D Y PATIL
INTERNATIONAL
UNIVERSITY
AKURDI PUNE

Optimizing Skin Cancer Detection Neural Networks: SVD for Storage Reduction and Model Pruning for Compute Efficiency

Project report

submitted to D Y Patil International University, Akurdi, Pune
in partial fulfilment of full-time degree.

BTech Computer Science and Engineering

(Data Science –Track)

Submitted By:

Prathamesh Deshpande	20200802056
Aditya Ambre	20200802023
Prathamesh Dhas	20200802212

Under the Guidance of

Dr. Bahaubali Shiragapur

Department of Computer Science and Engineering

D Y Patil International University, Akurdi,Pune, INDIA, 411044

[Session 2020-2024]

D Y PATIL
INTERNATIONAL
UNIVERSITY
AKURDI PUNE

CERTIFICATE

This is to certify that the project entitled **Optimizing Skin Cancer Detection Neural Networks: SVD for Storage Reduction and Model Pruning for Compute Efficiency** submitted by:

Prathamesh Deshpande	20200802056
Aditya Ambre	20200802023
Prathamesh Dhas	20200802212

is the partial fulfillment of the requirements for the award of degree of Bachelor of Technology in Computer Science Engineering and Applications is an authentic work carried out by them under my supervision and guidance.

Dr. Bahubali Shiragapur
(Supervisor)

Dr. Rahul Sharma
(Project Co-ordinator)

Dr Bahubali Shiragapur
Director
School of Computer Science Engineering & Applications
D Y Patil International University, Akurdi
Pune, 411044, Maharashtra, INDIA

DECLARATION

We, hereby declare that the following report which is being presented in the Major Project entitled as **Optimizing Skin Cancer Detection Neural Networks: SVD for Storage Reduction and Model Pruning for Compute Efficiency** is an authentic documentation of our own original work to the best of our knowledge. The following project and its report in part or whole, has not been presented or submitted by us for any purpose in any other institute or organization. Any contribution made to the research by others, with whom we have worked at D Y Patil International University, Akurdi, Pune or elsewhere, is explicitly acknowledged in the report.

Prathamesh Deshpande	20200802056	____________________
Aditya Ambre	20200802023	____________________
Prathamesh Dhas	20200802212	____________________

ACKNOWLEDGEMENT

With due respect, we express our deep sense of gratitude to our respected guide and coordinator **Dr. Bahubali Shiragapur**, for his/her valuable help and guidance. We are thankful for the encouragement that he/she has given us in completing this project successfully.

It is imperative for us to mention the fact that the report of major project could not have been accomplished without the periodic suggestions and advice of our project supervisor Dr. Bahubali Shiragapur and **Dr Rahul Sharma** (Project Co-ordinator).

We are also grateful to our respected Director, Dr. Bahubali Shiragapur and Hon'ble Vice Chancellor, DYPIU, Akurdi, Prof. Prabhat Ranjan for permitting us to utilize all the necessary facilities of the college.

We are also thankful to all the other faculty, staff members and laboratory attendants of our department for their kind cooperation and help. Last but certainly not the least; we would like to express our deep appreciation towards our family members and batch mates for providing support and encouragement.

Prathamesh Deshpande (20200802056)
Aditya Ambre (20200802023)
Prathamesh Dhas (20200802212)

Abstract

Malignant and non-malignant forms of skin cancer constitute a significant global issue. The most dangerous kind, melanoma, can spread quickly and has a high mortality rate if it is not detected in its early stages. Although non-melanoma skin malignancies are more common, untreated cases of squamous and basal cell carcinomas can result in deformity and morbidity. Among the primary issues at hand are late detection, low doctor-to-patient ratios, lack of delivery mechanisms, and lack of awareness. Many Deep Neural Networks and Deep Convolution Neural Networks have been developed with the aim of early skin cancer diagnosis. Large designs like these demand a lot of computation and storage, which might be expensive or unavailable in settings with limited resources. Further, these Models require millions of high resolution dermatoscopic Images for training resulting in increased storage requirements. This Article proposes a scalable and reusable approach to compress and reconstruct these images with feature preservation and minimal loss using Singular Value Decomposition (SVD) compressing a dataset of 2.72 GB to less than 1GB achieving a compression ratio of 3.77 and saving 73% storage space. It also discusses a simple, uninformed and naive Model pruning method which reduces convolution filters by a pruning factor iteratively until a minimal model is achieved. We Prune 4 developed baseline architectures and achieve a best reduction in learnable parameters of approximately 5% of original baseline model and model size reduction from 444MB to less than 10MB. We also propose a Web Application delivered via Medical Kiosk with capabilities like interacting with a chat-bot for diagnosis, PDF report generation for doctors and Appointment booking and other administrative tasks made easy bridging gap between doctors and patients.

Keywords: Convolution Neural Networks, Kiosk, Learnable Parameters, Model Pruning, Pruning Factor, Singular Value Decomposition

TABLE OF CONTENTS

List of Figures

List of Tables

1. INTRODUCTION

1.1. Background

The World Health Organisation (WHO) estimates that there are approximately 132,000 new cases of melanoma detected every year globally. Countries have different rates of skin cancer; those that have high UV exposure and a majority of fair-skinned populations—such as Australia, New Zealand, North America, and portions of Europe—have the highest rates. In tropical nations with strong UV indices, like India, the number of cases is rising, especially in fair-skinned people and those who spend a lot of time outside. According to the World Cancer Research Fund International, in 2020 there were over 150,000 new cases of melanoma worldwide, making it the 17th most common cancer globally. The highest incidence rates of melanoma are observed in countries with predominantly fair-skinned populations and high UV exposure, such as Australia and New Zealand.[1]

In India, less than 1% of all diagnosed cancers are skin cancers. While Basal Cell Carcinoma(BCC) is the most common skin malignancy worldwide, SCC(Squamous Cell Carcinoma) is reported to be the most prevalent skin cancer in India. Despite the lower incidence of skin cancer in India compared to the Western world, the absolute number of cases is significant due to the country's large population. Various studies from India have consistently shown SCC as the most common skin malignancy, contrasting with global trends where BCC predominates. Thus, although skin cancer incidence is lower in India, the large population base and a low doctor: patient ratio means the absolute number of cases is substantial.[2]

Supporting 1.4 billion people, the Indian healthcare system is a complex web of public and private organizations. Even with all of its progress, there are still issues including inadequate public funding, fragmentation, discrepancies between rural and urban regions, a lack of healthcare professionals, inadequate infrastructure, and restricted health insurance coverage. These difficulties are made worse by an increasing number of non-communicable diseases.

With the Final Objective being diagnosing Skin Cancer in Early Stages, there have been numerous methods proposed utilizing Machine Learning Techniques and Deep Neural

Networks and Convolutional Neural Networks. Deep Learning Architectures such as Alexnet, Resnet-50, Resnet-101 and many custom architectures have been trained and performed remarkably. DNNs and CNNs work on Raw Images and Data and identify and learn the discriminating Features during training and hence reduce the overhead of feature extraction before model training. However, developing and training of these models is very resource intensive due to complex architecture, high resolution and dimensionality of Images, High Compute Requirements. High Resolution Images also demand large storage space and these models are trained on millions of images to achieve benchmark performance.

Currently, Most of the Skin Classification based ML Models use the Images of the Skin Cancer in their Model for prediction but factors such as Age, Sex, and Localization and many more are also important for providing a more nuanced and personalized assessment. In Medical Domain, All these factors are very important to consider. For example, in women, the most common place for melanoma to develop is on the legs while in men, melanoma is most commonly found on the chest and back.[3]

1.2. Objectives

1. To Optimize Storage requirement of High resolution Dermatoscopic Images using Singular Value Decomposition (SVD) and Use a Naive Approach to Model pruning to Reduce Model Complexity and Training Time and Size to ensure faster training, deployment and inference times by models without compromising in accuracy.

2. To Design a Neural Network that can process Textual Features like Age, Sex, Localization i.e. Region of Body consisting the Skin Lesion along with Actual Image for skin cancer detection.

3. To create a Web Application for Booking Appointments, Automated diagnosis support via Custom Neural Network and Skin Cancer Report generation capabilities. The Booking Appointment and Auto-Diagnosis Module will be delivered via Medical Kiosks. Medical kiosks aim to close the gap between people and healthcare institutions by offering on-site screening services, which help to diagnose skin cancer early. They can perform preliminary exams, take and process pictures of skin lesions, and provide suggestions for more medical testing. The Kiosk will also be used for booking appointments with

Doctors in Proximity.

1.3. Problem statement

Rural areas face multiple obstacles in diagnosing skin cancer, especially pigmented lesions, due to factors like lack of knowledge, inadequate healthcare resources, remote locations of facilities, budgetary constraints, and restricted access to technology. Ignorance hinders prompt treatment and delays the detection of likely cases of skin cancer. Ignorance can hinder prevention and postpone the identification of potential cases of skin cancer. Inadequate resources for healthcare might lead to delays in diagnosis and appropriate patient care. The distance between them may deter people from seeking timely medical attention or follow-up appointments, which could delay diagnosis and treatment. People without adequate health insurance or financial resources may find it difficult to pay for transport, which restricts their access to skin diagnostic tests and preventative care.

In the Medical Field, Images have very High Resolution and dimensionality especially Dermatoscopic images of Skin lesions where fine-detailed patterns are very important. This increases the storage Requirements of the Medical Storage system and the Number of Features to be learned by the Models to focus on the Fine-detailed Patterns. This increases training time and computes the efficiency required for achieving benchmark performances. Since the number of Features to be learned increases, the number of learnable parameters(in 10s and 100s of millions), the model size, and training time for the model and inference time for predicting an image also increases. To train a CNN model effectively, powerful computational approaches are required like GPUs or TPUs.[4]

2. LITERATURE REVIEW

2.1. Literature review

India has a serious shortage in medical personnel, especially physicians, nurses, and paramedics. Both the cost and availability of healthcare are significantly impacted by this shortage, particularly in remote regions. One factor contributing to this problem is the a restricted capacity of medical and nursing schools to educate prospective employees. Among urban and rural areas, there is an evident disparity in the accessibility and quality of healthcare. Better infrastructures more skilled personnel, and specialized care are generally found in urban locations, whereas poor facilities and a shortage of human resources are prevalent in rural areas.[5]

There has been increasing concern in the Malwa belt of Punjab regarding the rising incidence of cancer, as highlighted by print and electronic media. Various studies have attempted to correlate this trend with environmental factors. While skin cancer remains less common, its incidence has been progressively increasing over the past few decades. Basal cell carcinoma (BCC), squamous cell carcinoma (SCC), and malignant melanoma are the most frequent skin cancers, with BCC and SCC collectively termed Non-Melanomaous Skin Cancers (NMSC). Globally, skin cancers are relatively uncommon, not ranking among the top ten common cancers, but there has been a progressive increase in incidence in recent decades.[2]

Despite the conventional belief that melanin provides Indians with protection against skin cancer, smaller studies suggest that non-melanoma skin cancers (NMSCs) may be increasing in India. Although national data is lacking, hospital-based studies report a significant number of cases requiring excision and reconstructive surgery. NMSCs, which arise from keratinocytes, are primarily caused by UV radiation from sun exposure. Other etiological factors include arsenic, radiation therapy, and hydrocarbons. Arsenic, found in contaminated drinking water and used in pesticides, is a significant carcinogen, especially in combination with UV exposure. Studies indicate elevated levels of arsenic, selenium, and mercury in water sources in Punjab, coupled with high pesticide usage.[2]

A Number of algorithms have been suggested for skin lesion classification, including decision trees, ensemble learning, optimization, feature augmentation and random forests. One of the suggested approach makes use of CNNs' special qualities in conjunction with machine learning to increase interpretability and accuracy. This approach uses CNNs ability to learn discriminatory patterns while training and using extracted features such as Color Histograms, Hu Moments for Shape and Haralick Textures for accurate diagnosis providing a condensed view of the images.[6]

The next approach increases the precision and effectiveness of skin lesion categorization by utilizing the benefits of both data augmentation and transfer learning. Skin lesion photos are resized to fit the AlexNet input size, color space converted, and noise removed. On the skin lesion datasets, AlexNet's pre-trained weights serve as a foundation for fine-tuning. In doing so, the training time and complexity are decreased while using the deep learning model's built-in feature learning capabilities. The skin lesion datasets are used to train AlexNet's final layers, which then modify the pre-trained features to suit the particular lesion classification job.[7].

Improving patient outcomes depends on early and precise skin cancer identification. For this purpose, deep convolutional neural networks (CNNs) have shown great promise because of their high accuracy and automation potential. This research presents an improved method for classifying skin cancer using transfer learning and a bespoke DCNN. They suggested a deep learning-based deep convolutional neural network (DCNN) model for the precise categorization of benign and malignant skin lesions.With a big dataset, our suggested DCNN model outperforms other deep learning (DL) models in terms of accuracy. Compared to other transfer learning models like AlexNet, ResNet, VGG-16, DenseNet, and MobileNet, the suggested DCNN model requires a significantly shorter execution time to process the output results.The HAM10000 dataset is used to evaluate the model, and the results show that we have the greatest training and testing accuracy (93.16% and 91.93%, respectively.[8]

The next approach combines LSTM with Mobilenet v2 for classification.The type of skin illness is classified using MobileNet V2, and the model's performance is improved using LSTM, which keeps track of the features' state information from its previous iteration of picture classification. MobileNet is a CNN-based model that is widely used for image classification, in contrast to MobileNet V2 . The primary benefit of utilizing the MobileNet architecture is that, in comparison to the traditional CNN model, it requires significantly less computing work, which makes it appropriate for use with mobile devices and PCs with less processing power.The MobileNet model is a convolution layer-based simplified structure that can be used to identify the detail based on two controllable characteristics that efficiently transition between the parameter's accuracy and latency. Reducing the size of the network is a benefit of the MobileNet paradigm.[9]

Patient Metadata and Images have also been used together to create a CNN plus ANN Combined model creating an efficient and resource-efficient model deployed on smaller devices. It achieved an accuracy of up to 92% and also reduced the over-fitting of the model. Both these models process respective data in parallel and later combine the results.[10]

2.2. Drawbacks of existing system

In conclusion a) The Current Lack of Skilled Professionals and Low Doctor: Patient Ratio is a concern in Remote Areas for Skin Cancer Diagnosis. High Transportation Cost, Medical Fess and Lack of Awareness are main factors that prevent these patients from getting regular diagnosis and consequently not getting diagnosis in early Stages of the Disease. b) There have been Number of Algorithms and Architectures proposed for diagnosis of Skin Cancer. Deep Neural Networks and Convolutional Networks have been very effective due to their complex architectures able to learn discriminatory patterns from high resolution images. Their complex architectures, high compute and Storage requirements demand for use of expensive hardware like GPUs and TPUs which may incur heavy costs in these resource-constrained environments (like rural healthcare settings). Complex and Deep Architectures lead to increased latency and large inference time after deployment if no specialized Hardware like GPUs is used. Larger Training Time also mean an increase time in overall training life-cycle which requires a lot of iterative fine-tuning. c) These Models are trained on Millions of High resolution Images to improve accuracy, scaling them into a medical application means storage of these images, which can lead to very large storage requirements. For example HAM10000 Dataset consisting of 10,000 Images is 2GB in Size. Hence, this approach of storage hinders scalability. d) Transfer Learning Approaches have been proposed to reduce model size, but it also poses challenges such as it can stifle the development of new, specialized models designed specifically for medical applications. e) There also have been many pruning methods for pruning models based on magnitute, similarity, sensitivity analysis which reduce computational load and are also informed methods. However, these are complex implementations and require complex analysis and need for fine-tuning to avoid a significant reduction in performance which increases overhead in terms of training time in model lifecycle.

2.3. Gaps Identified

1. Gap1: There is a huge lack of awareness , lack of skilled healthcare professionals in Remote Areas. There is a need for an Application Connecting and bridging the gap between these patients and healthcare professionals for early diagnosis of Skin Cancer.
2. Gap2: Deep Learning Architectures used in Skin Cancer Detection have complex architectures which lead to higher training and inference times, requirement of compute expensive hardware such as GPUs which can be difficult to access in the resource constrained environments.
3. Gap3: Millions of High Resolution Dermatoscopic Images are required for training of these architectures. This can increase storage requirements of the Technology Used.
4. Gap4: Textual Features such as Age, sex and localization of the lesion which can provide a more Nuanced and personalized diagnosis are not being utilized by many model architectures.

