of several layers, including:

- **Convolutional Layers**: Extract low- and high-level features, such as edges, textures, and complex patterns. Filters were fine-tuned for feature extraction relevant to cancerous tissue.
- **Pooling Layers**: Downsample feature maps to reduce dimensionality and retain critical information while mitigating overfitting. Max pooling was employed to preserve dominant features.
- **Batch Normalization**: Applied after convolutional layers to stabilize and accelerate training by normalizing activations.
- **Dropout Layers**: Incorporated between fully connected layers to reduce overfitting by randomly deactivating a fraction of neurons during training.
- **Fully Connected Layers**: Used for combining extracted features and producing the final classification probabilities.

To leverage the power of pre-trained models, architectures like VGG16, ResNet, and InceptionV3 were evaluated. Transfer learning allowed the model to build on learned features from large datasets, significantly improving performance with limited medical imaging data. The final layers were retrained with the project's specific dataset to fine-tune for cancer detection.

Loss Function and Optimization:

The **binary cross-entropy loss** function was used for binary classification tasks (e.g., cancerous vs. non-cancerous), while **categorical cross-entropy** was used for multi-class problems (e.g., cancer subtypes). The **Adam optimizer** was employed due to its efficiency and adaptability in handling sparse gradients. Learning rate scheduling was implemented to adjust the learning rate dynamically during training for faster convergence.

Regularization Techniques:

To address overfitting, L2 regularization and dropout layers were incorporated into the model. Data augmentation further enhanced robustness by exposing the model to a diverse range of input variations.

Model Training:

The model was trained using mini-batch gradient descent with a batch size of 32. The training process included extensive hyperparameter tuning, adjusting parameters such as the number of layers, filter sizes, learning rates, and dropout rates. Early stopping was employed to terminate training once validation performance plateaued, preventing overfitting.

Evaluation Metrics:

The model's performance was assessed using metrics like accuracy, precision, recall, F1-score, and Area Under the Receiver Operating Characteristic Curve (AUC-ROC). These metrics provide a comprehensive view of the model's ability to correctly classify cancerous and non-cancerous cases, minimizing false positives and negatives.

Explainability:

Techniques such as Gradient-weighted Class Activation Mapping (Grad-CAM) were incorporated to provide visual explanations for the model's predictions. This enhances trust in the model by allowing healthcare providers to interpret which regions of the image influenced the classification.

By combining a robust architecture with state-of-the-art optimization techniques, the CNN was able to achieve high accuracy in cancer detection. The model was carefully designed to balance computational efficiency with diagnostic precision, ensuring its suitability for real-world applications on mobile devices.

.

3.4 Model Integration into the App:

Integrating the trained Convolutional Neural Network (CNN) model into the mobile application involved seamless communication between the app's frontend, developed using React Native Expo, and the backend, built with Python using FastAPI. This integration ensured efficient model inference, secure data handling, and a responsive user experience.

The trained CNN model was exported as a serialized format, such as TensorFlow SavedModel or PyTorch's TorchScript, allowing compatibility with the FastAPI backend. The backend API endpoints were designed to handle requests from the app, process uploaded images, and return classification results. The integration workflow consisted of the following steps:

- **Model Deployment in Backend**:
  The CNN model was hosted on a secure server within the FastAPI framework. FastAPI was chosen for its high performance and asynchronous capabilities, enabling it to handle multiple requests simultaneously. Endpoints were created for receiving medical images, preprocessing them to match the model's input requirements, and invoking the model for inference.
- **API Design and Communication**:
  RESTful APIs were developed to facilitate communication between the app and the backend. These endpoints allowed the app to send images for analysis, retrieve prediction results, and provide feedback to users. For instance:
  - **Image Upload Endpoint**: Accepts the uploaded medical image in a secure format (e.g., Base64 or multipart form data).
  - **Inference Endpoint**: Processes the image through the CNN model and returns the prediction in real-time, including probabilities and labels (e.g., "Cancerous" or "Non-Cancerous").
- **Frontend Integration**:
  In the React Native Expo app, the frontend was connected to the backend using HTTP requests via libraries like Axios or Fetch API. Key features implemented in the frontend include:
  - **Image Capture and Upload**: Users can upload medical images directly from their gallery or camera. The app validates image formats and ensures secure transmission to the backend using HTTPS.
  - **Result Display**: After receiving the inference results from the backend, the app visually displays the classification outcome along with a confidence score. Grad-CAM visualizations are also

shown to highlight areas of the image influencing the prediction.

- **Authentication and Data Security**:
  To ensure the security of sensitive medical data, the app integrated secure authentication mechanisms using JSON Web Tokens (JWT). Data transmission between the app and backend was encrypted using HTTPS, safeguarding user privacy and compliance with regulations like GDPR and HIPAA.
- **Performance Optimization**:
  The backend was optimized to handle model inference with low latency, ensuring the app delivers results in real-time. Techniques such as model quantization were considered to reduce computational overhead without compromising accuracy.

