# Bone Fracture Detection on Multiple Body Parts Using Deep Learning CNN Models

Bill Joshua Swamidoss

Balasubramanian C

# Bone Fracture Detection on Multiple Body Parts Using Deep Learning CNN Models

**Abstract**

Research done on human bone structures using deep learning is still in its growing stages. Among the total 206 bones in the human body, the humerus is considered one of the most crucial ones, which we will focus on in this research along with the shoulder bone structure. The goal is to train the model to classify more than one bone in a human body and report if the bone is healthy or fractured through deep learning classification models. With TensorFlow, which is an open source library for deep learning, tested the image classification with different DNN models, and our research evaluates the effectiveness of CNN models such as VGG16, ResNet50, DenseNet121, and Densenet169 and has also created a custom CNN architecture for real-time comparison between pre-existing optimized models for transfer learning and custom-made models. Publicly available "MURA" dataset with large number of X-ray images comprising more than 40,000 broken and healthy bone structures. We have trained and tested with multiple models that are mentioned above, and the Densenet121 outperformed all lighter models while testing with a single bone structure, and VGG16 achieved 0.70 training accuracy while working with humerus, and our customized CNN by giving an accuracy of 0.74 while loading over 1000 images of humerus. The Densenet121 model for the humerus bone produced 0.71 training accuracy for the shoulder bone structure while compiling the Densenet121 model. Our focus is to find the most suitable model for bone fracture and bone type classification and then make it as a two stage architecture, first stage to classify the type of bone and the second stage to classify between healthy and damaged bone. The architecture is structured as DenseNet169 for first stage and Densenet121 for second stage classification, and the reason of model selection is based on the single bone classification that we performed above. The study continues with more improvement to be tested and carried out in the future, to pick the optimal model combination to attain the goal with higher accuracy.

## 1 Introduction

Humerus is a narrow bone in the upper arm, stretching from the elbow to the shoulder. It joins the ulna and radius at the elbow joint and the scapula (shoulder blade) at the shoulder joint. The humerus and the shoulder bones play a vital role in the movement of the arm and are involved in various functions like lifting, throwing, and rotating. Also, as we know humerus has a simple straight bone structure, it's easier for the models to detect abnormalities in it. On contrary, the shoulder is a complicated joint, harder to classify and analyse fractures, hence in our study we can analyse how far can we train our model with narrow and complex bone. The use of X-ray images, which are widely available and less expensive compared to other imaging modalities, highlights the study's commitment to practicality and accessibility in medical settings. Hence its natural to obtain less accuracy while including shoulder along with humerus. Model building and hyperparameter tuning had been done to improve performance and the issue has been partially handled with image preparation. Included Canny edge detection function and data augmentation to improve accuracy.

Our goal is to create a deep learning model that uses CNN architectures and transfer learning techniques to categorize X-ray and scanned pictures of humerus and shoulder bone fractures to provide the most accurate diagnosis possible.


Figure 1: Bone Structure

The project aims to create an automated classification system that can accurately detect multiple types of bones in the human anatomy and then further classify between the fractured and healthy shoulder and humerus bones from the X-ray images, improving diagnostic efficiency and potentially assisting healthcare professionals in the early identification of bone injuries.

This classification is crucial, as it provides a fast and accurate path to detect bone fractures from X-ray images, which can significantly improve the diagnosis process, especially while dealing with a bulk number of cases at short intervals. Early and precise identification of fractures is crucial for effective treatment and recovery, reducing the risk of complications such as misalignment or delayed healing. By automating this process with deep learning, healthcare systems can alleviate the workload on radiologists, especially in settings with limited medical resources, while ensuring consistent and reliable fracture detection. This process may still need to be improved in its accuracy to be reliable in real-time scenarios; for that, each bone in the human body must be considered, and the model should be trained to handle all of them. In this paper we have taken two of the major skeletal supporting bone structures, the shoulder and humerus, and our average accuracy reaches 0.72 in classifying between the bones and then in classifying the fracture-positive images.

## 1.1 Research Question

How effective is a deep learning-based CNN image recognition system in identifying bone structures and detecting the fractures from diagnosed images, and how accurately can it generate results in multi-class bone classification?

## 1.2 Problem Solution

The problem question has been addressed in our project as we have used two different bone structures and then diagnosed them for fracture detection using advanced deep learning models and have obtained a decent accuracy through our trained models. This forms the base of our problem, and further in the same way, many more bone structures can be incorporated into our model, and the accuracy could be improved with different combinations of models, along with the fracture location indication system that can be added in future development of this project. In this image classification we have utilized multiple models and have compared their performance to pick the optimal performing model, and as mentioned, further investigation can be carried out to find even more suitable models in the future.

# 2 Materials and Methods

## 2.1 Database

In this paper, we have employed the X-ray images from the MURA database with and without fractures in the conducted experiment. This is a publicly available database, and the data were collected by Stanford University for the bone X-ray deep learning competition. This dataset comprises multi-view musculoskeletal radiographs comprising seven organs from the human body, namely the elbow, finger, hand, humerus, forearm, shoulder, and wrist. The total number of images taken from 12173 patients for a decade of period and it equals 40561 images. Each of these images is run through reliability checks by radiology experts and gets annotated to determine normal and abnormal cases; here, fractures and hardware implants are considered common abnormalities. Even though this dataset has seven bone parts covered, most studies are done based on specific bone structures. This is the reason why we have decided to take two or more bone parts in this project paper.

## 2.2 Deploying and Extracting Data (MySQL)

Our dataset has been moved to a MySQL database, and the images are stored as table for retrieving them back into the planned directory structure. The database table has the following features, image path (VARCHAR()), isfractured label (Boolean), which denotes either 0 or 1, and a study feature that tells us the type of bone, in this case humerus and patientid. The same table has been created for Shoulder bone structure. Then the data is extracted and moved in the system’s local directory in a specific directory structure, and its directly used as training and validation data sets. The test set is taken as 20 percent of the training set.

### 2.3 Data Preparation

From the MURA database, we have taken the shoulder and humerus bone X-ray images to initiate our research. This leaves us with precisely 9771 images comprising both fractures and healthy shoulder and humerus bones. We have used the provided labels to separate the training images into two divisions, positive (fractured) and negative (healthy), with the help of directory structures. The training and the validation image labels are provided in the MURA database, and further we have taken 10 percent of the training images as testing data. Two model stages are applied to this dataset structure: model one is trained to identify the kind of bone, and the second model is taught to distinguish between bone structures that are damaged and those that are healthy.

