International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Skin Disease Detection

Prof. Avanti Patil [a], Rahul Ingale [b], Rahul Badachi [c], Rohit Swami [d], Sandesh Zele [e]

[a,b,c,d,e] Department of computer Science and Engineering ,Angadi Institute of technology & management, Belagavi

ABSTRACT :

The proposed "Skin Disease Detection" project is a healthcare initiative that uses machine learning and deep learning technology, including distinct deep learning models apart from CNNs, to enhance the diagnosis of various skin conditions and diseases. Skin diseases can range from common ailments like acne, dermatitis, and eczema, to more serious conditions like melanoma and cold sores. The project provides a user-friendly and accurate solution, improving healthcare outcomes and patient well-being. The chosen deep learning models like Recurrent Neural Networks (RNNs), Long Short-Term Memory networks (LSTMSs), Transformer models (BERT, GPT), distinct from CNNs, and a comprehensive methodology are proposed. An architecture diagram visually represents the complex connection between components, while design diagrams explore into specific system complexities. The expected results enhances improvements in accuracy and efficiency, validating the efficacy of the proposed methodology.

Keywords: Melanoma, Deep learning, RNNs, LSTMs, BERT, GPT.

Introduction :

Skin diseases present a significant global health burden, affecting millions of individuals and posing challenges in diagnosis and treatment. From common conditions like acne and eczema to more severe ailments such as melanoma, the accurate identification of skin diseases is crucial for effective management and prevention of complications. However, dermatological diagnostics have traditionally relied heavily on subjective visual assessment by healthcare professionals, leading to variability in diagnoses and potential delays in treatment initiation.

In recent years, the emergence of artificial intelligence (AI) technologies has sparked considerable interest in leveraging machine learning(ML) and deep learning(DL) algorithms to enhance healthcare delivery. Among these innovations, Skin disease detection stands out as a pioneering AI-powered skin disease classifier designed to transform the landscape of dermatological diagnostics. By harnessing the capabilities of deep learning algorithms, Skin disease detection offers a revolutionary platform for automated analysis of skin lesions, providing rapid and accurate identification of various dermatological conditions.

The development of Skin disease detection represents a convergence of cutting-edge technology and clinical expertise, with the goal of addressing longstanding challenges in dermatological practice. Through its intuitive interface, individuals can easily upload images of skin lesions, allowing Skin disease detection's neural network to analyze and classify the condition with unprecedented accuracy. Leveraging a vast repository of annotated skin images, Skin disease detection has been meticulously trained to recognize patterns and features indicative of different skin diseases, including acne, cold sores, eczema, dermatitis, and melanoma.

we explore the underlying principles and technical advancements that underpin Skin disease detection's functionality. We goes through the complex workings of ML and DL algorithms, examining the process of data acquisition, preprocessing, feature extraction, and classification. Furthermore, we assess the performance of Skin disease detection in real-world settings, evaluating its accuracy, sensitivity, specificity, and clinical utility in comparison to traditional diagnostic methods.

Beyond technological prowess, Skin disease detection holds profound implications for both patients and healthcare providers. For individuals, Skin disease detection offers a convenient and accessible means of obtaining timely and reliable diagnostic insights, empowering them to take proactive steps towards their skin health. For healthcare professionals, Skin disease detection serves as a valuable decision support tool, augmenting their clinical expertise and facilitating more efficient classification, diagnosis, and treatment planning.

The integration of AI-driven diagnostic tools like Skin disease detection into clinical practice is not without challenges. Concerns related to data privacy, regulatory compliance, and ethical considerations must be carefully addressed to ensure the responsible and ethical deployment of AI/ML technologies in healthcare settings. Moreover, ongoing research and development efforts are needed to continually improve the performance and generalizability of Skin disease detection across diverse patient populations and dermatological conditions.