2.4. Objectives

1. The research study aims to study the effectiveness of compressing High-resolution Images using Singular-Value Decomposition for Reducing their storage size while preserving the features required for identification of skin lesion. We also propose a Simple and Naive Method for Pruning Number of Hidden Layer Neurons and Convolutional Filters to reduce Model Complexity and Model Size.
2. Next, We Design four Neural Network that can process Textual Features like Age, Sex, Localization i.e. Region of Body consisting the Skin Lesion along with Actual Image for

skin cancer detection and can be deployed on small scale applications and utilized and trained on less compute expensive hardware as well without compromising on accuracy metrics. The 4 Architectures are pruned using proposed method and the best one is selected for the application based on Training , Validation and Testing Metrics.

3. Finally, We deploy our Model on created web application to help with automated self-diagnosis along with a trained chat bot to help the patients and report generation capabilities to support doctors in diagnosis. We also create another web application for booking appointments, bill generation and for reducing other administrative overhead on small clinics and doctors.

3. PROPOSED METHODOLOGY

3.1. Key Terminologies

3.1.1. Convolutional Neural Networks(CNN)

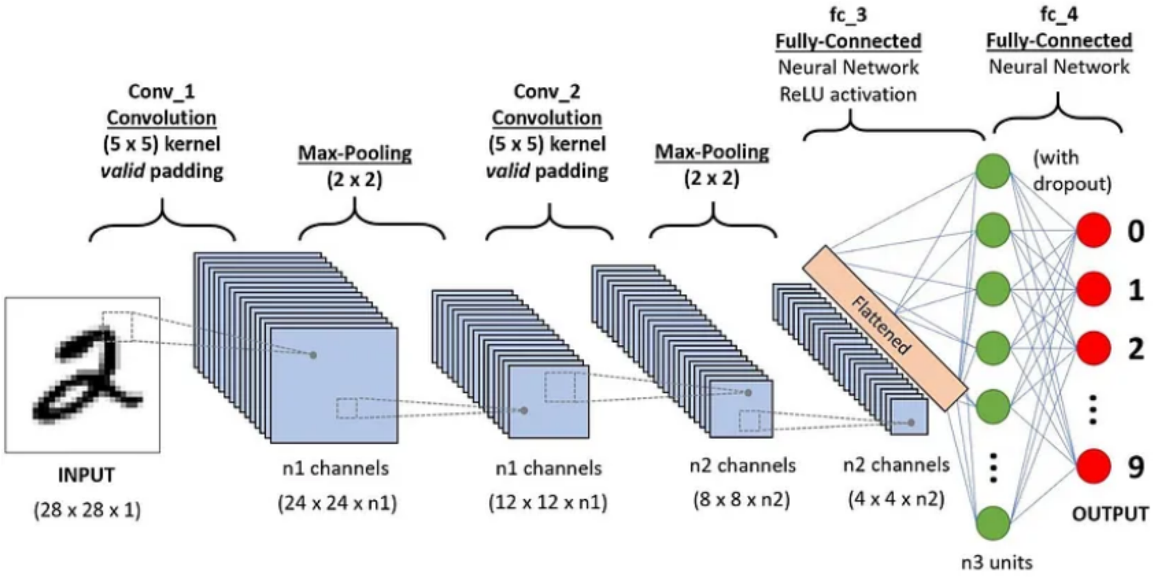


Figure 3.1: An simple CNN architecture, comprised of just five layers

There are four main areas in which the above CNN example's core operation may be divided.

1. The input layer will store the image's pixel values, same like in other ANN types.[11]
2. Then the Filter moves across the image as per stride and its kernel size reducing image dimensionality producing Feature Maps. The Filters Matrix Values are the weights that are learnt during training process to identify key features in an image after multiplying it with input region of image.[11]
3. To further reduce the amount of parameters inside that activation, the pooling layer will only downscale the input along its spatial dimensions capturing the features with highest , lowest activation or averaging out the region.[11]

4. Next, the fully-connected layers will carry out the tasks included in conventional ANNs and try to generate class scores from the activations, which will be utilised for classification. Additionally, it is advised that ReLU be utilised in between these layers in order to enhance performance.[11]

Formula: Element-wise Multiplication and Summation within the Kernel

$$Y[i,j] = \sum_{k,l} (W[k,l] \cdot X[i+k, j+l]) + b$$

$$\begin{aligned} Y &: \text{Output activation map} \\ W &: \text{Kernel weights} \\ X &: \text{Input data} \\ b &: \text{Bias term} \\ i,j &: \text{Indices of the output} \\ k,l &: \text{Indices of the kernel} \end{aligned}$$

3.1.2. Singular Value Decomposition (SVD)

Singular Value Decomposition (SVD) in image compression involves decomposing an image matrix A into three matrices U, Σ, and V^T, such that

$$A = U\Sigma V^T,$$

where U and V are orthogonal matrices and Σ is a diagonal matrix of singular values. By retaining only the largest singular values and their corresponding vectors, the image is approximated with fewer components, drastically reducing the data size. This approach effectively minimizes redundancy while preserving critical visual information, leading to efficient storage and transmission of images. The compressed image can be reconstructed by multiplying the truncated matrices, maintaining high visual fidelity despite the reduced data representation. .[12]

The SVD of matrix A $(m \times n)$ can be expressed as:

$$A_{(m\times n)} = U_{(m\times m)}\Sigma_{(m\times n)}V^T_{(n\times n)}$$

The matrices U and V^T are orthogonal. Singular values are found on the diagonal of the diagonal matrix

- **Singular Values:** In U and V^T, the singular values indicate the significance of every basis vector, or set of rows and columns. Greater single values suggest a higher contribution to the total variability of the data.[12]
- **Left Singular Vectors (U):** These vectors show which directions in the data there is the most volatility. They represent the fundamental organisation of the data, which frequently correlates to the primary components.[12]
- **Right Singular Vectors (V^T):** The vectors describing the directions of variation of the data points are called right singular vectors (V^T). In the basis that U defined, they stand in for the altered data points.[12]

3.2. Proposed Methodology

In this research study, we use the HAM10000 Dataset consisting of biopsy-proven dermatoscopic images [13] and 164 New Images for training captured and verified by Dermatologists. The New Images are Negative Samples, meaning they are images of Non-Cancerous Skin Lesion. This shall help the Model in differentiating between patterns of Skin Cancer Lesions and Non-Cancerous Skins. The New Samples are simple abrasions and the spots that develop subsequently from them. They were captured in a High Resolution Camera. The HAM10000 Dataset provided by International Skin Imaging Collaboration (ISIC) containing dermotoscopic images from various populations for seven classes of cancer: Actinic keratoses and intraepithelial carcinoma / Bowen's disease (akiec), basal cell carcinoma (bcc), benign keratosis-like lesions (solar lentigines / seborrheic keratoses and lichen-planus like keratoses, bkl), dermatofibroma (df), melanoma (mel), melanocytic nevi (nv) and vascular lesions (angiomas, angiokeratomas, pyogenic granulomas and hemorrhage, vasc).

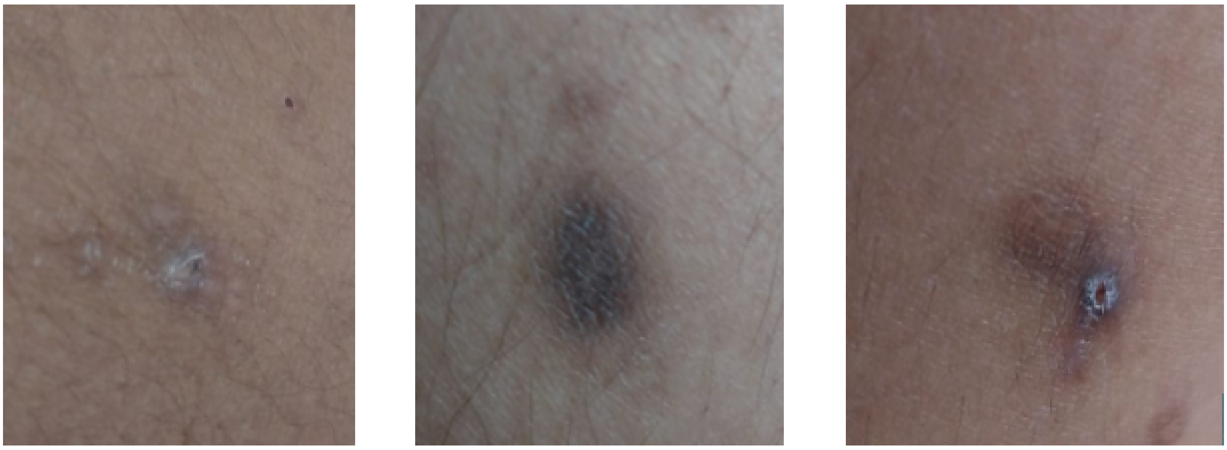


Figure 3.2: Sample Images as Negative Samples

The Overall Dataset is Highly Imbalanced and Consists of More than 8 classes including our new negative samples. We simplify this by keeping classes Melanoma and Melanocytic Nevi and rest as Non-Melanoma Cancer. The negative class is kept as it is. This is to ensure we keep our model architectures simple and limited to identifying Melanoma and Non-Melanoma Cancer. This also reduces the variety and diversity generated by the textual features combined with the images. All the Images are sized 450x600 Pixels as per HAM10000 Dataset.

After this we apply Singular Value Decomposition (SVD) on all images one by one along the Y-Dimension and reconstruct the images using Top-K Singular Values and their corresponding singular vectors in (U) and (V^T), truncating all vectors and values below K. The K-varies from 100, 200, 300, 350, 400,450.The initial step size is big (100) and as we get closer to half of the vertical axis(600) i.e. 300 we decrease step size by half(50).We Reconstruct the entire datasets at all K values. After this, we need to select the best k-value such that we get a good compression ratio but it also preserves the image characteristics and has minimal loss. Then, we study various metrics such as Peak-Signal-to-Noise Ratio, Structural Similarity, Histogram Intersection and Bhattacharya distance between Color Histograms, Euclidean Distance between Hu Moments and Haralick Textures, percentage of pixel values changed more than a threshold, Variety of Information and Normalized Mutual Information, the Mean Squared Error Between High and Low Frequency Components, Spectral Loss Coefficient (for frequency analysis) of Original and Reconstructed Data. We also analyse the mean values and standard deviations for the metrics across different values of K for reconstructed dataset. We select the optimal value of K and use it for training our Models such that Compression ratio is good, Essential Features are preserved and storage size reduction is maximum.

Models Trained on HAM10000 Have resized the input images to reduced dimensions for reducing latency and reduce memory consumption.[6] Hence, we also consider some reduced standard dimension sizes like (256,256) and also try reducing it down by one axis- across y-axis (450,150) , across x axis (300,600), across both axis - (200,100), (300,300). The selection of Resolution of Pixel size is random but less pixels mean less information resulting in less feature maps in convolution and faster training times and reduced inference times. We Compare the above mentioned metrics after resizing our reconstructed dataset to select the best performing resolution with optimal metrics. We also study the effect of how Image Size affects the number of learnable parameters of the model since less resolution means less number of feature maps passed onto subsequent layers.

Next we start pre-processing the dataset, applying normalization and scaling to Numeric Attribute Age and Label Encoding to Sex and Localization. We fill Null Values in Age using Median of same column since it is a right-skewed distribution in the dataset. We also apply data augmentation to balance the dataset and enhancing generalization ability and robustness of the task.

We have already designed 4 Custom Neural Architectures on original Dataset with benchmark accuracy and precision in training, testing and validation stages. The 4 architectures are - a) Lenet-5 based Neural Network, b) Alexnet Based Neural Network, c) Custom Neural Network - 1 d) Custom Neural Network - 2(utilizing Residual Skip Connections). The Key Concepts we use here is the Text Model a Branch of Neural Network that processes the Metadata features, and the Concatenation of Final Features by Combining Image and Text Model Outputs for making a final Diagnosis.

Table 3.1: Sample Architecture of the Lenet-5 Based Neural Network We designed for Skin Cancer Prediction

Layer	Type	Filters/Units	Kernel Size
Text (MLP) Model			
1	Dense	4	-
2	Dropout	-	-
3	Dense	6	-
4	Dense	2	-
5	Dropout	-	-
6	Dense	1	-
Image (CNN) Model			
1	Conv2D	6	5x5
2	MaxPooling2D	-	2x2
3	BatchNormalization	-	-
4	Conv2D	16	5x5
5	MaxPooling2D	-	2x2
6	BatchNormalization	-	-
7	Flatten	-	-
8	Dense	84	-
9	Dense	120	-
Combined Model			
1	Concatenate	-	-
2	Dense	16	-
3	Dense	4	-

Next in the research study, we train the same model using the reconstructed image dataset and observe the testing , training and validation metrics. We compare these with baseline model and observe the results if model is able to capture features from reconstructed data and achieve similar baseline performance metrics. We train the same model on resized images where resolution is selected above after analysis to reduce memory consumption and reduce amount of learnable parameters. The benchmarks for Lenet-5 based Neural Network in validation metrics is 75% and for other models is 80%. For Training it remains 90% for all Models. For Testing Metric Comparison , we select the model with values closest to baseline metrics with accepted error range of -2%.

After this we select a pruning factor as 50%, we reduce the number of convolution filters and dense units in subsequent fully connected layers by this factor in an iterative way till we achieve a model under-performing in training, validation and testing metrics. Then, we make the pruning factor 33.33 %, repeating the same procedure until a minimal model is achieved. If still facing issues, reduce pruning factor by half in every iteration. 50% and 33% are selected as staring pruning factors since they cause exponential reduction in learnable parameters of model in iterative iterations and polynomial reduction if only some convolutional filters and dense layers are reduced. Training Precision and Accuracy, Validation Accuracy and Precision, Testing Metrics like accuracy, recall and precision are compared to the baseline model metrics to select the optimal model minimizing model size and maximizing performance.

3.3. Block Diagram

The Block Diagram for the proposed methodology in this research study is show below in two parts. One deals with Data Preprocessing and flow of how to create the reconstructed dataset using SVD. Second Part deals with how models are trained on reconstructed dataset and pruned further.