### 2.4 Novelty of the Study

In our research, we have done the bone fracture classification of more than one part of the body, and this can be further increased to the whole body in the future as a multiclass classification. We have made two stages of classifications; the first model is a multiclass classification model, as it will classify between different human body part bone structures, and in the second stage model, In order to provide insight into the model's attention, we distinguish between healthy and shattered bones and show the hotspot by highlighting the areas in the input image that were most important to the network's conclusion.

## 3 Related Work

In this section you need to situate your work in the academic literature; this entails a critical (positive, negative, helpful) review of similar work. If you can't find similar work, you haven't looked hard enough. Ideally, you want to be reading around 50 papers; of which at least 25 should appear in the paper itself. Note that urls are not references, they are footnotes.

You are expected to provide a critical/analytic overview of the significant literature published on your topic. Comment on the strength and weakness/limitation of work in each reviewed paper.

The literature review should end in a paragraph that summarises the findings from the state of the art, why the previous solution are not adequate and justifies the need for your research question.

The content sections of your report should of course be structured into subsections. Note that here there are 2 subsections subsection 1.2 and subsection **??**.

### 3.1 Research on Bone Fractures

Healthcare field is among the most critical for the wellbeing of the society all over the world, hence it needs rapid and accurate diagnostics and quick responsiveness would make a difference in the healthcare system in the future. This improvement would be made possible with the help of the data availability and digitalization of medical records and patient details. One of the key motivations for applying deep learning in healthcare is its potential to alleviate the burden on overworked healthcare professionals, particularly in resource-constrained settings, by offering fast and consistent decision-making support. This technology can bridge the gap between growing patient demands and limited medical expertise, making healthcare more accessible and efficient globally. In recent years many have taken the challenge of bone fracture diagnosis research which gave insights for this research analysis. CNN is the choice of framework in deep learning specifically for image processing. The first study guides us with CNN and traditional edge detection techniques, to handle the noisy images and detect fine details in medical images. X-ray images as input, the study utilized Discrete Wavelet Transform (DWT) for edge detection and Spatial Fuzzy C-Means (SFCM) clustering for image segmentation. Even though no complicated neural network is involved, this still performed well with noisy and gave an accuracy of 0.78 and, they included a user-friendly MATLAB GUI for easy accessibility. As we focused more on the multi bone classification, this gave us the opportunity to learn edge detection and noise cleaning in the X-ray images and it highlights its potential to refine medical imaging techniques further.[1]

Following this study, we have another research done on bone fractures using CNN and this discusses more on the different approaches of implementation that we can endeavour while trying to achieve our optimal classification results. In this study they used publicly available bone x-ray dataset and taken 100 of such images. Here they focus only on the custom CNN model they evaluated and not much about image preprocessing. They further augmented the images to make them into a 400-image dataset by tilting and inverting the available images. Here they proved a basic solution on how CNN can be utilized for any bone image, but it doesn't prove to be better as it is limited to classifying only few bones with decent accuracy. With this in our mind, in our research, specific images are focused and targeted so that equal care can be distributed by the trained model for all the bones, and we have concluded that each bone may vary in training difficulty, for example long bones are easier to train and joints are harder to train. So, in our research we have picked one of the most complicated bone joint structure the shoulder and a narrow bone humerus to overcome the limitations of this research. [2]

We have also discovered a more advanced study paper. This suggests a method for classifying and detecting bone fractures by combining an ensemble Convolutional Neural Network model with fuzzy-based picture enhancement approaches. Here, a fuzzy-based boosting technique is used to manage the low contrast and noisy x-ray pictures, which present the greatest challenge in identifying fractures. This study is driven by the urgent need for accurate and fast identification of orthopaedic diseases, which is necessary for successful treatment and better patient outcomes. Fuzzy logic and histogram equalization are combined in the suggested system to improve image quality. A triangle membership function is used to dynamically modify contrast and lower noise. Clearer imaging of bone structures is ensured by this preprocessing phase, which is essential for further analysis. Here, they have integrated ResNet50 and VGG16 to produce an ensemble model. We have learned a lot for our study from this endeavor, which will help us in the future as we analyze the best model to handle x-ray pictures while also using specific bones, like the shoulder, humerus, and forearm. Additionally, we have used little preprocessing to preserve the uniqueness of our data.[3]

Another observation is the research, which was spurred by the high frequency of bone fractures and the drawbacks of conventional diagnostic techniques, which are frequently labor-intensive, prone to human error, and need a lot of manual labor. and to get around the traditional problem with medical scans, such as MRIs, CT scans, and x-rays. According to the study, accuracy is increased when a CNN and AlexNet architecture are used together. To increase the model's resilience and decrease overfitting, methods including color transformation and data augmentation are used. To train and assess the suggested model, the study makes use of the MURA dataset, which comprises a variety of X-ray pictures of human bones. The approach is intended to dynamically detect fractures using bounding boxes for accurate localization and categorize bones into fractured or non-fractured categories. Because the MURA dataset was used in this study, we now know more about how to handle the data and how to make our model better for subsequent research. This study addresses the issues with earlier methods by demonstrating that the model achieves excellent classification accuracy with the usage of AlexNet and DCNN. After our study has produced a high-performing single model, we can post-analyze our ensembling model and incorporate AlexNet [7]. This study makes a substantial contribution to the field of automated medical diagnostics by fusing cutting-edge technology with useful applications to improve the efficiency and accessibility of healthcare.[4]

## 3.2 Research on models

The research investigates the use of convolutional neural networks (CNNs) for the automatic categorization of healthy and broken bones for smaller and more efficient databases, while up until now, the emphasis has been on developing a dependable deep learning solution to handle vast numbers of images. The work uses data augmentation techniques including rotation, flipping, and shearing to increase the dataset in order to improve the model's generalization, as limited datasets in deep learning frequently result in overfitting. A bespoke CNN architecture that is intended to extract features and categorize images with high accuracy is then trained and tested using the enhanced dataset. This paper discusses the appropriate optimizer utilization and activation functions while focusing on the fundamental neural network configurations. While optimizing or fine-tuning our bespoke CNN models, this provided us with information and insights on how to construct our neural network to obtain a greater advantage. By using a bespoke CNN with the Softmax function and Adam optimizer, we have enhanced the study's results and achieved respectable accuracy on both small and large data sets.[5]