Methodology

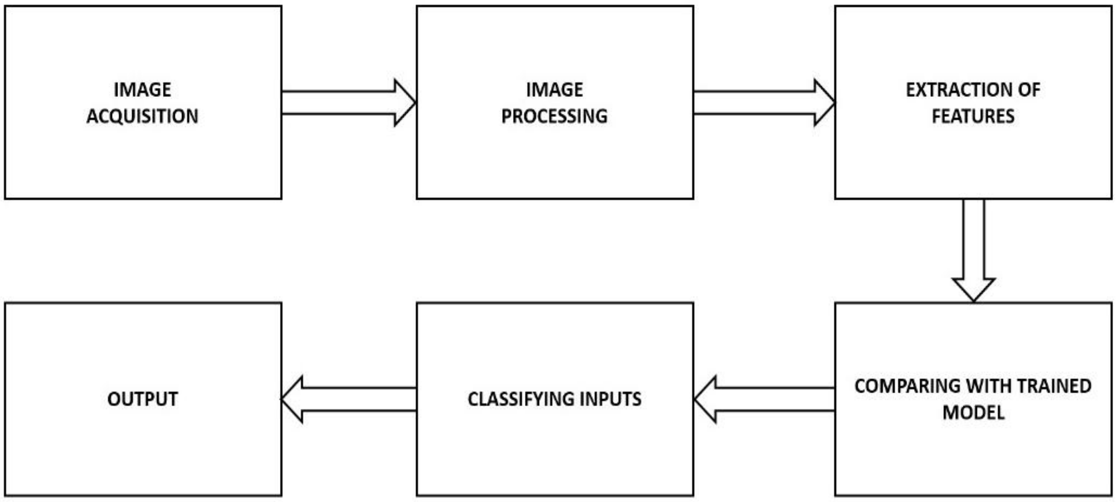

Fig 2.1: Block Diagram

i) **Data Collection:**

- Datasets used for this project are extracted from kaggle towards skin disease Detection . It consists of 8000 images of skin disease. The training data consists of 6000 images and testing data consists of 2000 images.

ii) **Image Preprocessing:**

- Image preprocessing is done by using OPEN CV and NUMPY.
- OpenCV:
  a) OpenCV-Python library of Python bindings in designed unravel computer vision problems.
  b) OpenCV-Python makes use Num py, by which may highly optimized library numerical operations a MATLAB-style syntax.
  c) All tin Open CV array are structures converted a and from Num py arrays.
  d) This also makes it easier to integrate other a libraries is that use Num py SciPy and Matplotlib.
  e) OpenCV to be capable image analysis and processing.
- NumPy:
  a) NumPy, that stands Numerical Python, be a library consisting of multi-dimensional as array objects and set a routines for processing those arrays.
  b) Using as Num Py, mathematical and logical on operations are arrays in often performed.
  c) The array object in NumPy is named ndarray, it provides tons of supporting functions that make working with nedarray very easy.
  d) NumPy is an open-source numerical Python library. Num Py a extension o Numeric and Num array.
  e) Num py contains random number generators. NumPy may wrapper around library implemented in C.
  f) Pandas is objects reply heavily NumPy objects. Essentially, Pandas extends Numpy.

iii) **Image Segmentation & Feature Extraction:**

- Image segmentation is a process of dividing image into regions or categories. In the dermoscopic images two types of fabric things first normal skin and second is lesion area so here we have done segmentation with Otsu thresholding technique. Using Texture-Based segmentation extracting the features from the image. GLCM (Gray Level Co-occurrence Matrix) is the statistical method examining the spatial relationship between the pixel. This technique works by creating the cooccurrence matrix were to calculate the frequency of occurrence of a pixel with the grey-level value is adjacent to a pixel with grey-level value j in any given direction and selected separating distance The GLCM matrix gives four statistics Correlation, Contrast, Energy, Homogeneity. There some problem in segmentation of dermoscopic images due to the contrast of images like under segmentation and over-segmentation so we are concentrating on segmentation based on texture features.

iv) Classification:

- The output of the Non-CNN's feature extraction layers is flattened and passed through one or more fully connected layers. These layers combine the extracted features to make predictions about the input image's class. The final layer often consists of a SoftMax activation function, which converts the raw output into a probability distribution over different skin disease class. The class with the highest probability is then selected as the predicted skin disease.