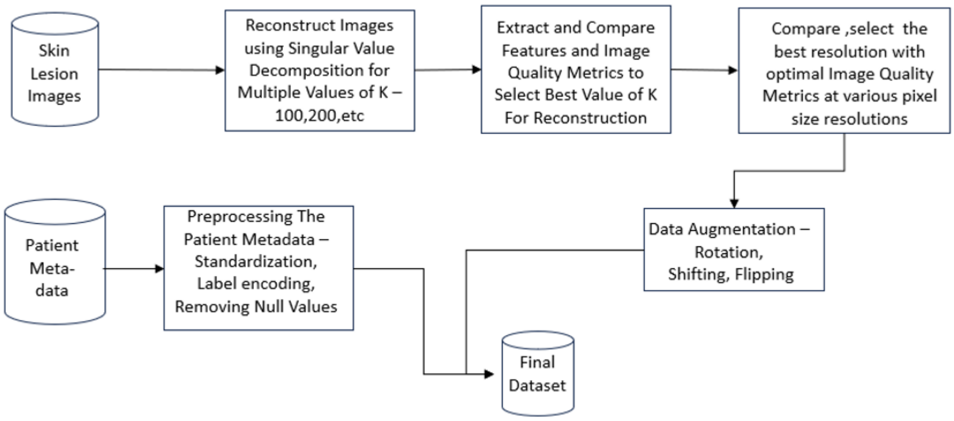

Skin Lesion Images
Reconstruct Images using Singular Value Decomposition for Multiple Values of K – 100,200,etc
Extract and Compare Features and Image Quality Metrics to Select Best Value of K For Reconstruction
Compare ,select the best resolution with optimal Image Quality Metrics at various pixel size resolutions
Patient Meta-data
Preprocessing The Patient Metadata – Standardization, Label encoding, Removing Null Values
Data Augmentation – Rotation, Shifting, Flipping
Final Dataset

Figure 3.3: Block Diagram for Preprocessing , Augmentation and Generating the Reconstructing Dataset using SVD

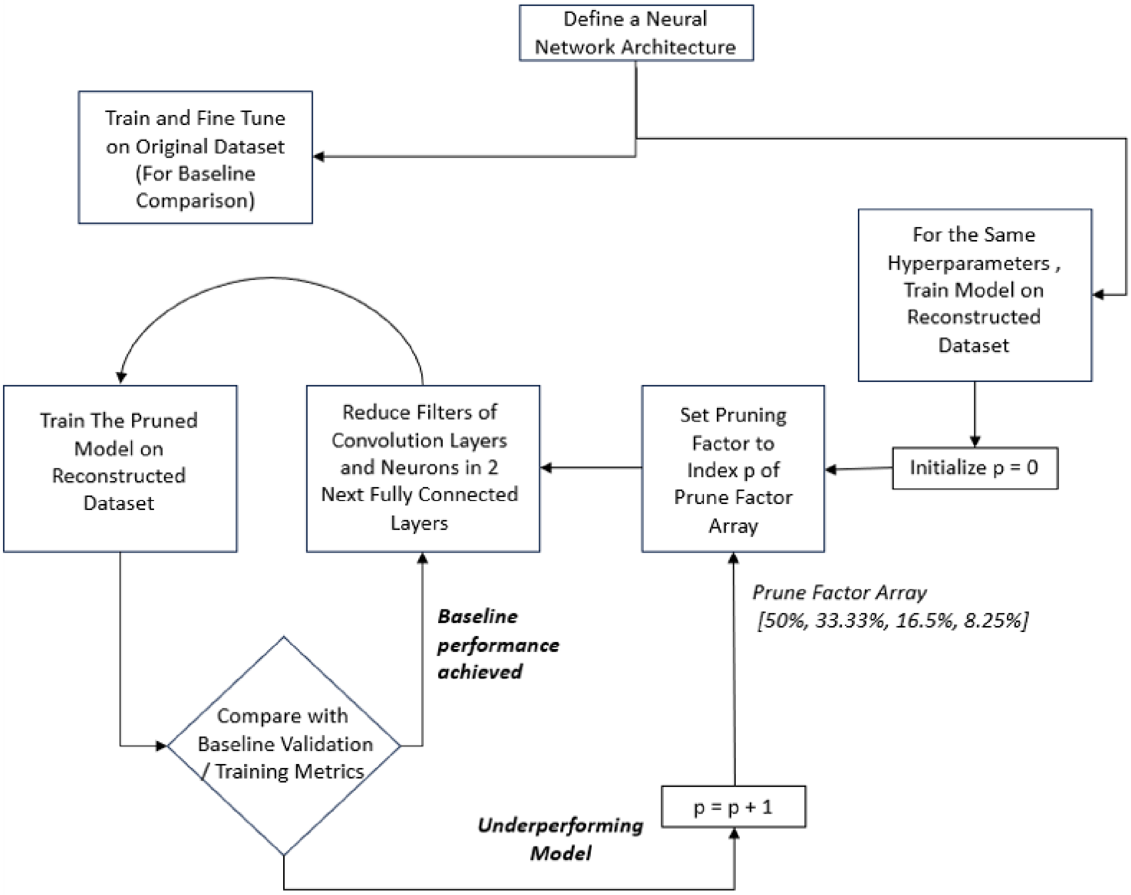

Define a Neural Network Architecture
Train and Fine Tune on Original Dataset (For Baseline Comparison)
For the Same Hyperparameters , Train Model on Reconstructed Dataset
Train The Pruned Model on Reconstructed Dataset
Reduce Filters of Convolution Layers and Neurons in 2 Next Fully Connected Layers
Set Pruning Factor to Index p of Prune Factor Array
Initialize p = 0
Prune Factor Array [50%, 33.33%, 16.5%, 8.25%]
Baseline performance achieved
Compare with Baseline Validation / Training Metrics
p = p + 1
Underperforming Model

Figure 3.4: Block Diagram for Training of Custom Neural Networks and their Model Pruning

3.4. Tools Used

3.4.1. Setup & Installation for Django Application

Prerequisites

- Python 3.8+
- Django 3.x
- Pip

Installation Steps

1. **Navigate to the Project Directory**

   ```
   cd project_name
   ```

2. **Create and Activate Virtual Environment**

   ```
   python –m venv env
   env\Scripts\activate.bat
   ```

3. **Install Dependencies**

   ```
   pip install −r requirements.txt
   ```

4. **Apply Migrations**

   ```
   python manage.py migrate
   ```

5. **Run the Development Server**

   ```
   python manage.py runserver
   ```

3.4.2. Technologies Used For Django Application

Django

Django is a high-level web framework for Python that facilitates efficient development and clear, practical design. It was chosen for its scalability, resilience, and the availability of the following features:

- **MTV Architecture** Django follows the Model Template View architecture, which encourages a distinct division between the user interface and functional code.
- **Built-in Admin Interface** The framework includes a robust admin interface that simplifies management during the initial phases of application and data development.
- **Security** Django provides built-in defenses against common security risks such as SQL injection and cross-site scripting.
- **Community Support & Documentation** Django has a sizable, vibrant community and comprehensive support, making it easier to find solutions and resolve issues.

SQLite

SQLite is a lightweight disk-based database system used for storing and managing application data. It is chosen for the following reasons:

- **Zero Configuration** SQLite is easy to integrate and use because it does not require setup or management.
- **Serverless** SQLite operates as a serverless database, simplifying deployment by directly reading and writing data to standard disk files.
- **Transactional** SQLite transactions ensure data integrity by fully complying with ACID (Atomicity, Consistency, Isolation, Durability) properties.
- **Lightweight** The SQLite library is compact, with a total size of less than 500KB, making it suitable for applications that require a simple, low-cost database solution.

3.4.3. Setup & Installation for Streamlit Application

To run the application locally, follow these steps

1. **Clone the repository**

```
cd s-kare_chatbot
```

2. **Install requirements**

```
pip install -r requirements.txt
```

3. **Run the Streamlit application**

```
streamlit run app.py --server.enableXsrfProtection false
```

3.4.4. Technologies Used For Streamlit Application

The application utilizes various libraries and technologies to deliver its functionality:

Libraries Used

- **Streamlit** For the web interface and user interaction.
- **PIL (Pillow)** For image processing.
- **spacy** For natural language understanding and processing.
- **nltk** For tokenization and stemming.
- **PorterStemmer** For word stemming.
- **tf (TensorFlow)** For machine learning model integration.
- **torch (PyTorch)** For deep learning model handling.
- **FPDF** For creating a PDF of the report generated by the ML algorithm.

Image Analysis

- **numpy** For numerical operations.
- **PIL (Pillow)** For handling image files.
- **glob** For file pattern matching.
- **pandas** For data manipulation and analysis.

Speech Recognition

- **speech_recognition** For converting speech to text.

3.5. Advantages & Disadvantages

3.5.1. Advantages & Disadvantages of Algorithm

Storage Optimization

The Storage Optimization using SVD significantly compresses data while capturing most of the variance and discriminatory features of the images. This reduces storage requirements on any platform significantly and also be performed at scale without much loss in information. There is also another observation here, that reconstructed images now have psycho-visual and inter-pixel redundancy introduced so comparatively less number of convolution filters can identify the discriminatory patterns.The disadvantage is the increased overhead time and cost of analysis of what number of singular values needed for reconstruction so there is no compromise in image quality and essential features are preserved.

Reduction of Model Size

Model Pruning reduces number of hidden neurons using a pruning factor for convolution layers and the next 2 fully connected layers. This helps in reduction of model size, reduction in number of filters, number of learnable parameters and faster training and inference times. This also makes it easy to deploy the model on small scale devices. The disadvantage here is this is a quite simple, uninformed and naive approach on model pruning and requires lots of iterations in fine-tuning it.

3.5.2. Advantages & Disadvantages of Delivery Mechanism

Based of the idea of making healthcare accessible to the rural and remote places of the region, the algorithm is made available to the public using Chatbot application along with kiosk and connect these centers with a central hospital to increase the efficiency of medical care. This would lead to the following advantages:

Enhancing Efficiency and Coverage

Particularly in underserved and rural regions, combining the use of medical kiosks with the healthcare workforce can improve efficiency and coverage. The accessibility of medical services can be increased by strategically placing kiosks in areas where access to healthcare institutions is constrained. They could function as a hub-and-spoke system, with each kiosk acting as a satellite hub connected to a central healthcare facility. This setup facilitates more effective utilization of available resources and enhances the flow of patient management. Technologies such as electronic health records (EHRs), telemedicine systems, and AI-based tools for diagnosis can be integrated into kiosks to improve efficiency and reduce waiting times for patient consultations and follow-ups. These technologies enable kiosks and healthcare practitioners to communicate and share data in real-time, ensuring continuity of care and reducing the risk of medical errors. This integrated approach can significantly enhance healthcare access and quality in India by expanding coverage and improving efficiency.

Connecting Patients with Dermatologists

Medical kiosks equipped with telemedicine technology facilitate remote consultations between patients and dermatologists. This is particularly beneficial in areas with a shortage of specialized healthcare providers. Patients can have their skin lesions examined remotely by a dermatologist, who can then provide a diagnosis and treatment plan. This connectivity ensures that patients receive timely and expert medical advice without the need for long-distance travel, making specialist care more accessible.

Follow-Up Care and Monitoring

Follow-up care and ongoing monitoring are essential components of effective skin cancer management. Medical kiosks can support these processes by allowing patients to have regular check-ups and consultations through telemedicine. Dermatologists can review the progress of treatment, monitor for any new or recurring lesions, and adjust treatment plans as necessary. This continuous care model helps in early detection of recurrences and ensures that patients adhere to their treatment regimens.

Disadvantages Kiosk Dependencies

As the success rate of the application significantly depends upon the utilization of the application developed in the project as well as the collaboration with Kiosk and Central Hospitals along with the medical staff for assistance in healthcare ,this would lead to the following disadvantages:

Operational and Technical Difficulties

Due to their heavy reliance on technology, medical kiosks are vulnerable to hardware failures, software flaws, and malfunctions. Finding prompt solutions for these issues can be challenging in impoverished and rural locations where technical support may be lacking. Kiosk malfunctions can lead to delays in diagnosis and treatment by disrupting screening and consultation processes. Regular maintenance is essential to keep kiosks accurate and functional, but providing efficient maintenance in remote areas can be challenging, potentially leading to inaccurate diagnoses if equipment calibration and upkeep are neglected.

Quality of Care and Diagnosis Accuracy

Limited Diagnostic Capability

While medical kiosks can perform preliminary screenings, they may lack the sophistication and comprehensive diagnostic capabilities of traditional healthcare facilities. This can result in false positives or negatives, causing unnecessary anxiety for patients or delays in necessary treatment.

Dependence on Non-Specialized Staff

Kiosk operators, although trained, are not healthcare professionals. The quality of initial screenings and patient interactions may vary based on the operator's experience and training. Misinterpretation of symptoms or improper use of technology could affect the accuracy of preliminary assessments.

Data Security and Privacy Concerns

Risks to Data Privacy

Medical kiosks gather and transmit sensitive health information, making it crucial to ensure the confidentiality and integrity of this data, especially in regions with inadequate cybersecurity infrastructure. Unauthorized access or breaches of patient data could seriously compromise patient confidentiality and erode trust in the system.

Regulation Compliance

Staying compliant with health data protection regulations, such as the DPDP in India, can be challenging, particularly in areas where enforcement of laws may be lax. Non-compliance could lead to legal repercussions and undermine patient and community confidence in the kiosk's operations.

4. ANALYSIS AND DESIGN

4.1. Experimental Design

Table 4.1: Architecture of the Lenet-5 Based Neural Network

Layer	Type	Filters/Units	Kernel Size
Text (MLP) Model			
1	Dense	4	-
2	Dense	6	-
3	Dense	2	-
4	Dense	1	-
Image (CNN) Model			
1	Conv2D	6	5x5
2	Conv2D	16	5x5
3	Flatten	-	-
4	Dense	84	-
5	Dense	120	-
Combined Model			
1	Concatenate	-	-
2	Dense	16	-
3	Dense	4	-

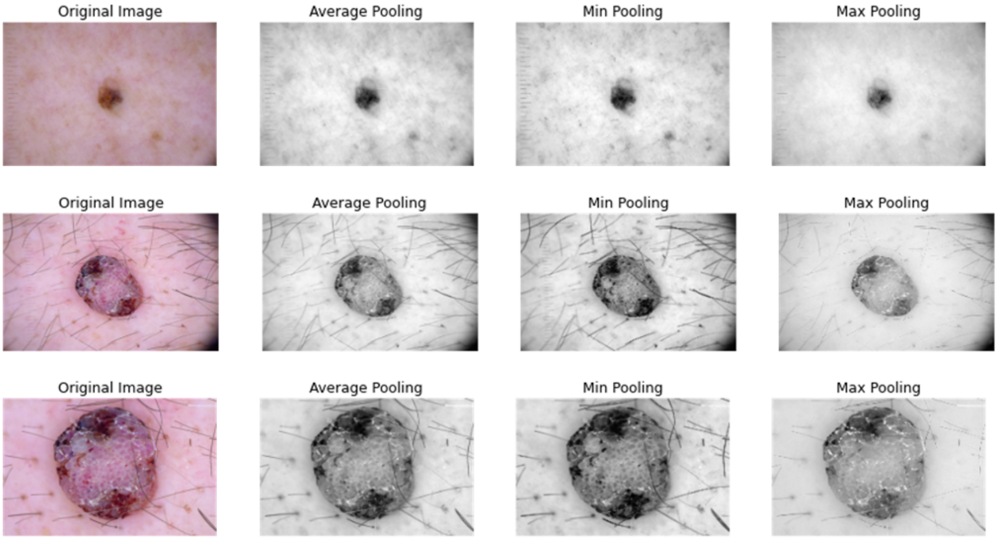


Figure 4.1: Pooling Functions Effect on 3 Images selected from the Dataset

Table 4.2: Architecture of the AlexNet-Based Neural Network

Layer	Type	Filters/Units	Kernel Size	Strides
Text (MLP) Model				
1	Dense	4	-	-
2	Dense	6	-	-
3	Dense	2	-	-
4	Dense	1	-	-
Image (CNN) Model				
1	Conv2D	64	5x5	2x2
2	ZeroPadding2D	-	2x2	-
3	Conv2D	192	5x5	1x1
4	ZeroPadding2D	-	1x1	-
5	Conv2D	384	3x3	-
6	ZeroPadding2D	-	1x1	-
7	Conv2D	384	3x3	-
8	ZeroPadding2D	-	1x1	-
9	Conv2D	256	3x3	-
10	Flatten	-	-	-
11	Dense	1024	-	-
12	Dense	1024	-	-
13	Dense	512	-	-
Combined Model				
1	Concatenate	-	-	-
2	Dense	8	-	-
3	Dense	4	-	-

Custom Neural Network-1 - The standard CNN model in our mixed neural network framework focuses on a straightforward, deep convolutional approach for processing image data. It begins with convolutional layers that capture the intricacies of the input images using progressively increasing filters: starting with 16 filters at a 7×7 kernel, moving to 48 filters at 5×5, and further layers with 92, 192, 192, and 256 filters each with a 3×3 kernel. Each convolutional layer is followed by max-pooling layers to downsample the feature maps, batch normalization to maintain stability and enhance training speed, and dropout to prevent overfitting by randomly deactivating neurons during training. These layers work together to create detailed feature maps from the input images. The CNN then uses fully connected layers with 1024, 1024, and 512 units to further refine and map the learned features into a compact form suitable for classification.