By integrating the CNN model with the React Native Expo frontend and FastAPI backend, the system provides users with a seamless, efficient, and secure tool for cancer detection. This integration bridges advanced deep learning technology with intuitive mobile app functionality, making early diagnosis accessible to a wider audience.

Building the Backend:

The backend of the cancer detection application was built using Python and the FastAPI framework, chosen for its high performance, scalability, and ease of integration with machine learning models. The backend serves as the central hub for processing user requests, managing the CNN model, and securely delivering predictions. The FastAPI framework was employed to create RESTful API endpoints that facilitate seamless communication between the frontend and the server. The CNN model, trained and serialized using frameworks like TensorFlow or PyTorch, was deployed within the backend to handle inference requests. Robust preprocessing pipelines were implemented to standardize images received from the app, ensuring compatibility with the model's input requirements. The backend also incorporates secure mechanisms for handling sensitive data, including HTTPS for encrypted communication and authentication protocols such as JWT for secure user sessions. Additionally, asynchronous processing capabilities were leveraged to handle multiple concurrent requests efficiently, ensuring low-latency performance. This backend infrastructure not only supports accurate and fast predictions but also provides a secure and reliable foundation for the overall system.

App Integration:

The mobile application for cancer detection was developed using React Native with Expo, enabling cross-platform compatibility for both Android and iOS devices. React Native was chosen for its efficiency in building high-performance mobile apps with a single codebase, while Expo streamlined the development process by providing built-in tools for faster deployment and testing. The integration of the app with the backend, powered by FastAPI, was accomplished through the use of HTTP requests, allowing seamless communication between the frontend and the server for real-time cancer detection.

Key features of the frontend include a user-friendly interface for image upload, intuitive navigation, and clear visualization of diagnostic results. The app allows users to easily upload medical images via their device's camera or photo gallery, which are then sent to the backend for processing. Once the image is processed and analyzed by the trained CNN model, the app displays the prediction results—such as whether the image is cancerous or non-cancerous—along with a confidence score.

To ensure a smooth user experience, the frontend also includes loading indicators to inform users while the model is analyzing the image, as well as error handling to address invalid image uploads or network issues. For enhanced interpretability, Grad-CAM visualizations are incorporated to highlight regions of the image that contributed to the model's decision, providing users with valuable insights into the diagnostic process. Additionally, secure data transmission is ensured through HTTPS, and user sessions are managed using secure authentication protocols like JWT, maintaining the privacy and confidentiality of medical information.

The integration of React Native with FastAPI and the CNN model ensures a powerful, scalable, and secure mobile solution for cancer detection, making advanced AI-powered diagnostics accessible to users anywhere, at any time.

3. 5 User Interface Design:

The user interface (UI) of the cancer detection mobile application was designed with the primary goal of creating an intuitive, accessible, and seamless experience for both patients and healthcare professionals. With the focus on simplicity and ease of use, the app features a clean and minimalistic design that guides users through the image upload process and provides clear results in an easy-to-understand format. The UI was developed using React Native and Expo, ensuring consistency across both Android and iOS platforms.

The main screens in the app are carefully structured to prioritize the user's needs, starting with a simple and welcoming **home screen** that offers easy navigation. Key components of the UI include:

- **Image Upload Interface**:

The app allows users to upload medical images either by capturing them directly via the device's camera or selecting from the gallery. A clearly labeled button for both options ensures accessibility for users with varying levels of technical experience. The image upload screen features a preview of the selected image, providing users with immediate feedback and confidence that the correct file has been chosen.

- **Processing Screen**:

After the user uploads an image, the app transitions to a processing screen where a loading spinner or progress bar is displayed, indicating that the image is being analyzed by the backend. This feature ensures that users are informed during the model's inference process, reducing uncertainty and enhancing the overall experience.

- **Results Display**:

Once the image has been processed, the results screen provides a clear, concise outcome. The diagnosis—whether cancerous or non-cancerous—is prominently displayed, accompanied by a confidence score to indicate the model's certainty in its prediction. This helps users understand the reliability of the result. The screen also provides additional details such as potential next steps, offering guidance for further medical consultation.

- **Grad-CAM Visualization**:

To enhance the interpretability of the AI decision-making process, the results page includes a **Grad-CAM** visualization. This heatmap overlays on the original image, highlighting the areas that influenced the model's classification, giving users transparency and insight into how the model arrived at its decision. This feature is particularly valuable for healthcare professionals who may require more information to interpret the results.

- **Error Handling and Notifications**:

The app includes clear error messages and validation checks, such as ensuring the correct file type is uploaded or alerting the user if the image is of insufficient quality. These notifications help guide the user back on track, ensuring that only valid data is processed by the model. Additionally, the app sends helpful reminders or instructions if something goes wrong during image upload or inference, maintaining a positive user experience even when issues arise.