v) Output:

- The final output of the classifier is the predicted skin disease name

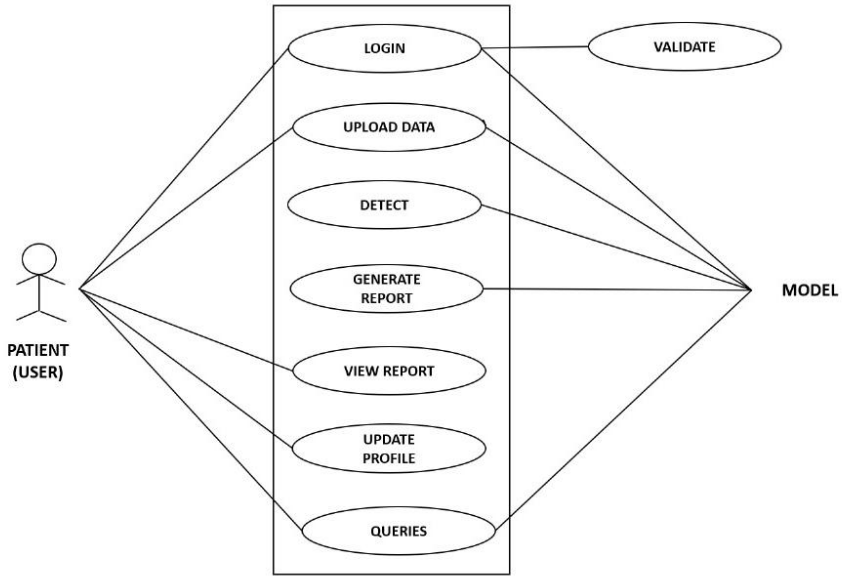

Fig 2.2: Use Case Diagram

Fig 2.2 shows the use case diagram here. Registration/login verification helps the user interact with the system to register a new account or log in to an existing account. The system verifies the user's credentials. The administrator downloads the stream from the user report. The use case is a system that automatically detects whether a patient has diabetes or not. The system generates a report based on the uploaded data or the detection process. The user views the report generated by the system. The user updates their profile information. The user logs out of the system.

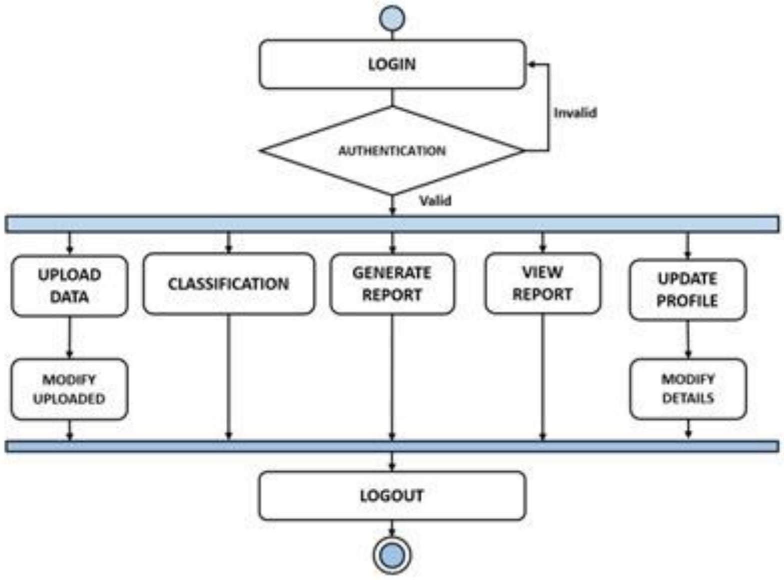

Fig 2.3: Activity Diagram

Fig 2.3 represents activity flow for a user interaction system involving several functionalities. The process begins with the user logging in, followed by an authentication step that determines if the credentials are valid or invalid. If the authentication is invalid, the process terminates; otherwise, it continues to the main functionalities. The user can perform various actions such as uploading data, classifying data, generating reports, viewing reports, and updating their profile. Each of these actions can further lead to specific sub-tasks like modifying uploaded data or modifying profile details. Once the user has completed their tasks, they can log out of the system, marking the end of the session. This flowchart clearly delineates the steps and options available to the user, ensuring a structured and efficient interaction with the system.

