SSRG International Journal of Electronics and Communication Engineering
ISSN: 2348-8549/ https://doi.org/10.14445/23488549/IJECE-V11I12P120

Volume 11 Issue 12, 219-229, December 2024

Original Article

Revolutionizing Bone Fracture Diagnosis: A Deep Learning Approach to X-Ray Image Analysis

A. M. Linchu[1], B. Ben Sujitha[2]

[1,2]*Department of Computer Science and Engineering, Noorul Islam Centre for Higher Education, Tamil Nadu, India.*

[1]*Corresponding Author : linchubinoy84@gmail.com*

Received: 25 October 2024 Revised: 29 November 2024 Accepted: 14 December 2024 Published: 30 December 2024

Abstract *- Bones are the very important part of the human body, which provides body structure and protection for the internal organs. A bone fracture is a widespread scenario for the human body, which can lead to serious complications. Misdiagnosis of fractures is the most common mistake, resulting in treatment delays and permanent impairment. So, timely and accurate fracture detection is critical for proper treatment planning and medical diagnosis. X-ray imaging is a widely used diagnostic tool since manual interpretation is prone to errors. This study proposes an AI enhanced bone detection framework utilizing a bone fracture dataset consisting of 9463 X-ray images of fractured and non-fractured cases. Different preprocessing and data augmentation techniques played a major role in improving the dataset diversity and generalizability. The proposed methodology employs ResNet 50 for the feature extraction, enhancing it with the Bottleneck Attention Module (BAM) with dual attention strategies to refine critical features for effective fracture detection. With an accuracy of 97%, 96.12% precision, recall of 96.70%, and 96.38% F1 score, the suggested model outperformed other models like YOLOv8, Ensemble Model, ResNet50-DenseNet 121, and CNN. The results demonstrated that with improved feature representation and accuracy in bone fracture detection, the proposed model exhibits a valuable tool for enhanced patient care through early intervention and accurate fracture diagnosis.*

Keywords *- Bone fracture, X-ray images, Medical diagnosis, Deep learning, Attention mechanism.*

1. Introduction

The skeletal system is the dynamic organ that shapes the human body, allows motor function and locomotion, creates marrow-derived cells, enables respiration, cares for vital organs, and plays a key role in homeostasis [1]. The bones undergo constant changes in the human body that remodel according to the ever-changing atmosphere. Normally, the human body consists of 206 bones. These bones are classified according to their shapes as short bones, long bones, irregular bones, flat bones, and sesamoid bones [2]. The skeletal system's structural support is provided by the long bones, like the femur and thigh bones.

Tarsal in the foot, carpals in the hand, and ankle and wrist bones are short bones that are wider but not so long. The flat bones, like ribs, provide a flat surface for the muscle to attach. Vertebrae from our spinal column and bones that make up our faces are oddly shaped and are examples of irregular bones. Next, the sesamoid bones are rooted with the ligament where two bones connect. These bones increase joint efficiency, and the patella or kneecap is a suitable example of the sesamoid bones [3].

Fracture, the discontinuity of a bone, is classified into five main categories. The first category is the fracture that can be identified based on the relationship with the external environment [4]. Normal bones can withstand some considerable amount of force, and when a fracture is sustained due to excessive force, such as a fall, road accident, or fight, that causes trauma and can be called a traumatic fracture. Pathological fractures occur due to some underlying medical conditions.

A stress fracture represents a fracture sustained due to serious repetitive injury or stress that causes breakage of the trabecular bone [5]. The second category depends on the displacements, which refer to an abnormal positioning of the bone. The displacement fracture was composed of displaced, angulated, and rotated types. The third category, a fracture with a break in the overlying skin and soft tissues, depends on the external environment, known as an external fracture or an open fracture. The fourth category is based on the complexity of treatment, such as simple or compound fracture. Finally, the fractures are categorized based on the patterns such as transverse, spiral, oblique, segmental, and comminuted.