- **Settings and User Authentication**:

The app includes a simple settings page for user management, including features for account creation, login, and session management using secure authentication protocols like JWT. Users can also track their history of previously uploaded images and results.

The design was focused on making the app as user-friendly as possible, minimizing complexity while maintaining powerful diagnostic capabilities. This approach ensures that even users with minimal technical knowledge can easily navigate the app, upload images, and interpret results, while also providing more advanced features like Grad-CAM visualizations for healthcare professionals who may need additional context. The result is an intuitive, efficient, and reliable mobile application for cancer detection that is accessible to a wide range of users.

Model Evaluation and Performance Metrics

The evaluation of the cancer detection model is a critical step to ensure that it performs accurately and reliably in real-world applications. The model's effectiveness was assessed using various performance metrics that provide a comprehensive view of its predictive capabilities, balancing both overall accuracy and its ability to correctly identify both cancerous and non-cancerous cases. Key evaluation methods and performance metrics used in this project include:

- **Accuracy**:

Accuracy is one of the most common metrics used to assess the overall performance of the model. It is calculated as the proportion of correctly classified instances (both cancerous and non-cancerous) out of the total number of instances in the dataset. While accuracy provides a general sense of performance, it may not be sufficient in imbalanced datasets, where one class (e.g., non-cancerous) may dominate.

- **Specificity (True Negative Rate)**:

Specificity measures the ability of the model to correctly identify non-cancerous images, or how well the model avoids false positives. It is the ratio of correctly predicted non-cancerous images (True Negatives) to all actual non-cancerous images (True Negatives + False Positives).

- **Area Under the ROC Curve (AUC-ROC)**:

The AUC-ROC is a graphical representation of the model's ability to discriminate between cancerous and non-cancerous images across different thresholds. The Receiver Operating Characteristic (ROC) curve plots the True Positive Rate (Recall) against the False Positive Rate (1 - Specificity). The AUC score ranges from 0 to 1, with higher values indicating better model performance. An AUC close to 1 indicates that the model has a high true positive rate while minimizing false positives.

- **Confusion Matrix**:

A confusion matrix was used to visualize the performance of the model across all classes. It provides insight into how many cancerous and non-cancerous images were correctly or incorrectly classified, which is essential for understanding where the model may need improvement.

- **Cross-Validation**:

Cross-validation was employed to assess the model's robustness by splitting the dataset into multiple subsets (folds) and training the model on each subset while evaluating it on the remaining data. This approach helps to ensure that the model is not overfitting to a specific subset of the data and provides a more generalized estimate of its performance.

The performance metrics of the CNN model were evaluated on the test dataset, and the results demonstrated that the model performed exceptionally well across various metrics. A high **accuracy**, **precision**, **recall**, and **F1-score** indicated that the model can effectively identify both cancerous and non-cancerous images. The **AUC-ROC** score showed that the model has excellent discriminatory power, making it suitable for real-time diagnostic applications. Furthermore, the confusion matrix helped identify areas for improvement, such as reducing false positives and negatives, which is crucial for minimizing unnecessary follow-ups or missed diagnoses.

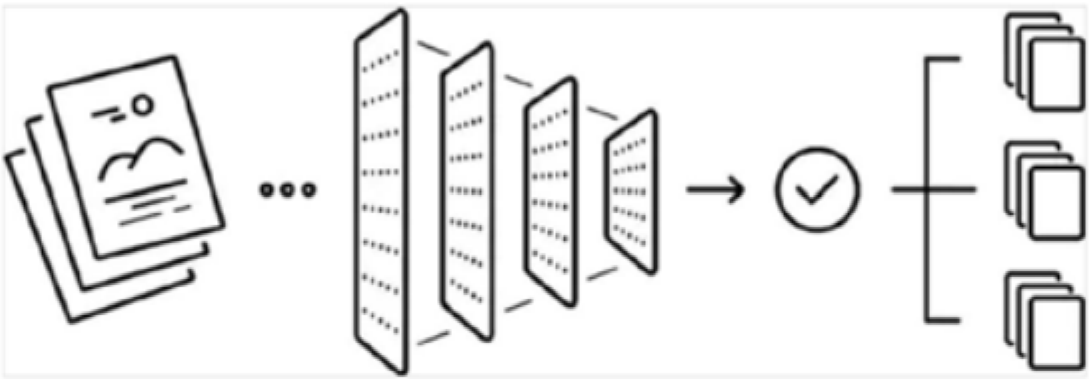


FIG 3.1 Basic process inside a CNN model

By thoroughly evaluating the model across these metrics, we were able to ensure that it met the necessary performance standards for deployment in a real-world cancer detection application.