Testing

A. Black Box Testing:

Black box testing focuses on testing the system's functionality without knowledge of its internal workings. Testers provide inputs to the system and verify that the outputs are as expected. For "Skin Disease Detection", black box testing would involve:

i. Providing various skin lesion images as inputs.
ii. Checking if the system correctly identifies and classifies the skin diseases in the images.
iii. Verifying that the recommended treatments are appropriate for the diagnosed conditions.

B. White Box Testing:

White box testing examines the internal structure and logic of the system. Testers have access to the system's code and perform tests to ensure that all paths and branches are executed correctly. For "Skin Disease Detection", white box testing would involve:

i. Examining the Non-CNN model architecture and ensuring that it has been implemented correctly.
ii. Analyzing the preprocessing steps to confirm that the images are being prepared appropriately for input into the model.
iii. Checking the implementation of the recommendation system to ensure that it provides accurate and relevant treatment suggestions.

C. Unit Testing:

Unit testing focuses on testing individual components or units of the system in isolation. For "Skin Disease Detection", unit testing would involve:

i. Testing each function or method responsible for image preprocessing, model training, prediction, and treatment recommendation.
ii. Ensuring that each unit behaves as expected and handles edge cases appropriately.

D. Integration Testing:

Integration testing verifies that individual units or components of the system work together seamlessly as a whole. For "Skin Disease Detection", integration testing involves:

i. Testing the interaction between the image preprocessing module, the Non-CNN model, and the treatment recommendation system.
ii. Verifying that data flows correctly between different components of the system and that they communicate effectively.

E. System Testing:

System testing evaluates the system as a whole to ensure that it meets the specified requirements and functions correctly in its intended environment. For "Skin Disease Detection", system testing would involve:

i. Testing the entire system end-to-end, from image input to disease classification to treatment recommendation.
ii. Assessing the system's performance, scalability, and robustness under various conditions.
iii. Verifying that the user interface is intuitive and user-friendly.

F. Accuracy Testing:

evaluates the accuracy of the skin disease classification performed by "Skin Disease Detection". It involves comparing the system's predictions with ground truth labels for a set of test images.

1. Algorithm Steps

Step 1: Data Collection

Step 2: Preprocessing

Example: Resize images to 100x100 pixels and normalize pixel values

```
def preprocess_image(image):
    resized_image = tf.image.resize(image, (100, 100))
    normalized_image = resized_image / 255.0  # Normalize pixel values
    return normalized_image
```

Step 3: Feature Extraction using Vision Transformer

```
model = models.Sequential([
    model.add(Convolution2D(32,(3,3),input_shape = (64,64,3), activation = 'relu')),
    model.add(MaxPooling2D(pool_size = (2,2))),
    model.add(Flatten()),
    model.summary(),
    model.add(Dense(128, kernel_initializer='uniform', activation='relu')),
    model.add(Dense(units=64, kernel_initializer='uniform', activation='relu')),
    model.add(Dense(units=5, kernel_initializer='uniform', activation='softmax'))
])
```

Step 4: Model Selection

```
model.compile(optimizer='SGD',
            loss='binary_crossentropy',
            metrics=['accuracy'])
```

Step 5: Training

```
train_datagen = ImageDataGenerator(rescale = 1./255, shear_range = 0.2, zoom_range = 0.2, height_shift_range = 0.2, width_shift_range = 0.2, horizontal_flip = True, vertical_flip = True)
test_datagen = ImageDataGenerator(rescale = 1)
x_train =
train_datagen.flow_from_directory(r"C:\Users\rohit\OneDrive\Desktop\Skin Diseases\train", target_size=(64, 64), batch_size=32, class_mode='categorical')
x_test =
train_datagen.flow_from_directory("C:/Users/rohit/OneDrive/Desktop/Skin Diseases/test", target_size=(64, 64), batch_size=32, class_mode='categorical')
```