To add furthermore novelty, we have searched for research done with single model like MobileNet and EfficientNet, as we thought about bringing them into our consideration as they prove to be more lighter models compared to complicated models like DenseNet121 and Densenet169. As this novel approach uses the lightweight and computationally efficient model like MobileNetV2, the study aimed to provide a robust, scalable, and accessible tool that can streamline diagnostic processes in clinical settings. And on top of it through transfer learning, this study gains additional improvement in MobileNetV2 with the help of supporting pretrained models, such as ImageNet while freezing some pre-trained layers allows the model to retain relevant features while fine-tuning for the specific task of fracture detection. We have also implemented transfer learning by utilizing pre-trained models with even large dataset, with more than 2000 training and validation images. But MobileNetV2 may struggle while handling larger datasets.[6]

And on analysing our research and venturing in the models to be implemented in our research we came across research papers that uses multiple models to conclude the best performing one. It is made possible to implement multiple complex models like VGG16, ResNet50, Densenet121, InceptionV3, and Xception—alongside a Random Forest classifier to analyse medical images from the MURA dataset, a well-known repository of musculoskeletal radiographs. By leveraging transfer learning, the models were fine-tuned to work effectively with small datasets, a common limitation in medical imaging tasks. The images, spanning multiple body parts such as the elbow, finger, humerus, and wrist, were resized and normalized to facilitate classification into two tasks: binary classification (fractured vs. non-fractured) and multi-class classification (fracture localization across body parts). Notably, Densenet121 achieved superior results. From this study we have taken the DenseNet models as our primary architecture as its proven to perform better in multiclass, and our study analyse the densenet169 and densenet121 along with custom added layers to see if its possible to improve its performance in binary and multiclass at the same time.[7]

By integrating Explainable AI (XAI) into specific bone structures, this work took a step closer to realizing the goal of integrating all these models into healthcare systems, which is likely to happen soon. The model's predictions are made more transparent and comprehensible through the usage of Saliency Maps and GRAD-CAM. This fosters trust and enables well-informed clinical decision-making by enabling medical professionals to comprehend which areas of the X-ray pictures influenced the model's judgment. Additionally, it would be connected with an intuitive interface in the medical system. This would be a support to the medical professionals in all fronts and it can be used on critical decision makings as a by standing option, doctors can consider if they need to double check and evaluate their decision, and it would act as a preventive measure to avoid fatal mistakes.[10]

### 3.3 Deep Learning and AI Role in Healthcare:

All these research that we have explored here has a major goal, that is to enhancing patient care by minimizing diagnostic errors, reducing the reliance on specialized expertise, and accelerating the treatment process. And they also demonstrate the potential of AI-powered diagnostic systems not to overpower the healthcare work and replace them with machines. It is to support the radiologists and orthopaedic surgeons and similar professional to have technical support on top of their expertise in the field. In the end, each of these research adds to the increasing amount of data demonstrating the benefits of integrating AI in healthcare, opening the door to more effective, accessible, and precise diagnostic procedures.

## 4 Methodology

### 4.1 CNN Architecture

GRAD-CAM with Saliency Maps to improve the predictability and interpretability of the model. This fosters trust and enables well-informed clinical decision-making by enabling medical professionals to comprehend which areas of the X-ray pictures influenced the model's judgment.

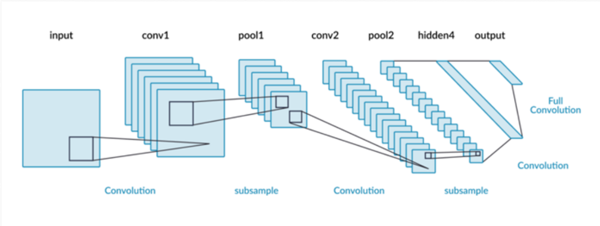


Figure 2: CNN Architecture

**Convolutional Layers:** The core of any CNN is the convolutional layer. In order to extract particular features like edges, corners, and textures from the input image, filters (also known as kernels) are used in this layer. In a technique known as convolution, these filters move across the image, scanning each pixel and calculating the dot product between the filter and the image patch at each spot. A feature map, which shows the existence of specific features in the image, is the end product of this procedure. Convolutional layers' capacity to identify spatial hierarchies in the data is one of its main advantages. While deeper layers record more intricate patterns (such forms and colors), the earlier convolutional layers concentrate on low-level data.

**Activation Functions:** Following convolution, an activation function is used to the feature maps. Usually, a threshold operation is applied via the ReLU (Rectified Linear Unit) activation function, which substitutes zeros for all negative values in the feature map. This gives the network non-linearity, which enables it to learn more intricate tasks. ReLU is popular because it expedites training and helps avoid the vanishing gradient issue.

**Pooling Layers:** In order to lessen the computational load and avoid overfitting, the pooling layer is employed to shrink the spatial dimensions of the feature maps. Max pooling is the most popular kind of pooling, in which the network chooses the maximum value within a short window (often 2x2 or 3x3). This downsampling operation preserves the most important features while reducing the size of the data passed to the next layers. Pooling helps the model become more translation-invariant, meaning it can recognize features in the image regardless of their position.

**Fully Connected Layers:** The output is usually flattened into a 1D vector and sent through one or more fully connected layers following the extraction and downsampling of features by the convolutional and pooling layers. These layers are made up of neurons that are connected to every other neuron in the layer above, much like classic neural networks. Based on the patterns the network has discovered, the fully connected layers process the data and learn to categorize them into distinct groups. The final fully connected layer usually ends with an activation function like SoftMax or sigmoid to produce the output probabilities for classification.

**Output Layer:** The final predictions are generated by the output layer. The last layer employs a sigmoid activation function for binary classification tasks (such as fractured vs. non-fractured), which yields a value between 0 and 1 that indicates the likelihood that the image belongs to the positive class. A SoftMax activation function, which generates a probability distribution across all potential classes, is utilized for multi-class classification applications.

### 4.1.1 BASIC CNN MODEL FOR HUMERUS BONE FRACTURE CLASSIFICATION

In order to establish a trustworthy tool for recognizing fractures from X-ray pictures, a simple Convolutional Neural Network (CNN) model for the classification of humerus bone fractures was created. Several layers that gradually harvest and process characteristics from the input images make up the CNN model architecture. In order to identify structural patterns in bone pictures, the design featured convolutional layers with 3x3 filters, which are crucial for capturing low-level data like edges and textures. In order to minimize computational complexity and ensure that the model concentrated on the most important features, max-pooling layers were applied after these layers to shrink the spatial dimensions of the feature maps. A dropout layer was employed to avoid overfitting, and fully connected layers processed the extracted features after the feature extraction layers. In order to get a probability score that indicated whether the image depicted a fractured or non-fractured bone, the last layer used a sigmoid activation function.