Custom Neural Network-2 - The ResNet-based CNN model for our mixed neural network leverages the power of deep residual learning to process image data efficiently. This architecture starts with a sequence of convolutional layers where the filters increase in size progressively: 16 filters with a 5×5 kernel, 32 filters with a 7×7 kernel, followed by layers with 64, 96, and 192 filters each using a 3×3 kernel. Each convolutional layer is paired with max-pooling to reduce the spatial dimensions, batch normalization to stabilize and accelerate training, and dropout to mitigate overfitting. This design extracts multi-scale feature hierarchies from the images. The model incorporates ResNet blocks—specifically, with 192 and 256 filters—that include skip connections to preserve the flow of gradients, thus enabling deeper networks to be trained without suffering from vanishing gradients

We begin by defining the architectures of our 4 Custom Neural Networks and the design choices in the common elements. The elements and building blocks of our custom defined neural networks have been described below. We can also view the number of filters, number of neurons , kernel size for two of the architectures in the Figures 4.1 and 4.2. The Other two architectures have been described due to their deep network architecture.

1. Two Separate Branches for Text and Image Processing: We create two separate model - Image and Text that learns features from respective input since patient metadata is also important in providing a nuanced and personalized diagnosis along with the Skin Lesions

2. Max pooling - We use MaxPooling for dimensionality reduction and to learn the most prominent features such as fine textures, edges which have high activation. It also helps in reduction of noise and smoothens out the surrounding lesion area since there are not many important and prominent edges outside of lesion boundary. An Example can be visible in 4.1

3. Concatenation - We Concatenate the Feature Maps Extracted from both the Models and combine them into a single feature map for subsequent Layers to Learn the patterns from

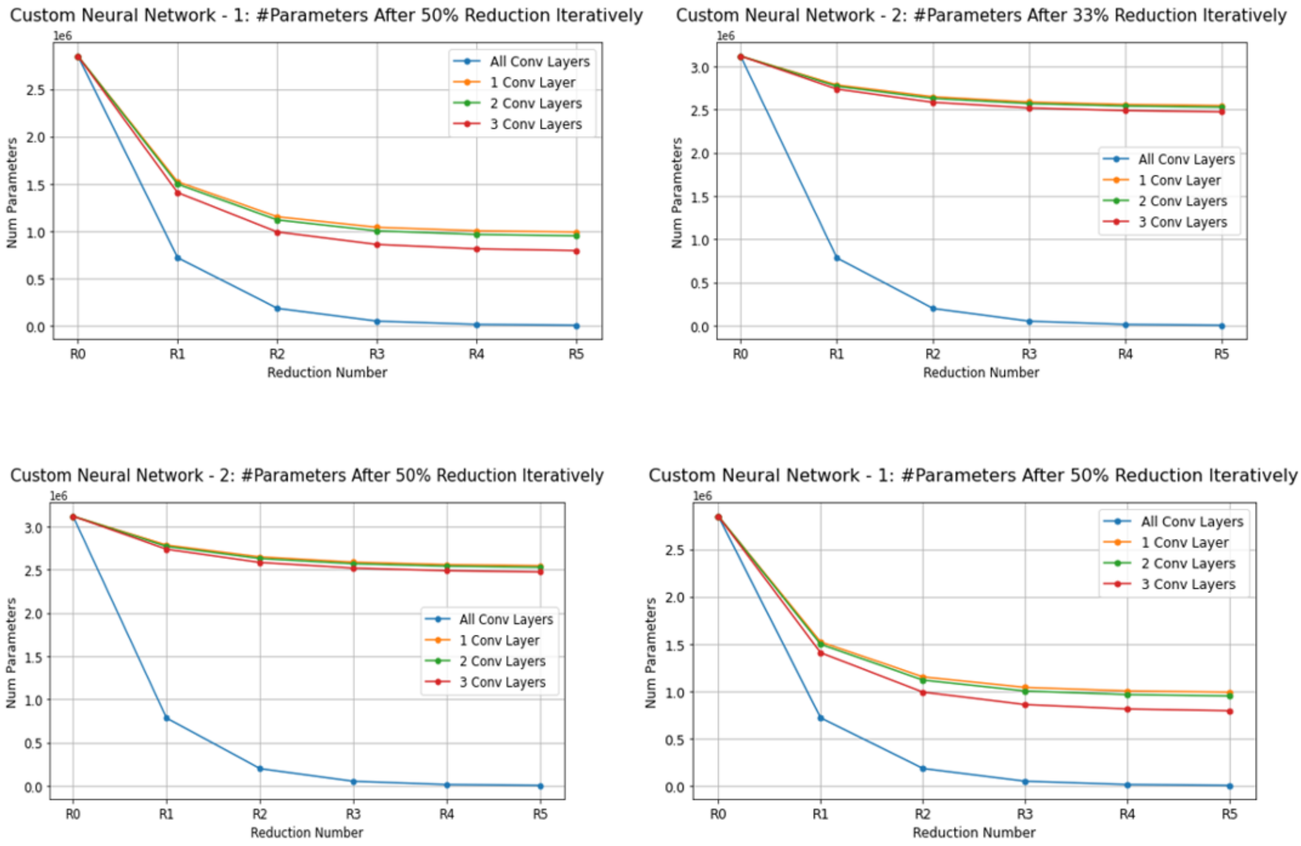
both of them before prediction.

4. Activation Functions : We use ReLU Activation Function as it is simple and computationally efficient. It also introduces a non-linearity in features and patterns.
5. Residual Blocks - Residual blocks, introduced in ResNet architectures, address the vanishing gradient problem by providing shortcut connections that bypass one or more layers. This design allows gradients to flow directly through these shortcuts, ensuring that deeper networks can be trained effectively.
6. Alexnet and Lenet-5 Architectures - We use them in two of our architectures and in the CNN Image Model as they are established and standard CNN Architectures that have performed well on Skin Cancer Dataset.

We then preprocess our Original Dataset, applying standardisation on age attribute, One-Hot encoding for class labels, Label encoding for sex and localization. We then Split the Dataset into training and validation, where 3300 samples are for training and 1100 samples are for validation. We also use a testing dataset of 360 Images. In each split, there is equal distribution of the classes.

The Models are trained on this data achieving good performance. The baseline benchmarks are setup for training and validation here, 90% in training and 80% in validation. Only for Lenet-5, we create 75% validation benchmark since the model depth is low and only 2 convolution layers are present with less neurons. All Architectures are trained for 75 Epochs , with a learning rate of 0.002 and Adaptive Moment Estimation Optimizer. The best performing model achieving baseline performance are saved at Custom Checkpoints.

We now study the effect of reducing filters and neurons in next 2 Fully connected Layers by the pruning factor iteratively on all our architectures. We first reduce all convolution filters in all layers, then of only first Convolution layers, then first two layers and then first 3 convolution layers. We reduce the Neurons in Fully Connected Layers in All cases .We Model the Graph using a Linear Regression, a Polynomial and exponential Model to study the coefficient of determination Value. As it can be observed in graphs of 2 architectures from 4.2, and R^2 is maximum and greater than 0.9 for Exponential Model when we reduce all convolution layer's number of filters iteratively and in polynomial time in other cases. Thus, reducing maximum convolution filters is desirable.


Custom Neural Network - 1: #Parameters After 50% Reduction Iteratively

Custom Neural Network - 2: #Parameters After 33% Reduction Iteratively

Custom Neural Network - 2: #Parameters After 50% Reduction Iteratively

Custom Neural Network - 1: #Parameters After 50% Reduction Iteratively

Figure 4.2: Number of Learnable Parameters (in millions) vs Reduction Iteration at different pruning factors

Now, First we train and compare the same architecture on reconstructed dataset and measure its performance with baseline metrics. Then, we apply Model Pruning using numerous pruning factors from Pruning factor array [50%, 33.33%, 16.5%, 8.25%] till we achieve a minimal model without much performance degradation. The Pruning Factor reduces filter and neurons in following fashion.

$$N_{\text{new}} = \lfloor N_{\text{old}} \times (1 - P[p]) \rfloor$$

where:

- N_{new} is the number of filters or neurons after pruning.
- N_{old} is the original number of filters or neurons before pruning.
- P is the Pruning Factor, expressed as a percentage. It represents the proportion of filters or neurons to be retained after pruning.

- $\lfloor \cdot \rfloor$ denotes the floor function, which rounds down to the nearest integer.

4.1.1. Algorithm

Algorithm 1 Image Compression using SVD and Quality Metrics Evaluation

```
Input: List of images $I$, SVD compression levels $K = \{100, 200, 300, 350, 400, 450\}$
Output: Best compression level $k^*$ based on quality metrics
for each image $I$ in $I$ do
    Initialize empty lists for PSNR, SSIM, RMSE,
    Initialize empty lists for Color, Hu Moments, Texture Features
    for each $k \in K$ do
        Compute SVD of image $I$: $U, \Sigma, V^T \leftarrow \mathrm{SVD}(I)$
        Retain top $k$ singular values in $\Sigma$: $\Sigma_k$
        Compute compressed image $\hat{I}_k = U \cdot \Sigma_k \cdot V^T$
        Compute quality metrics:
        Calculate PSNR, SSIM, RMSE for $\hat{I}_k$ compared to $I$
        Extract Color, Hu Moments, Texture Features from $\hat{I}_k$
        Extract Color, Hu Moments, Texture Features from I
        Compare Color, Hu Moments, Texture Features from I with $\hat{I}_k$
        Append quality metrics and features to respective lists
    end for
    Determine $k^*$ based on best performance across metrics
end for
return Best compression level $k^*$
```

Training Benchmark = 90% , Testing - 2% of Error Accepted , Validation = 80%

For Lenet-5 based Neural Network, Validation Baseline is set to 75% due to less number of convolutions and less depth of network. All of the Models are Trained on GPUP100 Available via Kaggle API.

Algorithm 2 Iterative Model Training with SVD-Reconstructed Data and Pruning

```
Input: Reconstructed data $\hat{X}_k$ for a specific $k$, Pruning factors $P = \{50\%, 33.33\%, 16.5\%, 8.25\%\}$,
  Baseline model metrics, Training, Validation, and Testing data
Output: Minimal pruned model achieving desired metrics
Initialize best model metrics
Initialize variable $p$ as index for pruning factors
$baseline\_met \leftarrow$ false
$baseline\_not\_met\_counter \leftarrow 0$
while true do
    Train model on $\hat{X}_k$ using training data
    Validate model on validation data
    Calculate precision and accuracy metrics
    if metrics meet baseline and testing error acceptance up to 2% then
        Save model and metrics as best model
        return Best model and metrics
    else
        $baseline\_met \leftarrow$ false
        break
    end if
    $p \leftarrow 0$
    while true do
        if $p \geq$ length($P$) then
            break
        end if
        Prune Convolution Layers, and next 2 Fully Connected Layers of model with factor $P[p]$
        Update number of filters/neurons using
            $$N_{\text{new}} = \lfloor N_{\text{old}} \times (1 - P[p]) \rfloor$$
        Retrain pruned model on training data
        Validate pruned model on validation data
        Calculate precision and accuracy metrics
        if metrics meet baseline and testing error acceptance up to 2% then
            Save model and metrics as best model
            $baseline\_met \leftarrow$ true
            $baseline\_not\_met\_counter \leftarrow 0$
        else
            $baseline\_met \leftarrow$ false
            $baseline\_not\_met\_counter \leftarrow baseline\_not\_met\_counter + 1$
            if $baseline\_not\_met\_counter \geq 2$ then
                break
            end if
            Increment $p$ by 1
        end if
    end while
end while
```

4.2. Feature Extraction and Analysis

This section deals with how we select the best k value for Singular Value Decomposition after comparing Multiple Image Quality Metrics and Minimizing Error between Features, and maintaining a significant compression ratio.

Peak Signal-to-Noise Ratio (PSNR) PSNR is a metric used to quantify the quality of an image reconstruction by measuring the ratio between the maximum possible power of a signal and the power of corrupting noise that affects the fidelity of its representation. It is calculated using the formula:

$$\mathrm{PSNR} = 10 \cdot \log_{10}\left(\frac{\mathrm{MAX}^2}{\mathrm{MSE}}\right)$$

where MAX is the maximum possible pixel value (typically 255 for 8-bit images), and MSE (Mean Squared Error) represents the average squared difference between the original and reconstructed images. A higher PSNR value indicates better reconstruction quality, suggesting that the reconstructed image closely matches the original without significant distortion. Generally a PSNR Value greater than 30dB is considered good.

Mean Squared Error (MSE) MSE measures the average squared difference between corresponding pixels of the original and reconstructed images. It is calculated as:

$$\mathrm{MSE} = \frac{1}{mn}\sum_{i=1}^{m}\sum_{j=1}^{n}[I(i,j) - \hat{I}(i,j)]^2$$

where $I(i,j)$ and $\hat{I}(i,j)$ are the pixel values of the original and reconstructed images, respectively, and $m \times n$ is the image resolution. A lower MSE indicates better image fidelity, as it reflects less distortion between the original and reconstructed images.

Structural Similarity Index (SSIM) SSIM is a perceptual metric that quantifies the similarity between two images by comparing their luminance, contrast, and structure. It measures how closely the reconstructed image resembles the original in terms of these perceptual factors. SSIM is calculated using local image patches and is defined as:

$$\mathrm{SSIM}(x,y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}$$

where μ_x, μ_y are the means of x and y, σ_x^2, σ_y^2 are the variances, σ_{xy} is the covariance, and C_1 and C_2 are constants to stabilize the division with weak denominator. SSIM ranges between -1 and 1, where 1 indicates perfect similarity.

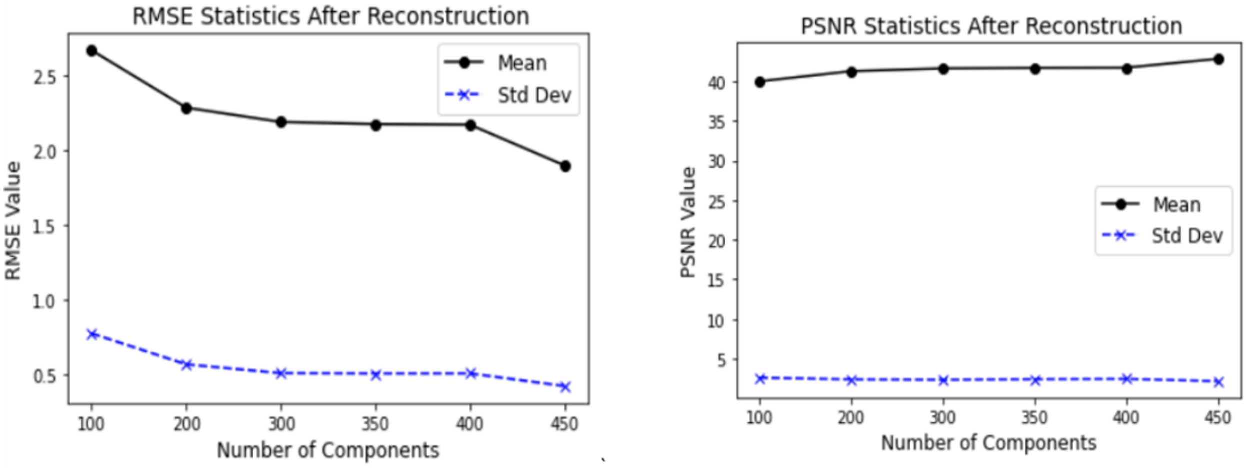


Figure 4.3: Image Quality Metrics Comparison after reconstruction for various values of K

In image quality assessment, PSNR provides a straightforward measure of fidelity based on pixel-wise differences, MSE quantifies the average error, and SSIM offers a perceptual evaluation considering structural information. Together, these metrics provide comprehensive insights into the accuracy and perceptual quality of reconstructed images, essential for various applications such as medical imaging, surveillance, and digital photography.

We observe Image Quality Metrics like PSNR, SSIM and RMSE of reconstructed dataset compared to original at various values of K in Figure 4.3 and observe for different values of K on x-axis, depicted as number of components used in reconstruction. As we increase number of components, we observe a better values of RMSE between the pixels and slightly better PSNR Values around 40 dB. The SSIM Value improves from 0.97 to 0.99 improving at a very slow rate as we improve number of singular vectors and values. The standard deviation for these metrics also follows the same trend.