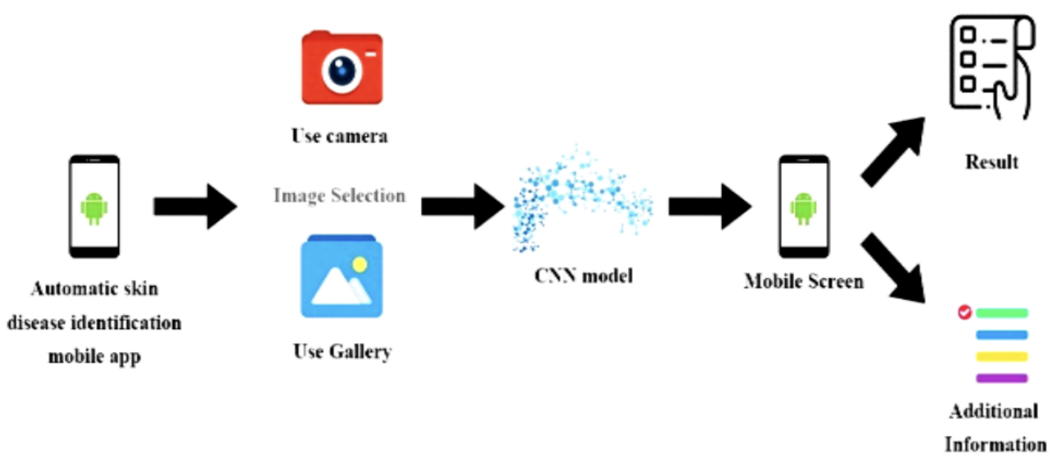


FIG 3.2 OVERALL ARCHITECTURE FOR SKIN CANCER DETECTION APP

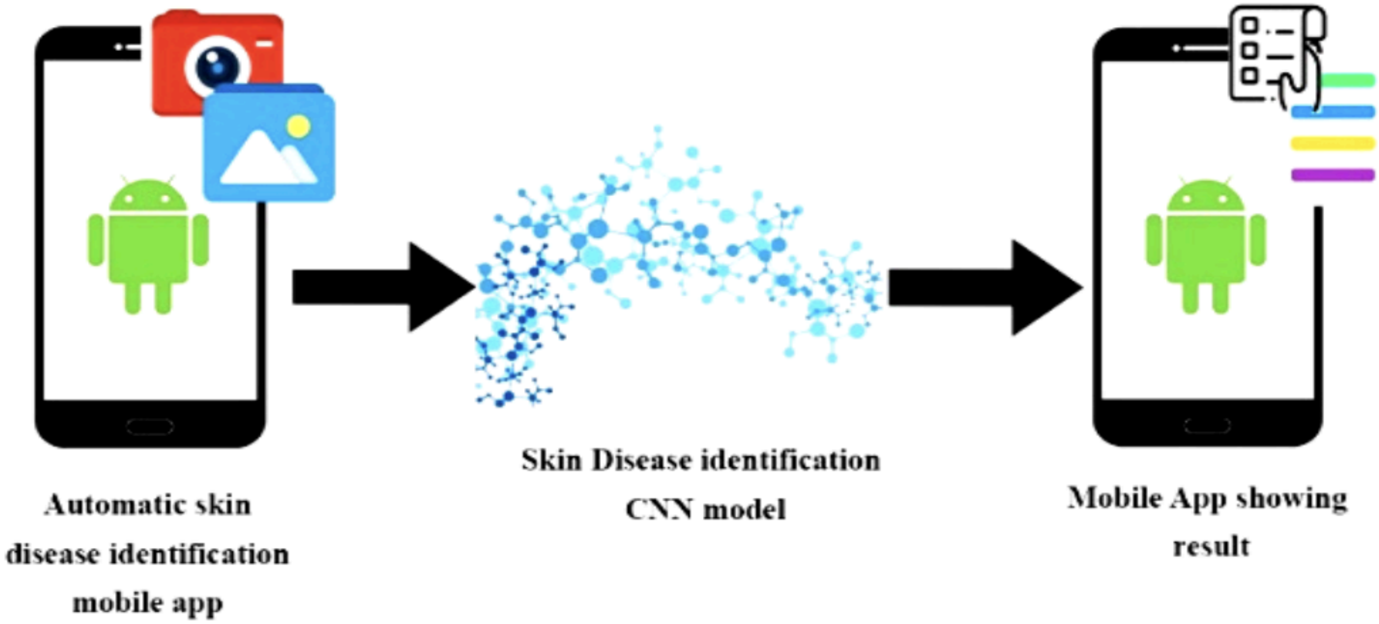


FIG 3.3 Model Functioning along with the application

The overall architecture of the cancer detection app is built on a robust client-server model, integrating a deep learning-based Convolutional Neural Network (CNN) for image classification with a mobile frontend developed using React Native Expo and a backend powered by FastAPI. The React Native app allows users to upload medical images, either through the device camera or gallery, which are then sent to the FastAPI backend via RESTful API calls. The backend processes the images, utilizing the pre-trained CNN model for cancer detection, and returns the results—such as classification labels (cancerous or non-cancerous) and confidence scores—back to the app. The frontend displays the results, along with visualizations like Grad-CAM to highlight key features influencing the predictions. Secure communication is maintained through HTTPS, and user data is protected with authentication protocols like JWT. This architecture ensures seamless integration between the frontend and backend, providing an intuitive, efficient, and secure solution for real-time cancer detection on mobile devices.