### 4.1.2 Training the Model

Images of both fractured and non-fractured humerus bones were used to train the model; the dataset was divided into training and validation sets. Binary cross-entropy loss and the Adam optimizer, which are common for binary classification applications, were employed during training. Using ImageDataGenerator for batch loading and real-time image augmentation, the training was carried out over ten epochs. By learning from multiple image transformations, this technique improved the model's capacity to generalize in a variety of situations. The model successfully learned from the training data, as evidenced by its training accuracy of 0.75 at the end of the training phase. However, the model's performance on the validation set was not that strong, indicating potential challenges in generalization.

### 4.1.3 Model Evaluation and Performance

Following training, the model's performance on unknown data was assessed using a different test set. In comparison to the training accuracy, the test accuracy was measured at 0.51, indicating a considerable decline. This discrepancy raised the possibility that the model was overfitting to the training set, which would explain why it was unable to generalize successfully to new samples while having learned patterns from the training set. The forecasts offered insightful information on the behavior of the model, even with the reduced test accuracy. By applying a threshold of 0.5 to the output probability from the sigmoid activation function, the model was able to classify images as either fractured or non-fractured. The results laid the groundwork for future advancements even though the forecast accuracy was lower than anticipated.

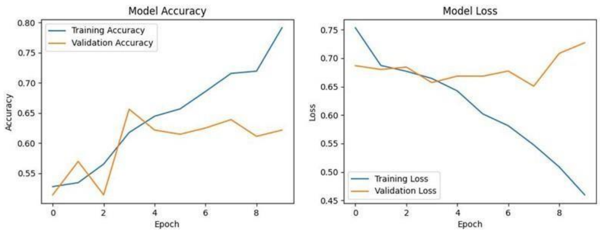


Figure 3: CNN accuracy

**Areas for Improvement** The need for better model generalization is shown by the decline in performance between the training and test data. Data augmentation, which can help expand the range of training instances, and hyperparameter tweaking, which can identify the best settings for learning rate and batch size, are two possible approaches for improving the model's performance. The model's capacity to generalize may also be enhanced by experimenting with more sophisticated CNN architectures, such as deeper networks or transfer learning. These enhancements would boost the model's accuracy and resilience, especially when it comes to forecasting bone fractures in practical situations.

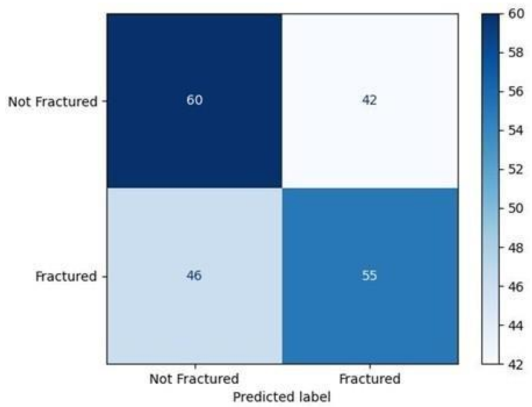


Figure 4: Confusion Matrix CNN

### 4.1.4 Conclusion

In summary, with an accuracy of 0.75 during training, the basic CNN model for classifying humerus bone fractures shown encouraging results. The test accuracy of 0.51, however, showed that the model had trouble extrapolating to new data. This performance emphasizes how important it is to optimize and improve the model. The model is a good place to start, but in order to increase its overall accuracy, more research into more complex architectures, improved data augmentation techniques, and hyperparameter tuning are required. This work shows how deep learning can be used to classify medical images and offers insightful information about how these technologies may be improved for efficient bone fracture detection in clinical settings.

| Class | Precision | Recall | F1-Score | Support |
|---|---|---|---|---|
| Not Fractured | 0.57 | 0.59 | 0.58 | 102 |
| Fractured | 0.57 | 0.54 | 0.56 | 101 |
| **Accuracy** | | | **0.57** | **203** |
| **Macro Avg** | 0.57 | 0.57 | 0.57 | 203 |
| **Weighted Avg** | 0.57 | 0.57 | 0.57 | 203 |

Figure 5: CNN results

## 4.2 VGG16 Architecture

Because of its depth and ease of usage, the VGG16 deep Convolutional Neural Network (CNN) is frequently utilized for image categorization applications. The 16-layer model bears the name of the University of Oxford's Visual Geometry Group (VGG). In order to gradually extract and process features from input images, the architecture mainly consists of three fully connected layers and thirteen convolutional layers.

The usage of tiny 3x3 convolution filters is one of VGG16's primary features. By stacking them in numerous layers, these tiny filters enable the network to learn complex features despite their simplicity. The network's ability to identify increasingly intricate patterns increases as the input moves through the layers. The spatial size of the feature maps is decreased by using max-pooling layers in between convolutional layers. This downsampling technique makes the model more efficient by lowering the computing load and guaranteeing that it concentrates on the most important features of the picture.

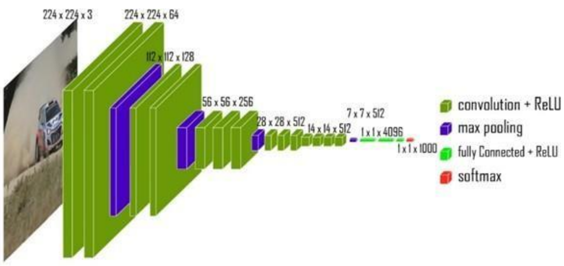


Figure 6: VGG16 Architecture

The first layer of the design is the input layer, which takes 224x224 pixel images with three RGB color channels. As the layers get deeper, the convolutional processes in the next layers use 3x3 filters to identify characteristics like edges, textures, and increasingly intricate patterns. The depth of the network is crucial because it enables hierarchical feature learning, in which the model learns low-level information in the first layers and progressively learns more abstract, higher-level features in the deeper layers. Max-pooling layers are used to downsample the feature maps after multiple convolutional layers, lowering their computational cost and spatial dimensions while maintaining the key characteristics. Two fully linked layers make up the network after the convolutional and pooling layers. To create a representation of the image, these layers further process the features that were retrieved. A SoftMax layer generates the model's final output by allocating class probabilities, such as determining whether a picture contains a dog, cat, or other items.