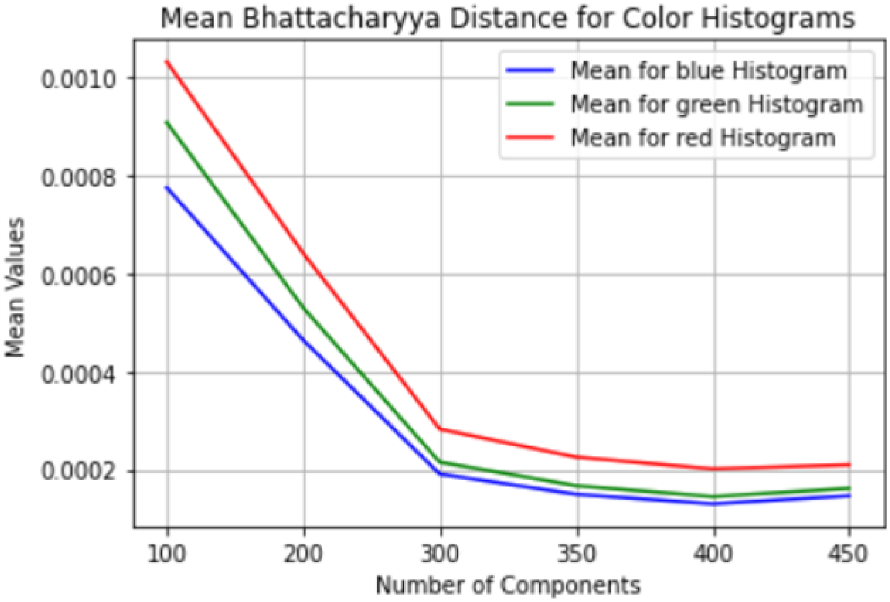


Figure 4.4: Comparing Average Bhattacharya Distance Between Color Histograms at different k-singular values

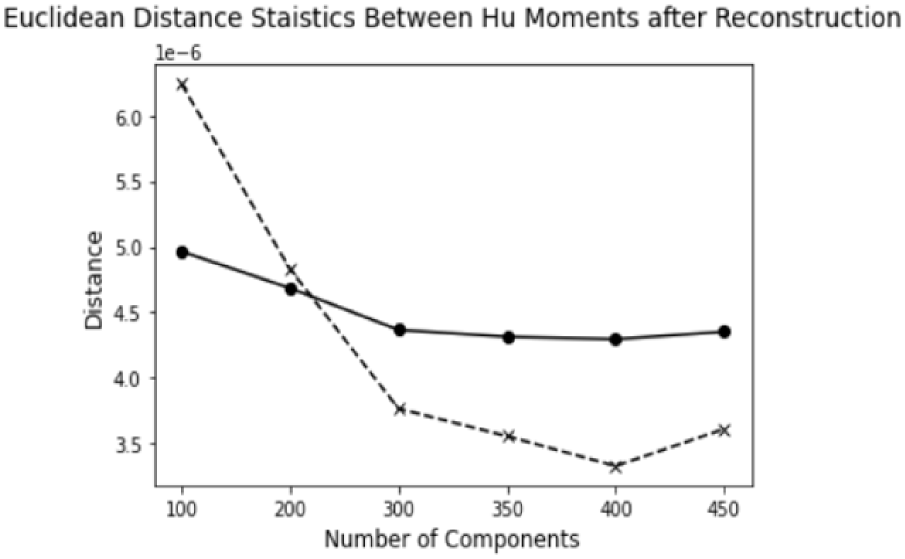


Figure 4.5: Comparing Euclidean Distance between Hu Moments for reconstructed dataset at different k-singular values a)Dotted Line represents the Avg. std deviation and Solid Line Average Mean

The Color, Shape and Texture are the most essential features of the skin lesions, so we analyze how these features are changed and compare them with features from original images , to see how much essential information we preserve and how much we lose.

Bhattacharya Distance- This is a statistical measure used to determine the similarity between two probability distributions. In the context of image processing, it is used to compare color histograms, which represent the distribution of colors in an image. A smaller

Bhattacharya distance indicates a higher similarity between the original and reconstructed images in terms of their color distribution.

In Figure 4.4,we calculate the Bhattacharya distance between the original and reconstructed dataset images' color histograms in the three channels. This helps us understand how well the color information is preserved during reconstruction. The Bhattacharya distance measures the similarity between two probability distributions, making it an effective tool for comparing color histograms. As we increase the number of singular values (SVD components), the distance minimizes, indicating improved color preservation, and stabilizes after k=350, suggesting that using 350 singular values captures the essential color information effectively

Hu Moments- Hu Moments are a set of seven invariant derived from the central moments of an image, which capture the geometric properties or shape information of objects within the image. These moments remain unchanged under image transformations such as translation, scaling, and rotation, making them highly effective for shape analysis.

In Figure 4.5, we compute the Euclidean distance between the Hu Moments of the original and SVD-reconstructed images to assess how well the shape information is preserved. We observe that the average Euclidean distance remains stable, but the standard deviation decreases until $k = 400$ and then slightly increases.This suggests that the shape information is preserved at $k = 350$ onwards due to stability in std deviation, balancing between retaining essential components and minimizing the number of singular values used for reconstruction. The correlation coefficient between the hu moments at all components is 0.999 ensuring high degree of similarity in shape between reconstructed and original dataset.

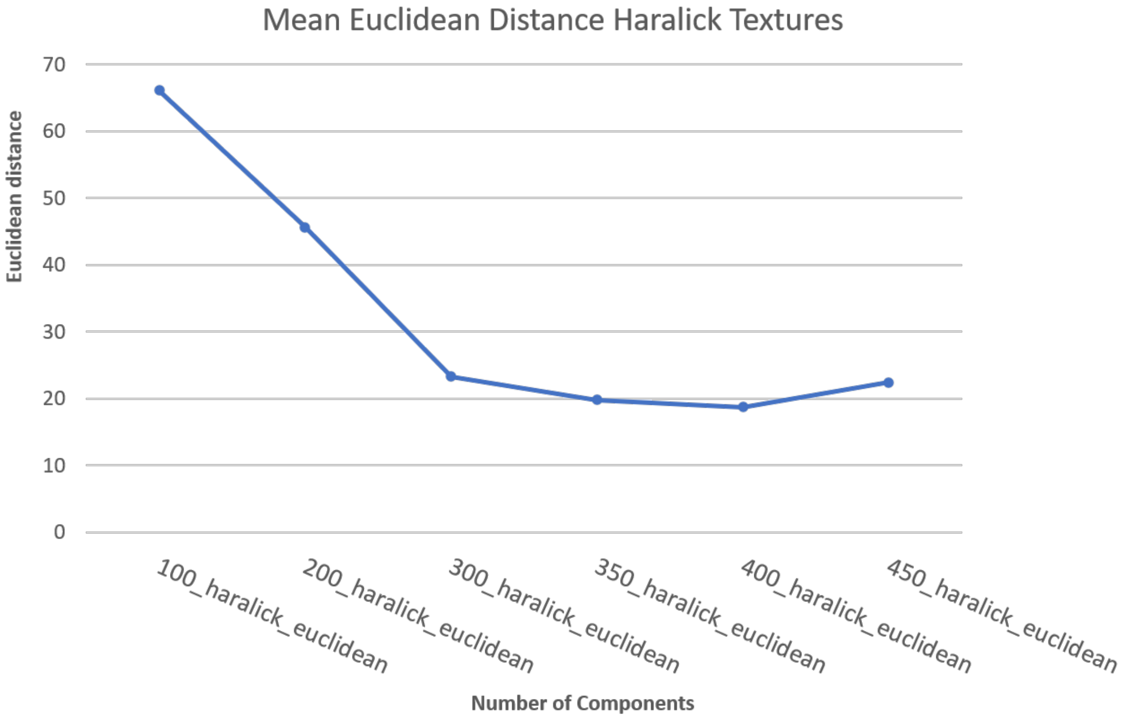


Figure 4.6: Comparing Euclidean Distance Between Haralick Texture Features at different k-singular values

Haralick Texture Features are a set of statistical measures derived from the Gray-Level Co-occurrence Matrix (GLCM) of an image, which quantifies the spatial relationships between pixel intensities. The GLCM calculates how often pairs of pixel with specific values and in a specified spatial relationship occur in an image, capturing the texture properties by examining the frequency and distribution of pixel pairs. Haralick features computed from this matrix include metrics such as contrast, correlation, energy, and homogeneity, which describe the texture's smoothness, coarseness, and regularity.[14]

In our study, we observe the Euclidean distance between the Haralick texture features of the original and reconstructed images in Figure 4.6. This distance helps us understand how well the texture is preserved during the reconstruction process. We find that as the number of singular values k increases, the Euclidean distance decreases, indicating that the texture of the reconstructed image becomes more similar to the original. This trend stabilizes at $k = 350$, suggesting an optimal balance between data reduction and texture preservation. However, for $k > 400$, we notice an increase in the Euclidean distance, likely due to noise introduced by the smaller singular values, which were previously truncated. Texture features, especially in sensitive areas like medical imaging of skin lesions, are crucial for accurate analysis. Even slight noise can significantly affect these metrics, as texture in skin lesions includes fine details that are essential for diagnosis. Therefore, maintaining the fidelity of these texture features without introducing noise is critical for the effectiveness of the reconstructed images.

Frequency Domain Analysis In image processing using methods such as the Fourier Transform, an image requires being transformed from the spatial domain to the frequency domain in order to be analysed in the frequency domain for image processing. We can examine the frequency components of an image by transforming it so that high frequencies capture the small details and low frequencies represent the broad structures. We can assess how well various frequency components are maintained during reconstruction by comparing the frequency domains of the original and rebuilt images.

In Figure 4.7, we observe the spectral correlation coefficient between the original and reconstructed images, which remains consistently high at 0.99 across all components. This high spectral correlation indicates that the reconstructed images retain their frequency characteristics almost perfectly, ensuring that both low and high-frequency details are well-preserved.

Furthermore, we examine the Mean Squared Error (MSE) between the high-frequency coefficients and the low-frequency coefficients of the original and reconstructed images. As the number of singular values k increases, we find that the MSE between these coefficients decreases. This trend suggests that increasing the number of components enhances the reconstruction's accuracy, especially in capturing the essential frequency details without losing fine details or introducing artifacts. The MSE also stabilizes at $k = 350$ as observed in Figure 4.7

From these observations, we infer that the reconstruction process efficiently preserves the image's frequency characteristics, crucial for maintaining the visual and structural integrity of images. In applications like medical imaging, particularly in analyzing skin lesions, preserving high-frequency details is vital as they often contain critical diagnostic information. Thus, selecting an optimal number of singular components, around $k = 350$, ensures that we retain these important details while minimizing unnecessary noise.

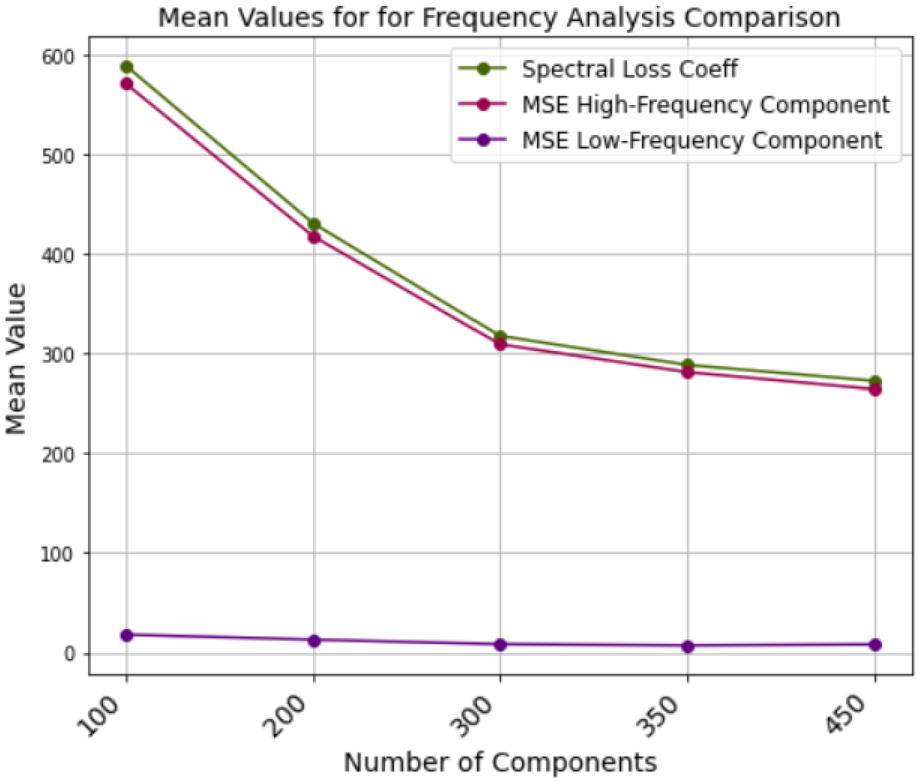


Figure 4.7: Comparison of Reconstructed Images at Various K Singular Values in Frequency Domain

Models Trained on HAM10000 Have resized the input images to reduced dimensions for reducing latency and reduce memory consumption.[6] Hence, we also consider some reduced standard dimension sizes like (256,256) and also try reducing it down by one axis- across y-axis (450,150) , across x axis (300,600), across both axis - (200,100), (300,300). The selection of Resolution of Pixel size is random but less pixels mean less information resulting in less feature maps in convolution and faster training times and reduced inference times. We Compare the above mentioned metrics after resizing our reconstructed dataset to select the best performing resolution with optimal metrics. We find that at (200,100), we get least number pixel resolution and optimal values for PSNR, RMSE, SSIM.

In Conclusion, All our metrics maximize/minimize at k = 350 and k=400, and stabilize afterwords. In some metrics like Haralick textures where even minimum of noise can affect the feature, the euclidean distance between haralick features starts to increase after k = 400. Hence, we select the minimum of the two, k = 350 to compress image along y-axis and perform reconstruction. The Explained Variance Ratio at k =350 is also greater than 98%.

4.3. E-R Diagram

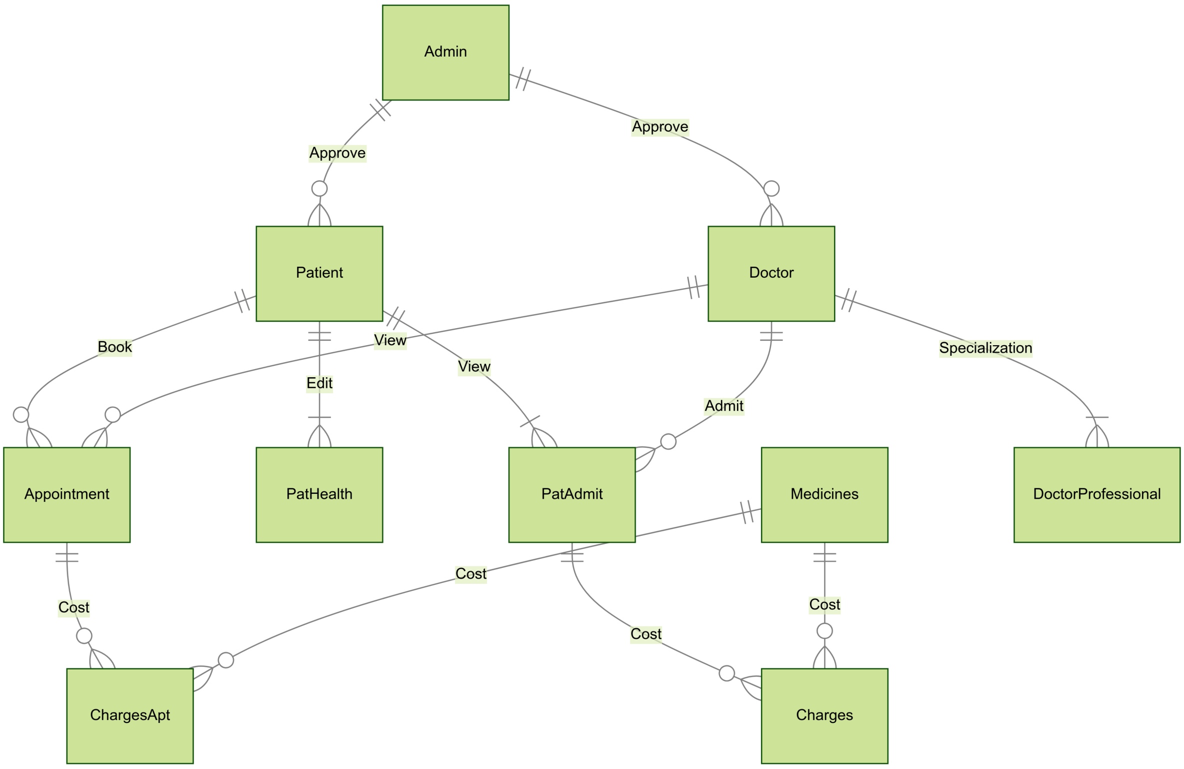


Figure 4.8: Entity Relationship Diagram

4.3.1. Cardinality, Participation, and Superkeys for Entities

Cardinality and participation restrictions manage the links between entities in the Doctor Appointment System, ensuring the database design's correctness and efficiency. Cardinality refers to the number of instances of one entity that can or must be connected with each instance of another entity, whereas participation indicates whether all or only certain entity instances take part in a connection. Superkeys are also essential for ensuring the uniqueness and integrity of entities within the system.