CHAPTER 4

RESULTS AND DISCUSSION

The conclusions made from the data show that the cancer detection mobile app is a valid instrument for early detection of skin cancer, which uses deep learning technology to make precise evaluations. On the other hand, the CNN models that delivered the highest accuracy rates are indicative of the power of AI in improving the diagnosis of dermal diseases. What accuracy plus memory indicates is that the model can tell apart the various kinds of non-cancer from the cancer types which are vital in mitigating unnecessary anxiety among users while at the same time keeping the suspicious cases that could be cancer for further examination.

The user engagement metrics depict the app as a function of both a diagnostic tool and an instrument that spirals awareness among users about skin cancer. The positive feedback regarding usability indicates that people are more likely to use technologies of this kind when they are made with the user alongside them. The huge sum of user satisfaction showcases that the mobile application is well-matched to the healthcare sector, at the same time winning over the patients' involvement in monitoring their health.

Nevertheless, many problems have been detected during the evaluation process. For a start, a notable issue was the different levels of image quality preferred by participants. Some images had low details and were not clear, this is likely to affect classification accuracy. To solve this problem, the app's future versions could involve users' instructions or tips on how to take high-quality images for analysis.

Moreover, although the application did show a remarkably high level of performance under the conditions of the pilot, larger-sized and more diverse samples are the areas of the incoming research, and further validation is needed to determine that the system can generalize across various demographics and skin types. A wider and varied dataset will assist in the model gaining robustness and eliminating the issues of underrepresented groups.

One more thing that can be improved is the implementation of a feedback mechanism when users report their results after the application gives its recommendations. The AI model would be receiving such information as data from its actual usage, thereby constructing a virtuous circle of learning.

To sum it all up, these results show that a mobile application that is built upon deep learning for cancer detection can be very useful in early diagnosis and patient education concerning skin cancer. The benefits of the joint usage of tech and design solutions will be harnessed by empowering individuals to own their health management and thus addressing rapid medical interventions are also made easy.

VISION:

The future prospects of the cancer detection app hold great potential for enhancing early diagnosis and expanding its capabilities to benefit a broader audience. As advancements in deep learning and medical imaging continue, the app can evolve by integrating more sophisticated AI models, including multi-class classification for detecting different types of cancers, and improving diagnostic accuracy through continuous model refinement using larger and more diverse datasets.

One key direction for future development is the expansion of diagnostic capabilities. By incorporating additional features, such as the ability to analyze histopathological slides, CT scans, or MRI images, the app could be extended to assist in the detection of a wider range of cancers. This could lead to a more comprehensive diagnostic tool for both healthcare providers and patients.

Additionally, real-time data analysis and the integration of telemedicine features could allow doctors to remotely diagnose and monitor patients' conditions. This could greatly enhance access to healthcare in underserved or rural areas, where patients may have limited access to medical facilities. The app could also incorporate chatbot features powered by AI to provide patients with immediate guidance on treatment options, lifestyle changes, or referrals to specialists.

With continuous updates, advancements in AI, and integration with new medical technologies, the cancer detection app can play a transformative role in revolutionizing cancer diagnosis.

OUTPUT SCREENSHOTS:

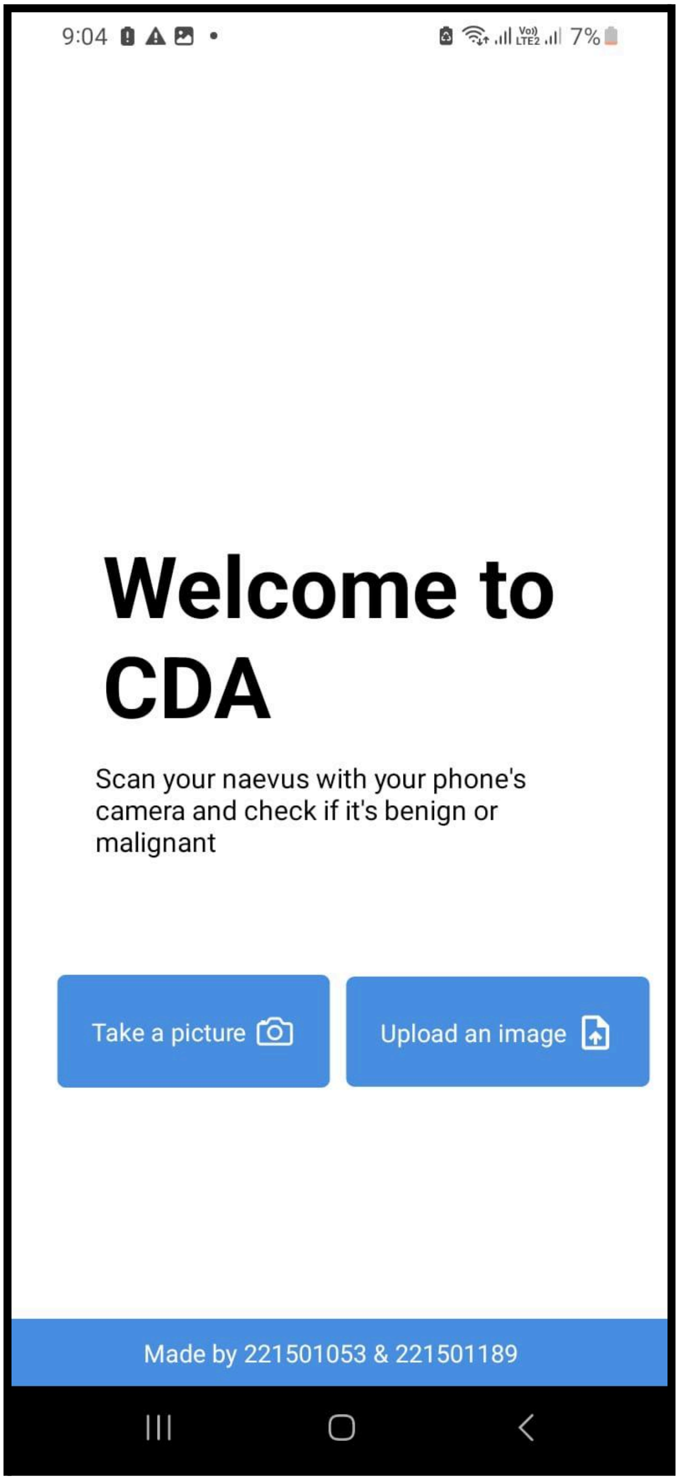


FIG 4.1 Home Interface of the App

FIG 4.2 Scan Interface of the App

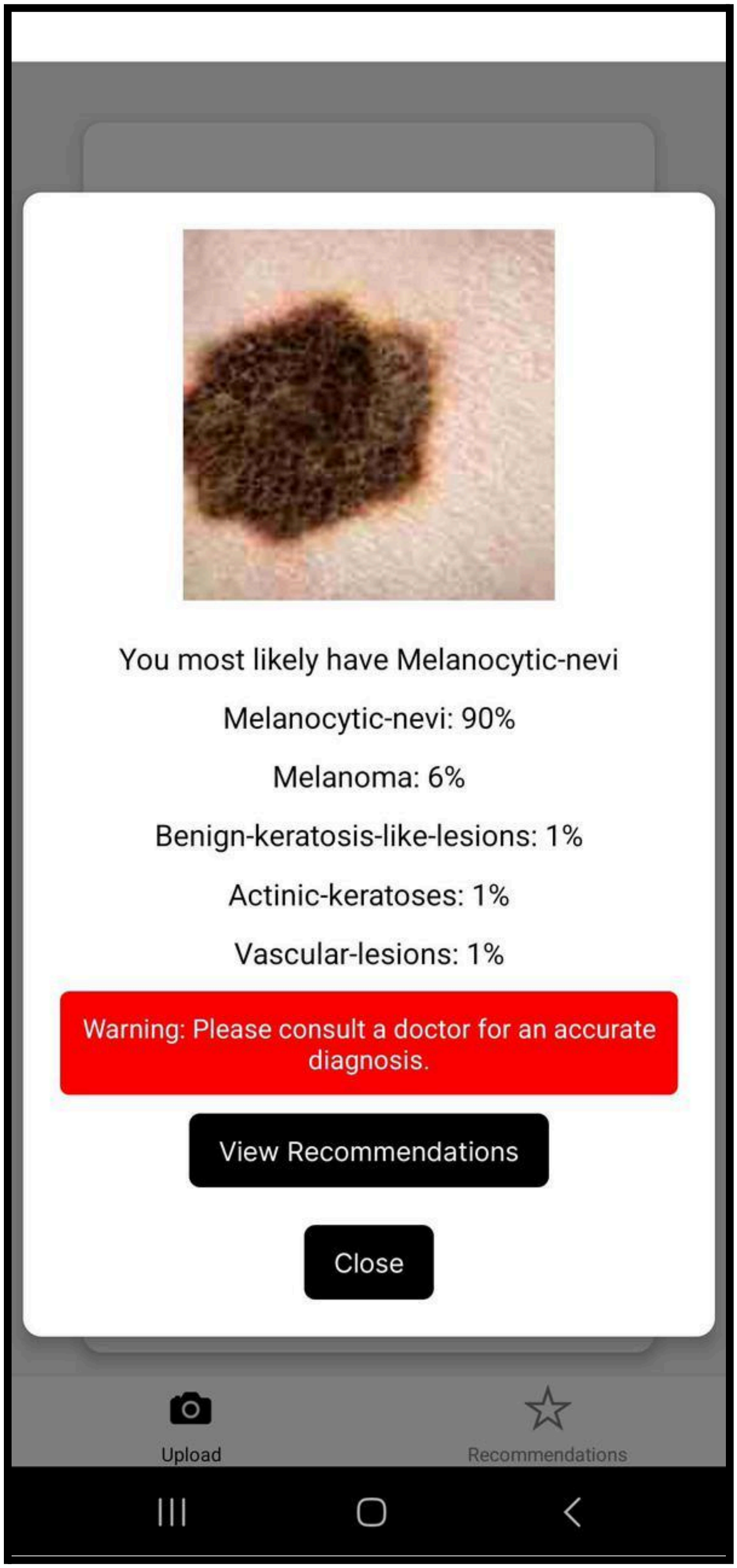


FIG 4.3 Results of the App

REFERENCES

[1] Ball, G. *et al.* 2002. An integrated approach utilizing artificial neural networks and SELDI mass spectrometry for the classification of human tumors and rapid identification of potential biomarkers. *Bioinformatics* 18: 395–404.

[2] Li, J. *et al.* 2002. Proteomics and bioinformatics approaches for identification of serum biomarkers to detect breast cancer

[3] Schafer, J.L. 1997. Analysis of Incomplete Multivariate Data. No. 72 of Monographs on Statistics and Applied Probability. Chapman & Hall/CRC. London.