VGG16's deep architecture, which enables it to extract intricate, rich features from photos, is responsible for its success. Small, constant filters are used throughout the model to achieve great performance without sacrificing simplicity. Furthermore, VGG16 is a great candidate for transfer learning because it has already been pre-trained on huge datasets like ImageNet. By using pre-training, the model may take advantage of the knowledge gathered from large amounts of image data. This knowledge can then be adjusted for certain tasks using smaller datasets, saving time and computing power. VGG16 is a commonly used architecture for a variety of image classification tasks due to its depth, simplicity, and transfer learning capabilities.

To sum up, VGG16 is a strong and effective model that is renowned for its capacity to extract intricate features from pictures. It is extremely adaptable to different jobs because to its simple architecture, which consists of deep layers and small filters. The model is a dependable option in the field of computer vision because it is available as a pre-trained model that can be adjusted for certain applications, thereby enhancing its efficacy.

### 4.2.1 Data Preparation and Image Pre processing

Loading the photos, which are arranged into folders based on their respective classes—such as fractured and non-fractured—is the initial stage in training the model. An ImageDataGenerator, a program that facilitates the fast loading and preprocessing of picture data for training, is used to process these folders. Among the many tasks carried out by the ImageDataGenerator are resizing the photos to a uniform size (128x128 in this instance) and rescaling the pixel values to fall between 0 and 1. By standardizing the input data, rescaling helps the neural network train more efficiently and avoid problems where specific pixel values could control the learning process.

### 4.2.2 Model Architecture and Transfer Learning

The foundation of the model is a pre-trained VGG16 architecture. With the help of a sizable dataset (ImageNet), the deep convolutional neural network VGG16 was pre-trained to identify fundamental aspects of images, such as edges and textures. The pre-trained VGG16 model is utilized rather than training the model from scratch, which would necessitate a sizable dataset and substantial processing power. This model's layers are frozen, which means that throughout training, their weights are not changed. As a result, the model may preserve the feature extraction skills it acquired from ImageNet without having to recalculate them.

Custom layers are added to the VGG16 base model in order to customize it for the particular goal of classifying bone fractures. In order to convert the 2D feature maps into a 1D vector that the fully connected layers can process, a Flatten layer is first applied. A Dropout layer is then added after a dense layer with 128 units. By randomly setting a portion of the inputs to zero during training, the Dropout layer helps to prevent overfitting and regularizes the model. A probability between 0 and 1 is output by the last layer, a Dense layer with a sigmoid activation function, indicating the possibility that the image is part of the fractured class.

### 4.2.3 Model Training and Optimization

The Adam optimizer, a well-liked option for deep learning model training because of its variable learning rate capabilities, is used to build the model during the training phase. To aid in the model's more effective convergence, the optimizer modifies the learning rate during training. Binary cross-entropy, a loss function frequently employed for binary classification problems such as this one, is employed. The prepared dataset is then used to train the model. The features (pictures) are then fed through the network, and the loss is determined by comparing the predictions to the actual labels. To reduce this loss, the optimizer modifies the model's weights.

A validation set is used to track the training process and evaluate the model's capacity to generalize to new data. Instead of knowing the training data by heart, the model may be learning significant patterns if it performs well on the validation set and attains a high accuracy.

### 4.2.4 Model Evaluation and Predictions

Predictions are made on the test set after the model has been trained and saved for further use. The photos in the test set were not seen by the model during training. These photos are loaded and preprocessed once more using the ImageDataGenerator; however, since the objective is to predict each image's class using the learnt characteristics, no data augmentation is employed this time.

In order to generate predictions, the model outputs a probability for every image, indicating the likelihood that the image is part of the broken class. After that, a threshold of 0.5 is applied when converting these probabilities into binary class labels. The image is categorized as fractured if the estimated probability is higher than 0.5; if not, it is classified as non-fractured.

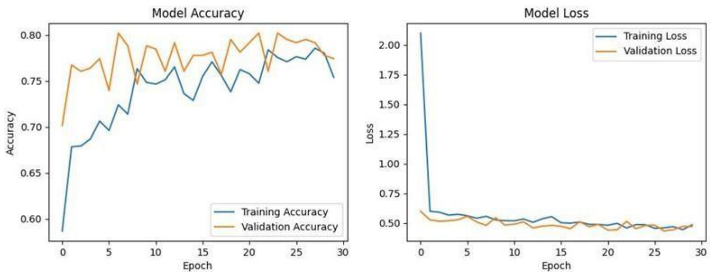


Figure 7: Enter Caption

### 4.2.5 Accuracy Calculation and Visualization

By contrasting the predicted labels with the actual labels of the test images, the model's accuracy is determined. On the training set, the model's accuracy is approximately 0.70, and on the test set, it is approximately 0.45. This suggests that although the model does very well during training, it has to be improved in order to generalize to new, unseen data. The model may be overfitting, which occurs when it learns particular characteristics of the training data that don't transfer well to other data, as indicated by the discrepancy between training and test accuracy.

Increasing the amount of the training dataset, adding data augmentation (such as random rotations, zooming, and flipping), fine-tuning the model, or experimenting with new architectures and hyperparameters are some methods that can be used to increase accuracy. The model can more accurately identify bone fractures in test photos that are not visible and make more accurate predictions by enhancing its generalization ability.

A number of test photos are shown with their projected labels in order to evaluate the model's performance in more detail. This makes it possible to quickly assess the model's accuracy in classifying the photos and gives a visual representation of its predictions.

Finding places where the model could be having trouble, such as with extremely difficult or confusing cases, is made easier by looking at the predictions for several test photos.