4.3.2. Cardinality in Relationships

Cardinality in the Doctor Appointment System is categorized into three main types: one-to-one (1:1), one-to-many (1:M), and many-to-many (M:M).

1. **One-on-One (1:1)** For example, each Admin is connected with precisely one User, and vice versa. This means that a single user profile connects only to one admin profile. In practice, this connection assures that an admin cannot be shared by numerous user profiles, and that each user who is an admin can only have one admin profile.

2. **One-to-Many (1:M)** Each Doctor can have many Appointments, but each Appointment is associated with only one Doctor. This mirrors the real-world scenario in which a doctor consults with numerous patients, yet each consultation is unique to one doctor. Similarly, each patient might have several appointments, but each appointment only applies to one patient. This guarantees that the system correctly records several interactions and consultations for each doctor and patient.

3. **Many-to-Many (M:M)** For example, the Appointment object facilitates a many-to-many link between patients and doctors. Each patient can see many physicians, and each doctor can see a large number of patients. This connection is critical for capturing the changes of patient-doctor interactions across time while also allowing flexibility and complete tracking of medical consultations.

4.3.3. Participation in Relationships

Participation restrictions specify whether all or some instances of an entity engage in a relationship, which is classified as entire (required) or partial (voluntary).

1. **Total Participation** For example, in the Patient-PatHealth interaction, each patient must have a PatHealth record. This implies that each patient in the system must have a health record, guaranteeing that no patient exists without documented health information. This obligatory linkage ensures that each patient's health condition is continually monitored and available.

2. **Partial Participation** For example, in the User-Admin connection, not all users are needed to be administrators, but every administrator must be associated with a user. This partial participation implies that, while a user can be an admin, every admin must have a user profile. This configuration provides for greater flexibility in user roles while also ensuring that admin profiles are properly authorised and handled.

4.3.4. Relationships and Integration

The Doctor Appointment System's linkages and integration are rigorously planned to enable a continuous flow of information and cooperation across multiple hospital operations. At its heart, the **User** entity serves as a hub, connecting to the **Doctor**, **Admin**, and **Patient** entities via foreign key connections. This design represents the real-world scenario in which a single person may perform many functions inside a hospital. For example, a **User** connected with the **Doctor** entity has information about their specialization, personal information, and professional status, but a **User** associated with the **Patient** entity includes extensive personal and medical history details.

Appointments are essential for hospital operations, and the **Appointment** entity is inextricably tied to the **Patient** and **Doctor** entities. This connection guarantees that each visit is precisely tracked, including information about the patient, the consulting doctor, appointment specifics, and status. Such connectivity enables the system to handle schedules more efficiently, facilitating both in-person and distant consultations via features such as video conference links.

Health management is further eased by the **PatHealth** entity, which is linked to the **Patient** entity and ensures that thorough health data are kept and easily available for each patient. This object stores crucial health measurements and history, which are required by clinicians to deliver personalized care. Similarly, the **PatAdmit** entity, which is linked to both the patient and the doctor, maintains hospital admissions, including admission and release dates as well as reasons for hospitalization. This link promotes efficient bed management and resource allocation.

Financial management is effectively linked across the **Medicines**, **Charges**, and **ChargesApt** entities. The pharmaceuticals entity lists all accessible pharmaceuticals, which are subsequently connected to financial records via the **Charges** and **ChargesApt** entities. These partnerships monitor the use and cost of medications throughout patient admissions and visits, guaranteeing correct invoicing and inventory management. Each charge record is coupled with either an admission (**PatAdmit**) or an appointment (**Appointment**), which allows for precise financial tracking and accountability.

Professional information on doctors is kept by the **DoctorProfessional** entity, which is linked to the **Doctor** entity. This link tracks professional activities such as appointment fees, admission fees, and the overall number of patients treated. Such integration aids in fee structure management and knowing the workload of medical professionals, guaranteeing equitable distribution and resource allocation.

Overall, the linkages and integration within the doctor appointment system guarantee that all parts of hospital operations are linked, allowing for efficient information flow and complete administration. This connection allows for more precise data monitoring, improves departmental collaboration, and, eventually, leads to better patient care and operational efficiency.

4.4. Class & Object Diagram

4.4.1. Class Diagram of Django Application

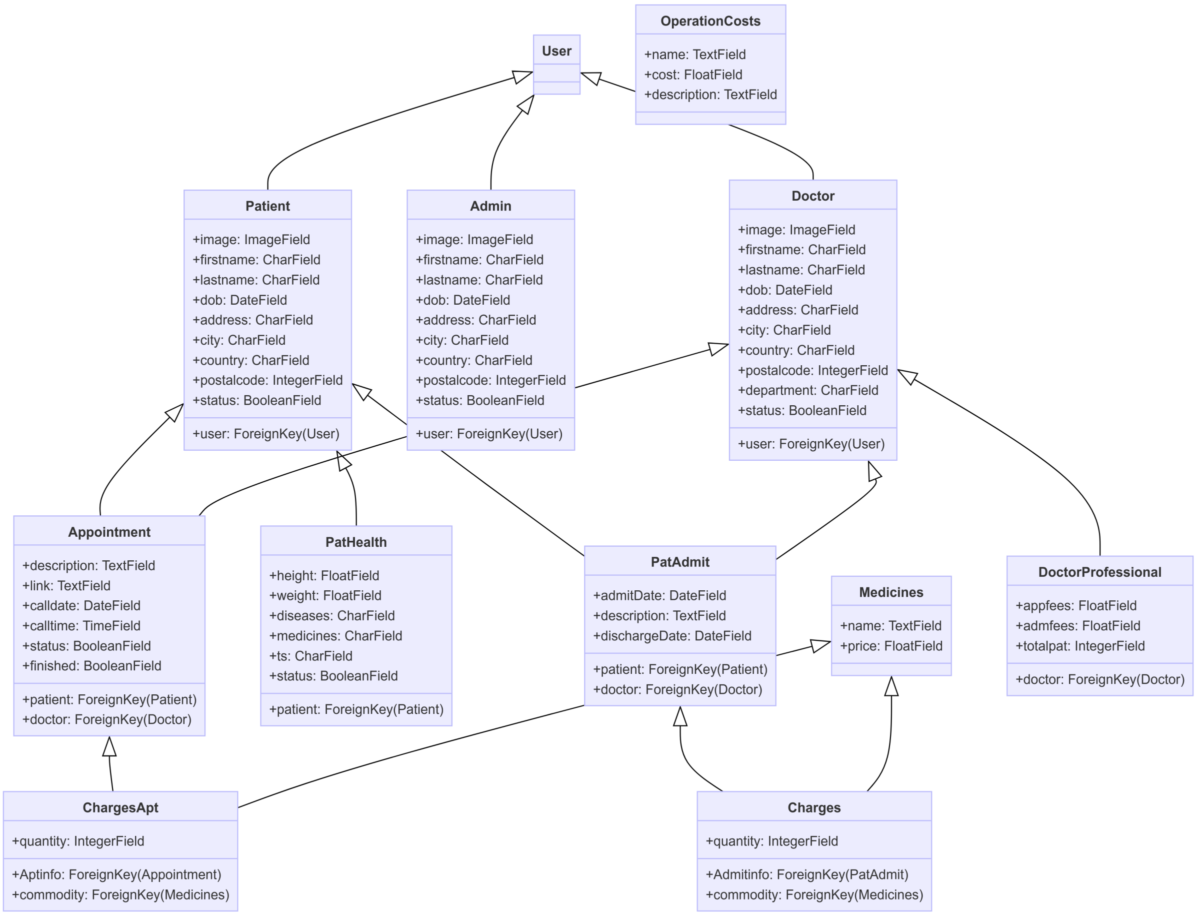


Figure 4.9: Class Diagram of Django Application

The Figure 4.9 outlines the structure of a medical management system, illustrating the relationships and attributes of various entities.At the core, the User class is the base entity from which Doctor, Admin, and Patient classes inherit. Each user type has attributes such as image, firstname, lastname, dob (date of birth), address, city, country, postalcode, and status.Doctors have additional attributes related to their department and professional status. They are linked to DoctorProfessional, which includes details about appointment fees, admission fees, and total patients.

Patients are associated with PatHealth, containing their health metrics (height, weight, diseases, medicines), and PatAdmit, detailing admission information like admit date, description, discharge date, and the attending doctor.Appointments connect Patients and Doctors, including description, video call link, call date, call time, status, and completion status.Medicines and Charges are linked, where Charges records information about the patient admission (via PatAdmit), associated medicines, and their quantities. Similarly, ChargesApt records the charges related to appointments.OperationCosts is another class detailing additional costs, with attributes for name, cost, and description. This structure supports a comprehensive system for managing user roles, patient health records, appointments, medical charges, and professional details of doctors, ensuring an integrated and efficient hospital management system.

4.4.2. Class Diagram of StreamLit Application

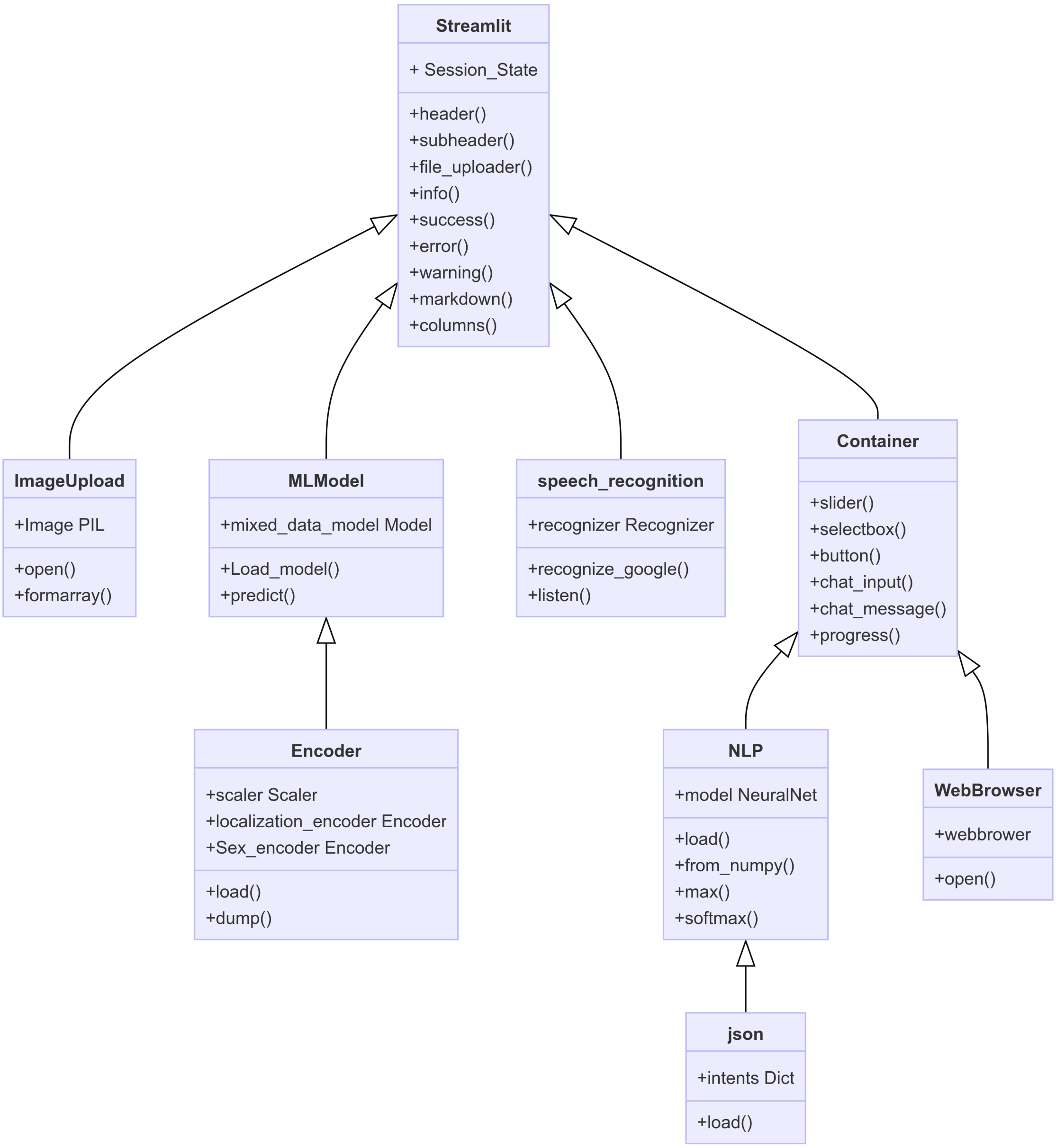


Figure 4.10: Class Diagram of StreamLit Application

The Figure 4.10 illustrates a medical application's architecture. Streamlit serves as the main interface, offering functionalities for headers, file uploads, and notifications. It interacts with various components to ensure a seamless user experience. Image Upload handles the processing of user-uploaded images. The ML Model predicts the likelihood of skin cancer, supported by the Encoder for data scaling and encoding. NLP interprets patient descriptions and symptoms, connected to a JSON component for managing intents. speech recognition enables voice interaction. The Container class provides UI elements like sliders, selection boxes, buttons, chat inputs, and progress indicators, and interacts with Web Browser components for external web interactions. These interconnected classes and functions create a comprehensive, user-friendly medical application.

4.4.3. Object Diagram of Django Application

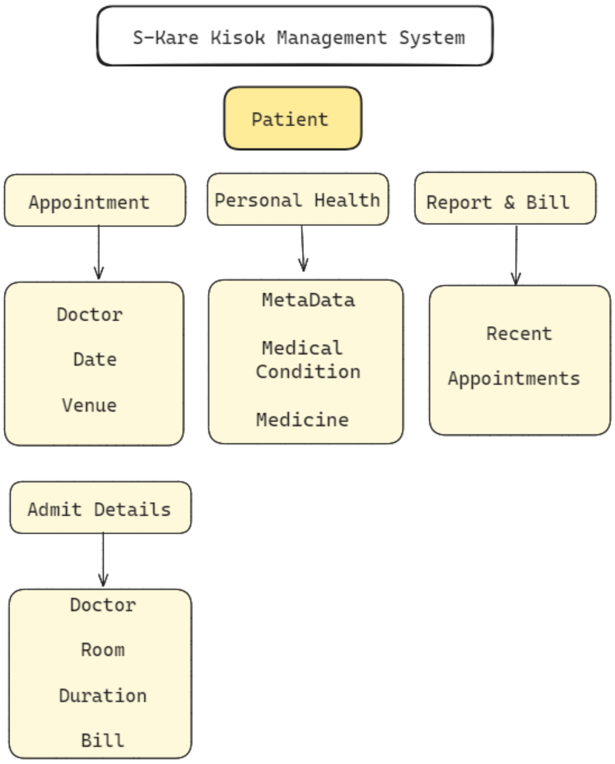


Figure 4.11: Object Diagram of Django Application

The above Figure 4.11 describes the object diagram of the Patient in the Django management system. It has three main functional classes. Under Appointment, it has three key attributes that dictate the details of the appointment created by the patient and then it is approved by the admin. The Personal Health function can be edited by the user without any authority from the admin, its initial data is filled during registration and can be updated later under the Your Health menu in the Navigation bar. Lastly, we have the Reports & Bills section where a patient can download a PDF receipt of their previous appointments and hospital bills and reports.