[4] Hoffman, P. & G. Grinstein. 2000. Multidimensional information visualization for data mining with application for machine learning classifiers. *In* Information Visualization in Data Mining and Knowledge Discovery. Morgan Kaufmann. San Francisco.

[5]B. M. Gayathri and C. P. Sumathi, "Comparative study of relevance vector machine with various machine learning techniques used for detecting cancer", 2016.

[6] F. Mohd and M Thomas, "Comparison of different classification techniques using WEKA for Breast cancer", 2007.

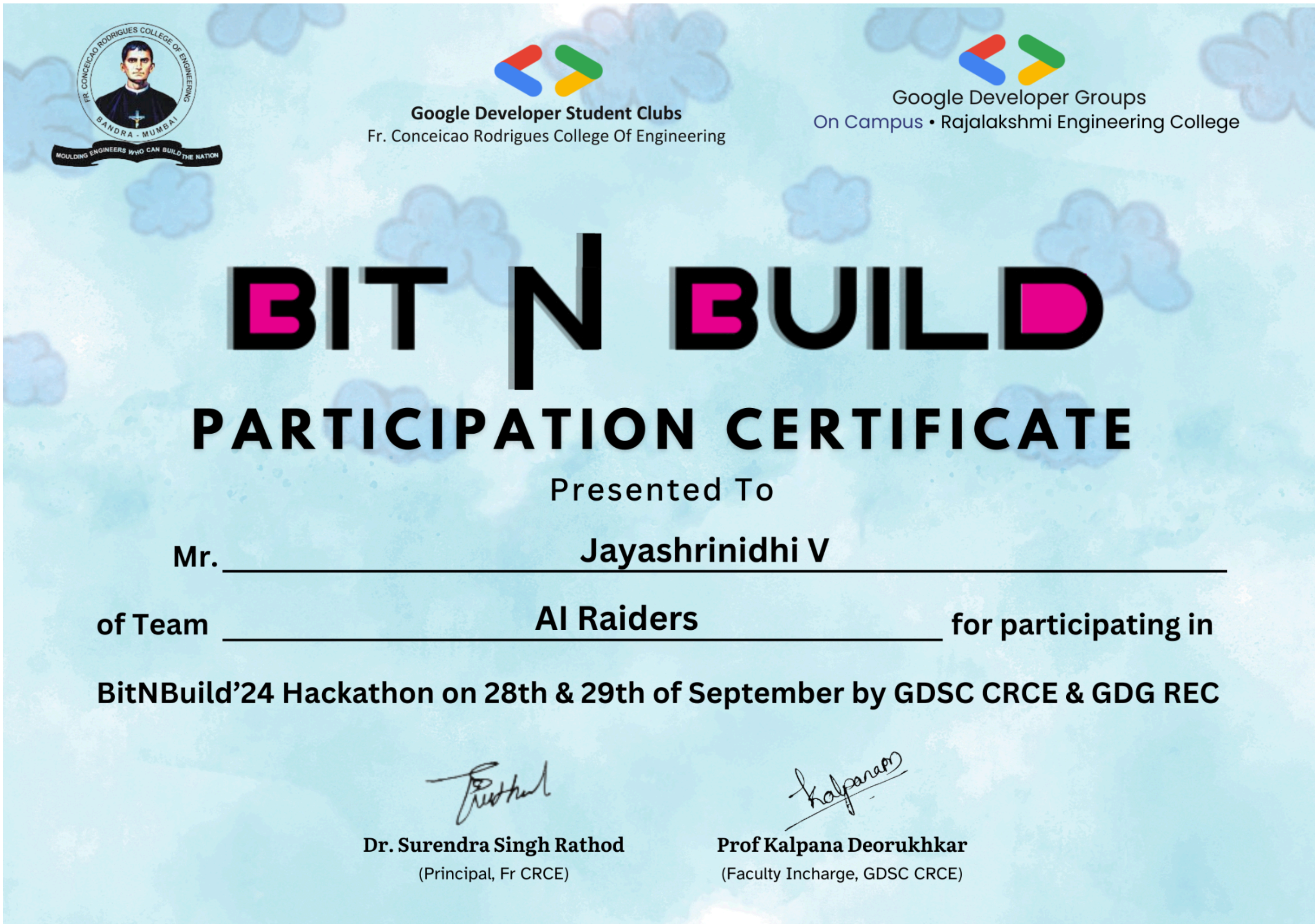
[7] T Choudhury, V Kumar, D Nigam and B Mandal, "Intelligent classification of lung & oral cancer through diverse data mining algorithms", International Conference on Micro-Electronics and Telecommunication Engineering, 2016.