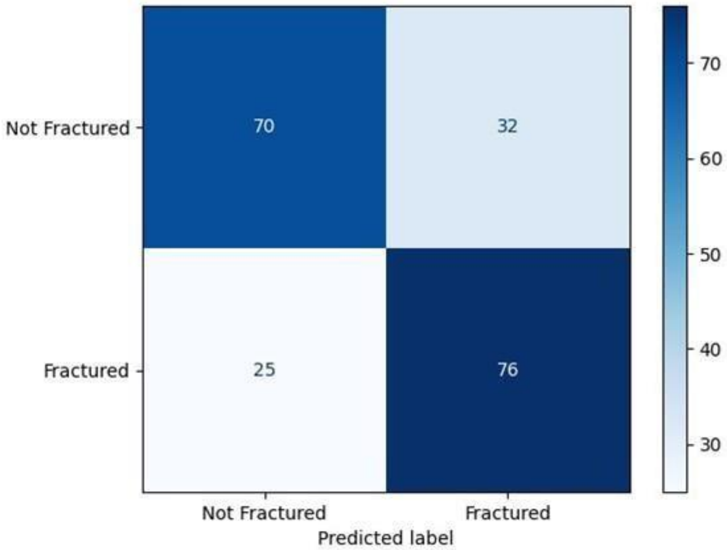


Figure 8: Enter Caption

### 4.2.6 Conclusion

Through the use of transfer learning with VGG16, the model may be trained on picture data, taught to diagnose bone fractures, and then predict new, unseen images. Pre-processing methods like resizing and rescaling guarantee that the data is in the right format for the model, and using a pre-trained model for transfer learning enhances performance and lessens the requirement for a sizable, labeled dataset. Following training, the model's predictions are assessed using accuracy computations and visualizations to determine their efficacy and pinpoint areas in need of development. The model's performance can be greatly enhanced with more refinement, increasing its accuracy and resilience for practical uses.

| Class | Precision | Recall | F1-Score | Support |
|---|---|---|---|---|
| Not Fractured | 0.64 | 0.74 | 0.68 | 102 |
| Fractured | 0.69 | 0.58 | 0.63 | 101 |
| **Accuracy** | | | **0.66** | **203** |
| **Macro Avg** | 0.66 | 0.66 | 0.66 | 203 |
| **Weighted Avg** | 0.66 | 0.66 | 0.66 | 203 |

Figure 9: Enter Caption

## 4.3 DENSENET121 and DENSENET169 COMBINED ARCHITECTURE

A sophisticated Convolutional Neural Network (CNN) with the goal of improving efficiency and feature reuse is the Densenet169 architecture. 169 layers make up this system, as indicated by its name, ”Densenet169.” Unlike other CNN architectures, such as VGG16, this model has a unique design. The usage of ”dense blocks,” which are the fundamental building elements of the architecture, is one of Densenet169’s main inventions. The layers are physically connected to each other within each dense block. This implies that every layer transmits its feature maps to every layer after it in addition to receiving input from the layer before it. Consequently, DenseNet121 and DenseNet169 establish a highly interconnected network with considerable feature map sharing, greatly enhancing feature efficiency and reuse.

The utilization of bottleneck layers inside the dense blocks is one of Densenet121’s primary characteristics. The purpose of these bottleneck layers, which are made up of 1x1 filters or convolutions, is to lower the quantity of features that are sent on to the following layers. This helps to improve the computational efficiency of the model and regulate its size. Transition layers are used in between the thick blocks. In order to decrease the size of the feature map and the number of channels, these layers use 1x1 convolutions in conjunction with 2x2 average pooling. This lowers the computational cost and memory use while preserving a high degree of performance.

Densenet121 uses smaller filters, namely 3x3 convolutions, in a manner akin to that of VGG16. The inclusion of dense connections, which promote better gradient flow and more efficient feature reuse and enhance model training, is the main distinction, though. A global average pooling layer condenses the feature maps into a single vector per feature map once they have traversed the last dense block. An output layer, usually a SoftMax layer for multi-class classification jobs, and a fully linked layer come next.

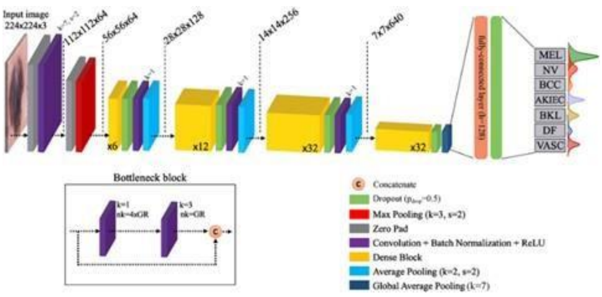


Figure 10: Enter Caption

Densenet121 has several benefits, especially when it comes to feature reuse. More varied and in-depth feature learning results from the thick connections, which enable each layer to access and expand upon features from every earlier layer. By reusing features instead of learning new ones from start, this approach lowers duplication and makes the model more economical in terms of parameter usage when compared to similarly deep networks like VGG16. Furthermore, the vanishing gradient issue is lessened by the dense connections, which enables the model to train efficiently even with extremely deep architectures. Densenet121 is a strong contender for transfer learning since it also takes advantage of the availability of pre-trained models on huge datasets like ImageNet. The model’s performance and efficiency can be enhanced when used on smaller datasets because only the higher layers need to be modified to fit new jobs. As a result, Densenet121 is an extremely effective and scalable architecture that performs exceptionally well in applications involving image recognition and deep learning. It can perform better than many conventional CNN models thanks to its combination of dense connections, bottleneck layers, and transition layers, particularly when jobs requiring transfer learning or big datasets are involved.

### 4.3.1 BODY PART CLASSIFICATION AND FRACTURE CLASSIFICATION USING DEEP LEARNING

In order to detect bodily parts like the humerus and shoulder and subsequently classify them as fractured or non-fractured, our project's goal is to create a deep learning model for medical picture classification. The method is based on transfer learning, in which new models are created for the particular tasks at hand and pre-trained models, such as Densenet121, are used to extract features.

**Body Part Classification** Making distinctions between various bodily components is the initial stage in the categorization process. In this instance, we concentrate on recognizing the humerus and the shoulder as two distinct bones. At this point, transfer learning is crucial. As a feature extractor, a pre-trained model such as Densenet121 is employed, which has been trained on huge datasets like ImageNet. Because it has been trained on millions of images, this model already knows about general image properties like edges, textures, and forms. We save a significant amount of time and computing resources by not having to train a model from scratch when we use Densenet121. The Densenet121 model's layers will be frozen, keeping the previously learnt features, and it will be modified to fit our particular job by adding a few custom layers. These custom layers are responsible for learning the final classification — whether an image is of a humerus or a shoulder.

A collection of photos with the labels ”shoulder” and ”humerus” are used to train the body part categorization model. With a sigmoid activation function in the output layer that produces values between 0 and 1, signifying the likelihood of each class, the model is trained using a binary classification technique. Dropout is used in the custom layers to prevent overfitting and make sure the model performs effectively when applied to unseen data. By comparing the predictions with the ground truth labels, accuracy serves as the main criterion for evaluating the model.