4.4.4. Object Diagram of StreamLit Application

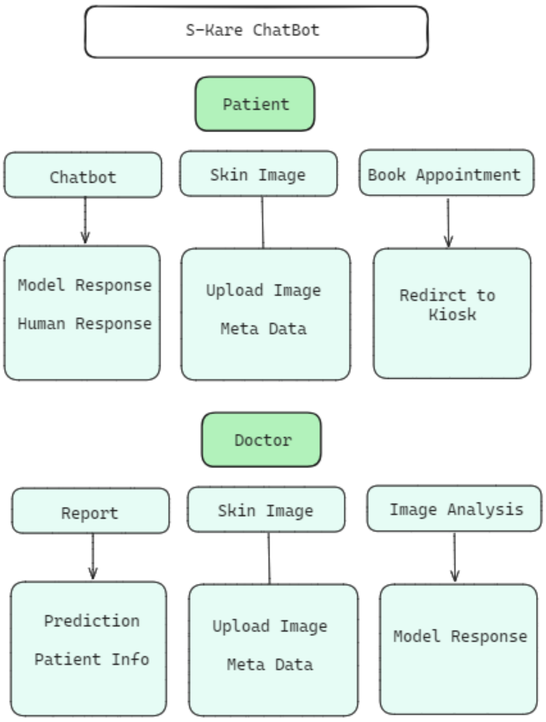


Figure 4.12: Object Diagram of StreamLit Application

The above Figure 4.12 is an object diagram of the SKare Chatbot design to be more publicly accessible. The Chatbot application has two different one for the public and the other for the doctor, the key difference between them is report generation present on the doctor side which comprises the details of diagnostics made by the machine learning algorithm, whereas on the public side, we have NLP chatbot which interacts with the user and understands their concern based of which it recommends to book an appointment with the nearest kiosk clinic

4.5. Data flow diagram

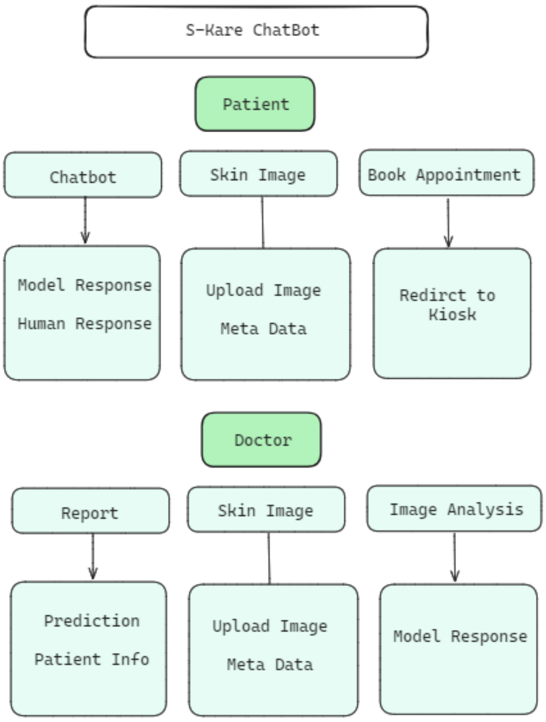


Figure 4.13: Object Diagram of StreamLit Application

The above Figure 4.13 represents the Data Flow diagram for the Appointment Booking Feature. Here once the patient has been authenticated they can book appointments from the Booking menu. After submitting the Appointment form, the application created a POST request to the server. If the request is valid, the new appointment is created in the database and reflected on the user interface as well as is updated to the admin for them to approve or put the appointment on hold.

5. RESULTS AND DISCUSSIONS

Let us review the basic classification and their significance before we dive into the results.

$$\text{Precision} = \frac{TP}{TP + FP}$$

The precision of the model is defined as the amount of true positive predictions (i.e., accurately predicted positive cases) among all positive predictions. When false positives are expensive, like in fraud detection or medical diagnosis, it is particularly useful.

$$\text{Recall} = \frac{TP}{TP + FN}$$

Recall is a metric that quantifies the percentage of true positive predictions that the model correctly recognized out of all actual positive cases. It is sometimes referred to as sensitivity or true positive rate. When identifying all positive examples is crucial, like in the case of disease identification or customer churn prediction, it is useful.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

Accuracy measures the proportion of correct predictions (both true positives and true negatives) out of all predictions made by the model. It provides an overall assessment of how well the model performs across all classes. However, accuracy may not be suitable for imbalanced datasets where one class dominates the others.

Table 5.1: Training and Validation Metrics for different architectures and datasets

Training and Validation Metrics (in %) - Original vs Reconstructed Dataset					
Architecture	Dataset	Training		Validation	
		Precision	Accuracy	Precision	Accuracy
Lenet-5 Based Neural Network	Original	99.84	99.84	73.72	74.06
	Reconstructed	98.72	98.66	78.31	77.55
Alexnet Based Neural Network	Original	97.79	97.90	80.24	80.00
	Reconstructed	96.94	97.05	83.55	83.87
Custom Neural Network-1	Original	97.61	97.66	80.55	81.03
	Reconstructed	97.03	97.20	80.18	80.38
Custom Neural Network-2	Original	97.12	97.35	77.09	78.13
	Reconstructed	97.15	97.26	80.36	80.44

Table 5.2: Testing Metrics (Precision, Accuracy, and Recall) for different architectures and datasets

Testing Metrics (in %) - Original vs Reconstructed Dataset				
Architecture	Dataset	Metrics		
		Precision	Accuracy	Recall
Lenet-5 Based Neural Network	Original	71.73	72.22	71.76
	Reconstructed	70.69	69.72	69.25
Alexnet Based Neural Network	Original	74.73	73.88	73.43
	Reconstructed	77.77	77.50	77.07
Custom Neural Network-1	Original	76.48	75.83	75.35
	Reconstructed	74.91	73.05	72.62
Custom Neural Network-2	Original	72.64	71.66	71.13
	Reconstructed	77.63	77.22	77.03

Training Benchmark = 90% , Testing - 2% of Error Accepted , Validation = 80%

For Lenet-5 based Neural Network, Validation Baseline is set to 75% due to less number of convolutions and less depth of network.

From Table 5.1 and Table 5.2, we observe classification metrics for the four architectures in both the training and testing phases, comparing performance on the original dataset and the dataset reconstructed using the top 350 singular values and vectors obtained through Singular Value Decomposition (SVD). The metrics indicate that the models achieve or even outperform the baseline performance of models trained on the original dataset, as evidenced by the comparison of accuracy, precision, and recall in both phases. This observation suggests that the SVD-based reconstruction method effectively preserves essential information from the original data. Despite reducing the dimensionality of the data through SVD, the reconstructed dataset retains sufficient discriminative features, allowing the models to perform comparably well.

In technical terms, accuracy measures the proportion of correctly classified instances among all instances evaluated. Precision measures the proportion of true positive predictions among all positive predictions made by the model, indicating how precise the model is when it predicts a positive class. Recall measures the proportion of true positive predictions among all actual positive instances in the dataset, indicating the model's ability to correctly identify positive instances.

The higher precision, recall, and accuracy achieved by the AlexNet-based Neural Network compared to the model trained on the original dataset suggest that the reconstructed dataset maintains or even enhances discriminative features important for classification tasks.

Table 5.3: Storage Space Optimization

Dataset	Original Size (GB)	SVD Reconstructed Size (GB)	Compression Ratio	Space Saved (%)
HAM10000 + Custom Images	2.72	0.72	3.77	73.5

We finally conclude here, with a compressed and reconstructed dataset, we achieve a compression ratio of 3.77 and save almost 73% storage space. This can be effective on any platform and storage system. The Same Model Architecture is able to achieve baseline performance so features or essential information is lost.

$$\text{Compression Ratio} = \frac{\text{Size before compression}}{\text{Size after compression}}$$

$$\text{Storage Saved (\%)} = \left(1 - \frac{\text{Size after compression}}{\text{Size before compression}}\right) \times 100$$

Now , we analyse the results of Model Pruning on Model architectures presented and the reconstructed dataset. The classification metrics for pruned models are compared in following tables. For displaying reduction of a pruned model we use convention R-I, Prune

Factor, which means the model has been pruned for I Number of iterations using same prune factor. If reduction is present as an array (R-I1, RI2) - (X1 %, X2%), it means for iteration I1 , X1 was used as pruning factor and X2 for I2 Iteration. We stop the pruning when the model architecture becomes very simple or when baseline performance is not met after 2 iterations of Pruning factor of 33.3%.

Table 5.4: Training and Validation Metrics for different architectures and datasets

Training and Validation Metrics (in %) - Iterative Neuron and Convolutional Filter Reduction					
Architecture	Model	Training		Validation	
		Precision	Accuracy	Precision	Accuracy
Lenet-Based architecture	Original	99.99	99.99	78.36	78.86
	R1: 50%	99.99	99.99	73.54	73.60
	R1: 33%	99.69	99.69	72.27	72.48
	R2: 33%	99.72	99.78	57.72	58.09
Alexnet Based Neural Network	Original	97.24	97.41	81.18	81.54
	R1: 50%	99.09	99.12	82.00	82.43
	R2: 50%	97.94	98.08	83.36	83.46
	R3: 50%	95.73	96.05	82.64	83.09
	R4: 50%	78.24	83.30	80.91	85.78
	R5: 50%	79.91	85.59	80.82	84.95
Custom Neural Network-1	Original	96.45	96.56	81.27	81.50
	R1: 50%	92.39	92.61	82.00	82.79
	R2: 50%	84.85	86.66	72.82	74.18
	R1, R2: 50%, 33%	85.98	78.73	78.73	80.77
Custom Neural Network-2	Original	92.30	92.89	80.09	80.87
	R1: 50%	87.36	88.73	69.27	69.60
	R1: 33%	86.21	87.96	73.55	77.91
	R2: 33%	79.58	83.98	76.36	80.44

Table 5.5: Testing Metrics for different architectures and datasets

Testing Metrics (in %) - Iterative Neuron and Convolutional Filter Reduction				
Architecture	Model	Metrics		
		Accuracy	Precision	Recall
Lenet-Based architecture	Original	72.22	73.08	71.89
	R1: 50%	67.22	66.90	66.74
	R1: 33%	68.88	69.68	68.33
	R2: 33%	63.88	61.94	63.38
Alexnet Based Neural Network	Original	73.33	74.81	73.08
	R1: 50%	75.00	74.95	74.58
	R2: 50%	73.33	75.65	73.13
	R3: 50%	73.88	73.89	73.58
	R4: 50%	70.83	76.88	70.64
	R5: 50%	72.77	77.91	72.74
Custom Neural Network-1	Original	76.94	77.42	76.33
	R1: 50%	76.66	77.27	76.23
	R1: 33%	71.66	75.22	70.83
	R1, R2: 50%, 33%	74.72	75.07	74.05
Custom Neural Network-2	Original	73.88	76.14	73.61
	R1: 50%	66.94	69.55	66.11
	R1: 33%	70.55	69.79	69.96
	R2: 33%	71.94	78.19	72.03

Observing Table 5.4 and Table 5.5, we can see pruned architectures of the proposed models and classification metrics evaluated on the reconstructed Dataset.Let's Look at the Models one by one, for Lenet-5 based Neural Network we see, we can achieve baseline metrics at Training and Validation but exceeds the 2% error rate in testing metrics. This is due to the simple, shallow architecture of Lenet-5 Architecture used in Image Model with only 2 convolution layers.For Alexnet Based Neural Network, we achieve baseline performance in training, testing and validation in comparison with the original model at iteration 3 with pruning factor 3. We keep reducing in with same pruning factor in iteration 4 and 5 but not achieve baseline performance on testing and training, but achieve it on validation. This suggests although the model is learning patterns quickly, it needs more fine-tuning of hyper-parameters and a longer time to converge to learn more patterns effectively. Also, it implies the model is not learning the patterns quickly and has become simple. For Custom Neural Network -2 we achieve pruned model after one reduction iteration 2 with prune factor of 33.33% and for Custom Neural Network- 1 we achieve it with first reduction using factor of 50% and second iteration of 33.33%.

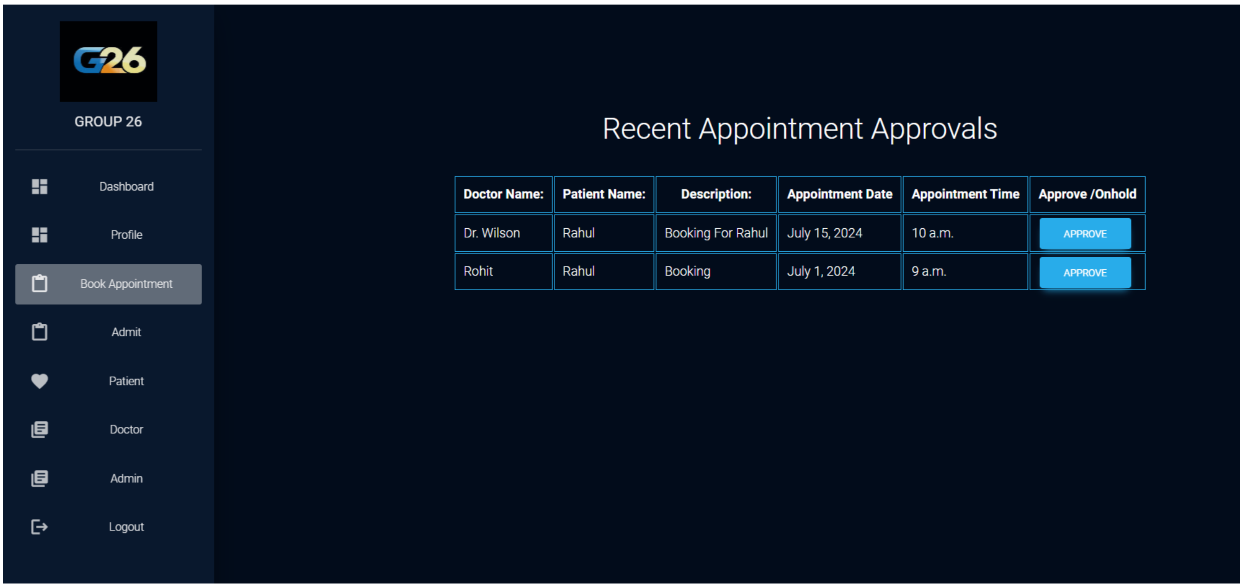
Table 5.6: Model Compute Optimization

Architecture	Learnable Parameters ($\times 10^5$)			Model Size (MB)	
	Original	Reduced	Prune Factor	Original	Reduced
Lenet-5 based Neural Network	14.04	8.34	59.40	352	9.5
Alexnet based Neural Network	110.60	6.08	5.49	444	2.17
Custom Neural Network-1	28.42	10.13	35.64	113	3
Custom Neural Network-2	31.16	3.70	11.87	125	4.56

As per Table 5.6, we achieve a significant reduction in number of learnable parameters and model size especially for Alexnet-based neural network. This will be used in the proposed web application due to its small size and faster inference times. In the pruned model for this architecture, we reduce the learnable parameters almost to 5% of ones contained in original model. For All Architectures there is a reduction in learnable parameters almost to 5% to 35% of learnable parameters in original ones. For Lenet-5 the final reduction of parameters is only by 59% due to its simple architecture. The Model Size is also reduced significantly to less than 10MB so it can be deployed on small scale hardware too. Overall for all architectures The final training time for all Architectures were less than 10 min observed an reduction of 10 to 20 mins of training time on GPUP100, which is very significant when same model is trained in resource constrained hardware CPU Architectures.