Step 6: Evaluation

```
model.fit(x_train, epochs=10, validation_data=x_test)
evaluation = model.evaluate(x_test)
print(f"Test Loss: {evaluation[0]}")
print(f"Test Accuracy: {evaluation[1]}")
model.fit(x_train, steps_per_epoch=50, epochs=200, validation_data=x_test, validation_steps=10)
model.save("Skin_Diseases")
```

Accuracy testing

Conclusion

The skin disease detection model marks a significant leap forward in dermatological diagnosis by utilizing Non-Convolutional Neural Networks (Non-CNNs) to achieve exceptional accuracy and clinical significance. The implementation of deep learning and machine learning algorithms with artificial intelligence technology allows for rapid and accurate identification of dermatological conditions, enabling healthcare professionals to intervene early, plan treatments effectively, and manage diseases efficiently, thus transforming the field of dermatological care. To maximize the potential of this groundbreaking technology, continuous research and development are crucial. Future enhancements will further improve the model’s performance, broaden its capabilities, and address any current limitations.

REFERENCES :

1. Tanzina Afroz Rimi, Nishat Sultana, Md. Ferdouse (2020). Skin Diseases Detection Using CNN. Proceedings of the International Conference on Intelligent Computing and Control Systems (ICICCS 2020) IEEE Xplore Part Number:CFP20K74-ART; ISBN: 978-1-7281-4876-2.

2. E.Megha, Dr.Rini Jones.SB (2020). Skin Problem Identification. Proceedings of the Third International Conference on Smart Systems and Inventive Technology (ICSSIT 2020) IEEE Xplore Part Number: CFP20P17-ART; ISBN: 978-1-7281-5821-1.

3. Jainesh R, Vishal W, Aniruddh Sodha, Dr. Prasenjit B. (2018). Diagnosis of skin diseases using CNN. Proceedings of the second International conference on Electronics, Communication and Aerospace Technology (ICECA 2018) IEEE Conference Record # 42487; IEEE Xplore ISBN:978-1-5386-0965-1.

4. Rathod, J., Wazhmode, V., Sodha, A., & Bhavathankar, P. (2018, March). Diagnosis of skin diseases using Convolutional Neural Networks. In 2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA) (pp. 1048-1051). IEEE. doi:10.1109/ICECA.2018.8474593.

5. Mane, S., & Shinde, S. (2018, August). A method for melanoma skin cancer detection using dermoscopy images. In 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)(pp. 1-6). IEEE. doi:10.1109/ICCUBEA.2018.8697804.

6. Bajaj, L., Kumar, H., & Hasija, Y. (2018). Automated System for Prediction of Skin Disease using Image Processing and Machine Learning. International Journal of Computers and Applications, 180(19), 9–12. doi:10.5120/ijca2018916428.

7. Ansari, U. B., & Sarode, T. (2017). Skin cancer detection using image processing. Int Res J Eng Technol, 4(4), 2875–2881.

8. Soumya Sourav, Nikhil Garg, Yasha Hasija (2017). Detection of Skin Disorders through Image-Processin & ML. Proceedings of the International Conference on Intelligent Sustainable Systems (ICISS 2017) IEEE Xplore Compliant - Part Number:CFP17M19-ART, ISBN:978-1-5386-1959-9.

9. Nezhadian, F. K., & Rashidi, S. (2017, October). Melanoma skin cancer detection using color and new texture features. In 2017 Artificial Intelligence and Signal Processing Conference (AISP) (pp. 1-5). IEEE. doi:10.1109/AISP.2017.8324108.

10. Suganya, R. (2016, April). An automated computer aided diagnosis of skin lesions detection and classification for dermoscopy images. In 2016 International Conference on Recent Trends in Information Technology (ICRTIT)(pp. 1-5). IEEE. doi:10.1109/ICRTIT.2016.7569538.