**Fracture Classification**

Following training, the body part classification model forms the basis for the subsequent stage of the project, which is the categorization of fractures. This step's objective is to categorize the photos as either fractured or non-fractured according to the body portion (shoulder or humerus) that was identified in the step before. This method is similar to the idea of fine-tuning, in which the previously learned model is further modified for a different but related goal. This involves adding further layers to handle the fracture classification task after the pre-trained body part model has been frozen (the layers remain unaltered). In particular, the new model incorporates fully connected, dropout, and batch normalizing layers, all of which enhance the model's generalization and performance.

Two nodes in the output layer, each with a SoftMax activation function, offer the likelihood that the image falls into the fractured or non-fractured class. The model's performance is further enhanced by employing sophisticated data augmentation techniques. These methods employ changes including rotating, shifting, zooming, and flipping to the training images. This keeps the algorithm from learning particular image details (a process called overfitting) and helps to artificially grow the dataset. The algorithm is trained to identify fractures from various viewpoints, lighting scenarios, and orientations by adding these variables to the training data.

For multi-class classification problems, the model is constructed with a loss function similar to categorical cross-entropy and a lower learning rate. The Adam optimizer, which is renowned for its effective management of sparse gradients and adaptive learning rates, is used to optimize it.

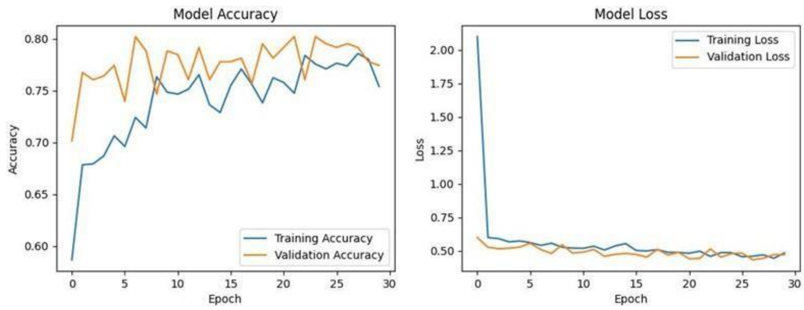


Figure 11: Enter Caption

**Training and Optimization**

Two key strategies are employed during training to maximize the model's performance. The first involves early terminating the training process if, after a predetermined number of epochs, the validation performance does not improve. This conserves processing resources and helps avoid overfitting. When the model's performance no longer improves during training, the second method modifies the learning rate. As a result, the model converges more quickly and stays out of less-than-ideal solutions.

A validation set is used to track the model's performance during training, and a training set containing different body part images and fracture labels is used to train the model over several epochs. Following training, the model is stored for later use, and its effectiveness is assessed using test data that has not yet been viewed.

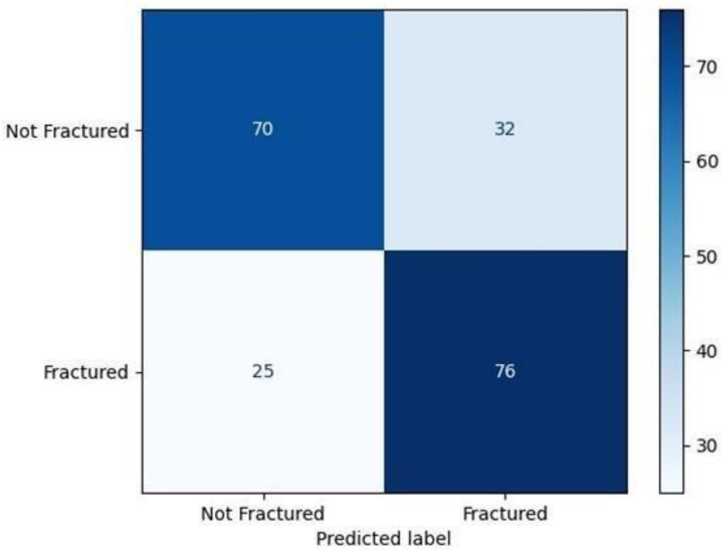


Figure 12: Enter Caption

### 4.3.2 Conclusion

This two-stage classification pipeline effectively combines the power of transfer learning and fine-tuning to create a robust solution for medical image classification. By leveraging pre-trained models like Densenet121, the system can accurately classify medical images based on both body part type and fracture status. The use of data augmentation, regularization techniques like dropout, and performance optimization strategies ensures that the model generalizes well and performs efficiently across a range of test cases. This approach can be extended to other medical image classification tasks, providing a solid framework for the development of automated diagnostic tools.

| Class | Precision | Recall | F1-Score | Support |
|---|---|---|---|---|
| Not Fractured | 0.74 | 0.69 | 0.71 | 102 |
| Fractured | 0.70 | 0.75 | 0.73 | 101 |
| **Accuracy** | | | **0.72** | **203** |
| **Macro Avg** | 0.72 | 0.72 | 0.72 | 203 |
| **Weighted Avg** | 0.72 | 0.72 | 0.72 | 203 |

Figure 13: Enter Caption

**Key Concepts** Several cutting-edge deep learning ideas were used in this research to enhance model performance and guarantee precise predictions. We were able to avoid starting from scratch by using pre-trained models thanks to transfer learning. The sigmoid activation function, which produced probabilities for each class, was essential for the binary classification of body parts. A probability distribution over the fracture and non-fracture categories was provided by the SoftMax activation, which was utilized to manage multi-class classification for fracture detection.

By using dropout as a regularization strategy, the model was able to avoid being overly dependent on any one feature, which improved its ability to generalize. The model becomes more resilient to unseen examples by using data augmentation to better handle data variances like changes in lighting or orientation. Batch normalization, which reduces the internal correlate shift and enables faster learning, was used to stabilize training and enhance convergence.

Lastly, optimization techniques such as the Adam optimizer were employed for efficient gradient descent and learning rate adjustments, allowing the model to converge more quickly while avoiding issues like vanishing or exploding gradients. These concepts, combined with careful training strategies, made it possible to build a powerful and effective system for medical image classification.

# 5 Conclusion and Future Work

## 5.1 Future Work

This study lays the foundation for the automated classification of humerus and shoulder fractures using deep learning techniques, while there are a number of interesting avenues for future research to expand the project's scope and improve its outcomes.