5.1. Application Flow & Interface

Django Kiosk Management System



Recent Appointment Approvals

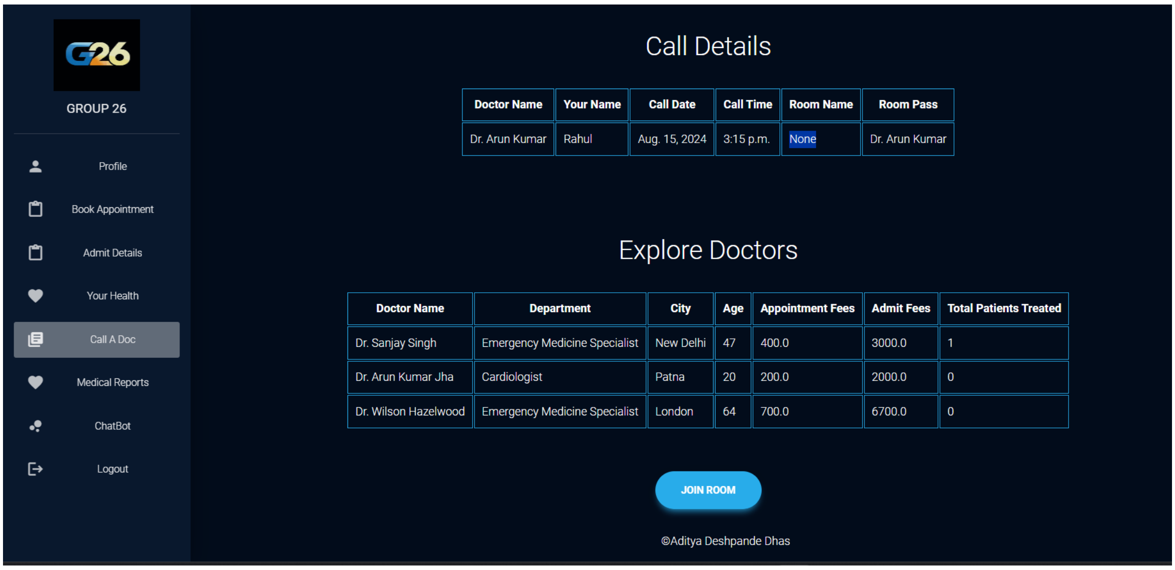
Doctor Name:	Patient Name:	Description:	Appointment Date	Appointment Time	Approve /Onhold
Dr. Wilson	Rahul	Booking For Rahul	July 15, 2024	10 a.m.	APPROVE
Rohit	Rahul	Booking	July 1, 2024	9 a.m.	APPROVE

Figure 5.1: Admin Appointments Approval Page

Admin

To sign up for an admin account at the hospital, users must first complete the registration process. Afterwards, their login requires approval from other admins. Once approved and logged in, admins can view and approve doctors who have applied for jobs at the hospital. They also have the authority to admit patients, view and approve patient details, and book appointments. Admins can generate and download invoices, taking into account medicine costs, room charges, doctor fees, and other charges. Additionally, they can view, book, and approve patient appointment requests.

Patient



Call Details

Doctor Name	Your Name	Call Date	Call Time	Room Name	Room Pass
Dr. Arun Kumar	Rahul	Aug. 15, 2024	3:15 p.m.	None	Dr. Arun Kumar

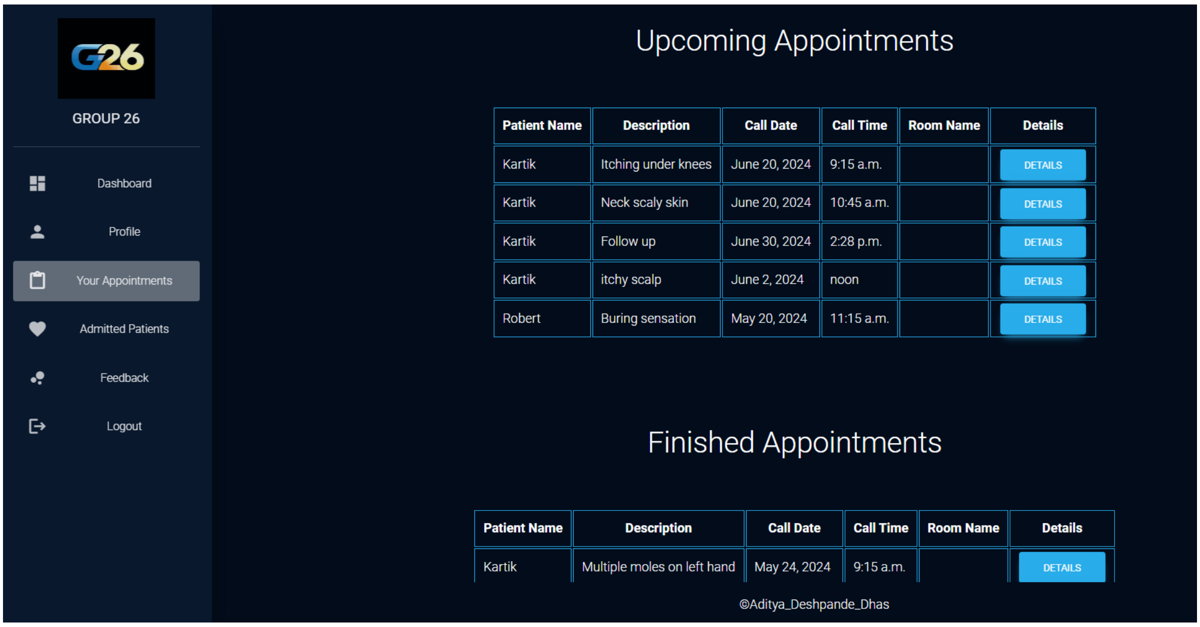
Explore Doctors

Doctor Name	Department	City	Age	Appointment Fees	Admit Fees	Total Patients Treated
Dr. Sanjay Singh	Emergency Medicine Specialist	New Delhi	47	400.0	3000.0	1
Dr. Arun Kumar Jha	Cardiologist	Patna	20	200.0	2000.0	0
Dr. Wilson Hazelwood	Emergency Medicine Specialist	London	64	700.0	6700.0	0

Figure 5.2: Patient Appointments & Doctor Details Page

To gain admission to the hospital, patients must first create an account. Following registration, patients require approval from the hospital admin to log in. Once logged in, patients can view the details of their assigned doctor, including specialization, mobile number, and address. Patients can also check the status of their booked appointments, whether pending or confirmed by the admin. Additionally, patients can book new appointments, which also require admin approval. After being discharged by the admin, patients can view and download their invoices in PDF format.

Doctor



Upcoming Appointments

Patient Name	Description	Call Date	Call Time	Room Name	Details
Kartik	Itching under knees	June 20, 2024	9:15 a.m.		DETAILS
Kartik	Neck scaly skin	June 20, 2024	10:45 a.m.		DETAILS
Kartik	Follow up	June 30, 2024	2:28 p.m.		DETAILS
Kartik	itchy scalp	June 2, 2024	noon		DETAILS
Robert	Buring sensation	May 20, 2024	11:15 a.m.		DETAILS

Finished Appointments

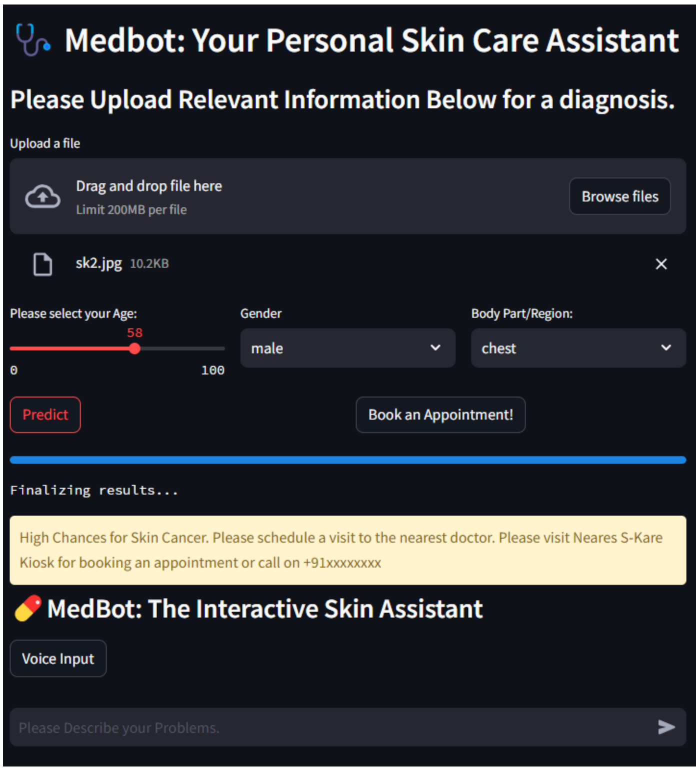
Patient Name	Description	Call Date	Call Time	Room Name	Details
Kartik	Multiple moles on left hand	May 24, 2024	9:15 a.m.		DETAILS

©Aditya_Deshpande_Dhas

Figure 5.3: Doctor Appointment Page

To apply for a job at the hospital, doctors must first complete the application process. Upon submission, they need approval from the hospital admin before they can log in. Once logged in, doctors can only view the details of the patients assigned to them, including symptoms and names. They can access their entire patient list and generate video calling links for consultations. Additionally, doctors can view details regarding their patients' appointments and admissions.

Streamlit Chatbot



Medbot: Your Personal Skin Care Assistant

Please Upload Relevant Information Below for a diagnosis.

Upload a file

Drag and drop file here
Limit 200MB per file

sk2.jpg 10.2KB

Please select your Age: 58

Gender: male

Body Part/Region: chest

Finalizing results...

High Chances for Skin Cancer. Please schedule a visit to the nearest doctor. Please visit Neares S-Kare Kiosk for booking an appointment or call on +91xxxxxxxx

MedBot: The Interactive Skin Assistant

Please Describe your Problems.

Figure 5.4: NLP Chatbot Interface for Image analysis

The application workflow begins with patient interaction, where the chatbot greets the patient and asks them to describe their skin issue in detail. Based on the patient's description, the chatbot assesses the symptoms to determine if an image upload is necessary for further evaluation. If an image is needed, the patient is prompted to upload a clear photo of the affected area and provide additional information such as their age and the location of the skin issue.Once the image and details are submitted, the application processes the image using a pre-trained model designed to predict the likelihood of skin cancer. This image-processing step involves analyzing the visual data to identify any concerning patterns or abnormalities that might indicate a potential risk.After the image is processed, the prediction result is displayed to the patient. This result includes a likelihood score indicating the risk of skin cancer based on the model's analysis. If the prediction indicates a high risk, the patient is immediately redirected to the kiosk page, where they can book an appointment with a doctor for a comprehensive evaluation and further medical assistance. This seamless workflow ensures that patients receive timely and accurate assessments, and those at high risk can quickly schedule a follow-up with a healthcare professional

6. CONCLUSION

In Conclusion, the Storage Optimization using Singular Value Decomposition(SVD) significantly compresses data while capturing most of the variance and discriminatory features of the images. This reduces storage requirements on any platform significantly and also be performed at scale without much loss in information. There is also another observation here, that reconstructed images now have psycho-visual and inter-pixel redundancy introduced so comparatively less number of convolution filters can identify the discriminatory patterns. The disadvantage is the increased overhead time and cost of analysis of what number of singular values needed for reconstruction so there is no compromise in image quality and essential features are preserved. Model Pruning reduces number of hidden neurons using a pruning factor for convolution layers and the next two fully connected layers. This helps in reduction of model size, reduction in number of filters, number of learnable parameters and faster training and inference times. This also makes it easy to deploy the model on small scale devices. The disadvantage here is this is a quite simple, uninformed and naive approach on model pruning and requires lots of iterations in fine-tuning it. Thus, a combination of SVD and Model Pruning can help in optimizing storage and compute of a Deep Neural Network based Application used in Skin Cancer detection for Deep and complex architectures provided a comprehensive analysis and fine-tuning is done by scientists and researchers.

This medical kiosk management application is designed to enhance the efficiency of medical services by leveraging modern web technologies. The web application is designed to manage a medical kiosk, providing a streamlined and efficient way to handle various tasks within a healthcare setting. With Key functionalities of the application include user authentication, data management, and appointment booking, all integrated into a cohesive system.The application aims to reduce administrative burdens, improve patient care, and streamline operations within a medical facility by integrating key functionalities into a single platform. Whether managing patient records, booking appointments, or ensuring secure access to sensitive information, this application provides a reliable and user-friendly solution for medical professionals and patients.

Future Works can include but not limited to reducing the number of convolution filters based on an informed heuristic. A standard framework can be developed using Model Pruning and SVD Storage Optimization to extend to other problem domains.

References

[1] World Cancer Research Fund International, “Skin cancer statistics,” 2023, accessed: 2024-06-19. [Online]. Available: https://www.wcrf.org/cancer-trends/skin-cancer-statistics/

[2] S. T. Lal, R. P. Banipal, D. J. Bhatti, and H. P. Yadav, “Changing trends of skin cancer: A tertiary care hospital study in malwa region of punjab,” *Journal of Clinical and Diagnostic Research*, vol. 10, no. 6, pp. PC12–PC15, June 2016, doi: 10.7860/JCDR/2016/19064.8018.

[3] Melanoma UK, “Melanoma facts & stats,” https://www.melanomauk.org.uk/melanoma-facts-stats, accessed: 2024-06-19.

[4] W. Salehi, S. Khan, G. Gupta, B. Alabduallah, A. Almjally, H. Alsolai, T. Siddiqui, and A. Mellit, “A study of cnn and transfer learning in medical imaging: Advantages, challenges, future scope,” vol. 15, p. 5930, 03 2023.

[5] A. Kumar, “The transformation of the indian healthcare system,” *Cureus*, vol. 15, 05 2023.

[6] B. Shetty, R. Fernandes, A. Rodrigues, R. Chengoden, S. Bhattacharya, and K. Lakshman, “Skin lesion classification of dermoscopic images using machine learning and convolutional neural network,” *Scientific Reports*, vol. 12, 10 2022.

[7] K. M. Hosny, M. A. Kassem, and M. M. Foaud, “Skin cancer classification using deep learning and transfer learning,” in *2018 9th Cairo International Biomedical Engineering Conference (CIBEC)*, 2018, pp. 90–93.

[8] M. Ali, M. S. Miah, J. Haque, M. M. Rahman, and M. Islam, “An enhanced technique of skin cancer classification using deep convolutional neural network with transfer learning models,” *Machine Learning with Applications*, vol. 5, p. 100036, 04 2021.

[9] P. Naga Srinivasu, J. Gnana Siva Sai, M. F. Ijaz, A. K. Bhoi, W. Kim, and J. J. Kang, “Classification of skin disease using deep learning neural networks with mobilenet v2 and lstm,” *Sensors*, 04 2021.

[10] D. N. A. Ningrum, S.-P. Yuan, W.-M. Kung, C.-C. Wu, I.-S. Tzeng, C.-Y. Huang, J. Y.-C. Li, and Y.-C. Wang, “Deep learning classifier with patient’s

metadata of dermoscopic images in malignant melanoma detection," *Journal of Multidisciplinary Healthcare*, vol. 14, pp. 877 – 885, 2021. [Online]. Available: https://api.semanticscholar.org/CorpusID:233408343

[11] K. O'Shea and R. Nash, "An introduction to convolutional neural networks," *CoRR*, vol. abs/1511.08458, 2015. [Online]. Available: http://arxiv.org/abs/1511.08458

[12] H. R. Swathi, S. Sohini, Surbhi, and G. Gopichand, "Image compression using singular value decomposition," in *IOP Conference Series: Materials Science and Engineering*, vol. 263, no. 4. IOP Publishing Ltd, 2017, p. 042082.

[13] P. Tschandl, "The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions," 2018. [Online]. Available: https://doi.org/10.7910/DVN/DBW86T

[14] R. M. Haralick, K. Shanmugam, and I. Dinstein, "Textural features for image classification," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. SMC-3, no. 6, pp. 610–621, 1973.