[8] T Choudhury, V Kumar and D Nigam, "Intelligent Classification & Clustering Of Lung & Oral Cancer through Decision Tree & Genetic Algorithm", *International Journal of Advanced Research in Computer Science and Software Engineering*, 2015.

[9] I. Kononenko, "Machine learning for medical diagnosis: history state of the art and perspective", vol. 23, 2001.

[10]B. Networks, F. Faltin and R. Kenett, "Bayesian Networks", Encycl. Stat. Qual. Reliab., vol. 1, no. 1, pp. 4, 2007.

OUTCOME



Google Developer Student Clubs
Fr. Conceicao Rodrigues College Of Engineering

Google Developer Groups
On Campus • Rajalakshmi Engineering College

BIT N BUILD

PARTICIPATION CERTIFICATE

Presented To

Mr. Jayashrinidhi V

of Team AI Raiders for participating in

BitNBuild'24 Hackathon on 28th & 29th of September by GDSC CRCE & GDG REC

Dr. Surendra Singh Rathod
(Principal, Fr CRCE)

Prof Kalpana Deorukhkar
(Faculty Incharge, GDSC CRCE)

OUTCOME


Google Developer Student Clubs
Fr. Conceicao Rodrigues College Of Engineering

Google Developer Groups
On Campus • Rajalakshmi Engineering College

BIT N BUILD

PARTICIPATION CERTIFICATE

Presented To

Mr. Prajein C K

of Team AI Raiders for participating in

BitNBuild'24 Hackathon on 28th & 29th of September by GDSC CRCE & GDG REC

Dr. Surendra Singh Rathod
(Principal, Fr CRCE)

Prof Kalpana Deorukhkar
(Faculty Incharge, GDSC CRCE)

OUTCOME

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY
KATTANKULATHUR, CHENNAI - 603 203
DEPARTMENT OF COMPUTING TECHNOLOGIES
in association with
IE(I) STUDENTS' CHAPTER

CERTIFICATE OF PARTICIPATION

This is to certify that Jayashrinidhi V of Rajalakshmi Engineering college participated in the National Convention on Unlocking the Potential of Artificial Intelligence and Machine Learning – **Model Competition** conducted on 7th November 2024, organized by the Department of Computing Technologies in association with IE(I) Students' Chapter, SRM Institute of Science and Technology, Kattankulathur, Chennai.

Dr M Uma Devi, FIE	**Prof. (Dr.) G Niranjana**	**Prof. (Dr.) M Pushapalatha**
Faculty Advisor	Professor & Head, CTECH	Associate Chairperson, SOC
IE(I) Students' Chapter	SRMIST, KTR.	SRMIST, KTR

OUTCOME

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY
KATTANKULATHUR, CHENNAI - 603 203
DEPARTMENT OF COMPUTING TECHNOLOGIES
in association with
IE(I) STUDENTS' CHAPTER

CERTIFICATE OF PARTICIPATION

This is to certify that Prajein C K of Rajalakshmi Engineering college participated in the National Convention on Unlocking the Potential of Artificial Intelligence and Machine Learning – **Model Competition** conducted on 7th November 2024, organized by the Department of Computing Technologies in association with IE(I) Students' Chapter, SRM Institute of Science and Technology, Kattankulathur, Chennai.

Dr M Uma Devi, FIE	Prof. (Dr.) G Niranjana	Prof. (Dr.) M Pushapalatha
Faculty Advisor	Professor & Head, CTECH	Associate Chairperson, SOC
IE(I) Students' Chapter	SRMIST, KTR.	SRMIST, KTR