It would be a significant stretch to classify fractures in bone places other than the humerus and shoulder. If the model includes more bones, like the femur, tibia, or spine, it might be more useful to a larger spectrum of medical circumstances. Additionally, examining state-of-the-art imaging methods like CT or 3D scans could enable a more thorough and precise diagnosis of fractures. Richer information regarding bone alignment and structure is provided by several modalities, which enables the model to identify intricate dislocations or little fractures that might not be apparent in 2D X-rays. However, issues like higher computational complexity and bigger dataset requirements will need to be addressed in order to integrate various modalities.

Addressing the problems of overfitting and computational limitations found in this investigation is another possible enhancement. Advanced methods like regularization, dropout layers, or significant data augmentation could help reduce overfitting, a situation in which the model performs well on training data but badly on unseen data. Additionally, exploring lightweight neural architectures optimized for resource-constrained environments could address computational limitations, enabling the deployment of these models on portable devices or edge computing platforms for real-time use in rural clinics and emergency settings.

Then, ensemble models could be investigated in greater detail in subsequent iterations of this study. By combining the advantages of several models, ensemble approaches provide a system that is more resilient and broadly applicable. We have already mastered the best-performing models, so we may combine them to improve the outcome to even higher prospects. Currently, we worked with different models to understand the usefulness of the model in its standalone state.

## 5.2 Future Work

This research successfully demonstrates the potential of deep learning-based models in addressing the critical challenge of bone fracture detection, focusing on the humerus and shoulder bones. By leveraging advanced convolutional neural networks (CNNs) and transfer learning techniques, this study has provided an automated system capable of accurately classifying fractures and healthy bones from X-ray images. Using the publicly available MURA dataset, the project not only achieved a commendable accuracy but also laid the groundwork for expanding similar methodologies to other skeletal regions.

A key highlight of this research is its two-tier classification approach: first identifying the type of bone, and then distinguishing between fractured and non-fractured states. Models like DenseNet169 and custom-built CNNs were tested and evaluated rigorously, with DenseNet121 and DenseNet169 emerging as the most effective, achieving the highest accuracy for fracture classification, and for bone type model MobileNetV2 performs the best as it is a lightweight task. Hence once the research has been developed to achieve higher accuracies with various ensemble models with all the bone parts covered and trained, the system would be ready for a real time testing in the future in a more supportive fashion.

# 6 Reference


*[1] S. S. Sinthura, Y. Prathyusha, K. Harini, Y. Pranusha and B. Poojitha, "Bone Fracture Detection System using CNN Algorithm," 2019 International Conference on Intelligent Computing and Control Systems (ICCS), Madurai, India, 2019, pp. 545-549, doi: 10.1109/ICCS45141.2019.9065305.*

*[2] Bone Fracture Detection Using CNN, Sai Prudhvi Vallurupalli and T. Anuradha,E3S Web Conf.Volume 409, 2023, International Conference on Management Science and Engineering Management (ICMSEM 2023) https://doi.org/10.1051/e3sconf/202340902015*

*[3]N. T. Htun and K. Mo Mo Tun, ”Fuzzy-based Image Enhancement and Ensemble CNN Model for Bone Fracture Detection and Classification System,” 2024 5th International Conference on Advanced Information Technologies (ICAIT), Yangon, Myanmar, 2024, pp. 1-6, doi: 10.1109/ICAIT65209.2024.10754918.*

*[4] Analysis and Classification of Bone Fractures Using Ma chine Learning Techniques, Ayesha Noureen1, M Azam Zia1 , Ahmad Adnan1, and M Hashim2, Department of Computer Science, University of Agriculture Faisalabad, Faisalabad, Pakistan,E3S Web Conf.Volume 409, 2023International Conference on Management Science and Engineering Management (ICMSEM 2023)*

*[5]D. P. Yadav and S. Rathor, ”Bone Fracture Detection and Classification using Deep Learning Approach,” 2020 International Conference on Power Electronics and IoT Applications in Renewable Energy and its Control (PARC), Mathura, India, 2020, pp. 282-285, doi: 10.1109/PARC49193.2020.236611.*

*[6]R. Bhuria and S. Gupta, ”X-Ray Insights: Comprehensive Dataset for Bone Fracture Detection Across Diverse Anatomical Regions,” 2024 5th International Conference on Smart Electronics and Communication (ICOSEC), Trichy, India, 2024, pp. 1242-1247, doi: 10.1109/ICOSEC61587.2024.10722406.*

*[7]S. Chauhan, ”Bone Fracture Detection with CNN: A Deep Learning Approach,” 2024 5th International Conference on Smart Electronics and Communication (ICOSEC), Trichy, India, 2024, pp. 1253-1258, doi: 10.1109/ICOSEC61587.2024.10722699.*

*[8] K. N. K. Reddy and V. Cutsuridis, ”Deep Convolutional Neural Networks with Transfer Learning for Bone Fracture Recognition using Small Exemplar Image Datasets,” 2023 IEEE International Conference on Acoustics, Speech, and Signal Processing Workshops (ICASSPW), Rhodes Island, Greece, 2023, pp. 1-5, doi: 10.1109/ICASSPW59220.2023.10193015.*

*[9]S. S. Tabarestani, A. Aghagolzadeh and M. Ezoji, ”Bone Fracture Detection and Localization on MURA Database Using Faster-RCNN,” 2021 7th International Conference on Signal Processing and Intelligent Systems (ICSPIS), Tehran, Iran, Islamic Republic of, 2021, pp. 1-6, doi: 10.1109/ICSPIS54653.2021.9729393.*

*[10]K. Barua et al., ”Explainable AI-Based Humerus Fracture Detection and Classification from X-Ray Images,” 2023 26th International Conference on Computer and Information Technology (ICCIT), Cox’s Bazar, Bangladesh, 2023, pp. 1-6, doi: 10.1109/ICCIT60459.2023.10441124.*

*[11]R. S. Upadhyay and P. Tanwar, ”A Review on Bone Fracture Detection Techniques using Image Processing,” 2019 International Conference on Intelligent Computing and Control Systems (ICCS), India, 2019, pp. 287-292, doi:10.1109/ICCS45141.2019.9065874.*

*[12]S. Kaur and J. Singh, ”Enhanced Detection of Osteoporosis in Femur Bones with Improved VGG16,” 2024 International Conference on Communication, Computer Sciences and Engineering (IC3SE), Gautam Buddha Nagar, India, 2024, pp. 1341-1345, doi: 10.1109/IC3SE62002.2024.10592883.*