The simplest and fastest method for diagnosing fractures is examination and X-ray imaging. A Magnetic Resonance Imaging (MRI), Computed Tomography (CT) scan, or bone scan can be done in special scenarios. If the fractures are not properly diagnosed, they can seriously affect the human body. The misdiagnosed fractures cause internal bleeding due to the damage in the blood vessels and sometimes lead to fat embolism that causes blood vessel blockage and loss of functions [6]. To overcome the misdiagnosis of fractures, integrating Artificial Intelligence (AI) can identify hidden or complex features, assist radiologists in diagnosing fractures accurately and quickly, and represent significant advancements in medical imaging technology. Thus, a Deep Learning (DL)-based model is suggested to detect bone fractures from X-ray images effectively. The main contributions of the study are given below:

- To propose an effective deep learning-based model with an attention mechanism for detecting bone fractures from X-ray images.
- To compare the effectiveness of the suggested model with existing models.
- The efficiency can be evaluated using accuracy, precision, recall, and F1-score metrics.

The remaining part of the paper is structured as follows: Section 2 provides a comprehensive literature review emphasizing the need for the current research. Section 3 details the methodology and deep learning model architecture for effectively detecting fractures. Section 4 offers the results and discussion, highlighting the potential of the suggested model. Finally, Section 5 concludes the paper by analyzing the major contributions.

2. Related Works

Alshahrani and Alsairafi [7] investigated the YOLOv8 model with its real-time object detection and image segmentation capabilities. The FracAtlas dataset, which had fractured and non-fractured bone images, was utilized. The classification capabilities of the YOLOv8 and VGG-16 models were evaluated, and the YOLOv8 model achieved an accuracy of 81%. Tahir et al. [8] proposed an ensemble model utilizing MobileNetV2, InceptionV3, VGG16, and ResNet50. The model employed the Mura-v1.1 dataset with Histogram Equalization (HE) and global average pooling for feature extraction. The proposed ensemble model outperformed others with an accuracy of 92% and possessed challenges for the hyperparameter tuning.

Thaarakaraam et al. [9] employed the ResNet 50 model for bone fracture detection with real-time prediction feedback. The model integrated object detection techniques such as R-CNN and RetinaNet to make decision boundaries for the fractured regions. The suggested model surpassed AlexNet and VGGNet with an accuracy of 78%, but the dataset lacked various anatomical variations and clinical conditions. Ma and Luo et al. [10] proposed a two-stage system integrating crack sensitive convolutional neural networks (CrackNet) by localizing the bone regions using the Faster R-CNN and identifying fracture regions within the contour. The system achieved 90.11% accuracy by analyzing 1052 X-ray images. The model sensitivity towards fracture lines resulted in more accurate fracture identification, with limitations in the higher computational complexity and longer processing time.

Wang et al. [11] proposed a dual-stage R-CNN network named ParallelNet, where the main network concentrates on minor fractures utilizing normal convolution. In contrast, the second network adopted dilated convolution for detecting large fractures. This framework achieved 87.8% accuracy and outperformed other existing deep learning models. The main limitations included the complexity of different dilated convolution rates and multiple backbone pathways. Parvin and Rahman [12] utilized a multi-modal bone fracture dataset containing 641 images to detect fractures from multimodal images, including MRI, X-ray, and CT scans. The authors implemented real-time detection and classification of bone fractures using the YOLOv8 model with a data augmentation step. The model attained precision of 95%, recall of 93%, and mean average precision (mAP) of 92%. However, a major limitation of the model is the relatively small dataset, which limits the model's generalizability. Using CNN, Saad et al. [13] automatically detected fractures from X-ray images. The complex patterns and features were extracted from a dataset of 9103 images. The outcomes demonstrated that the model achieved 91% accuracy. The study possessed limitations in the potential of overfitting.

Dey et al. [14] introduced a hybrid model combining ResNet50 and DenseNet121 architectures to detect and classify the humerus fractures utilizing the publicly available MURA dataset of 1266 X-ray images. The study employed Contrast Limited Adaptive HE (CLAHE) in preprocessing for enhanced performance. The hybrid model achieved 93.41% accuracy and demonstrated superior performance over other pretrained transfer learning models. Moreover, the study had limitations in the dependency on the quality and diversity of the dataset. Karanam et al. [15] employed various DNN classifiers to classify the bone images between fractured and healthy. The InceptionResNetV2 model demonstrated superior performance with an accuracy of 94.58%, surpassing other classifiers. The study possessed limitations in differentiating fractures under their types, such as normal, oblique, spiral, etc. Medaramatla et al. [16] proposed a deep learning model incorporating YOLO NAS, Efficient Det, and a detection transformer. The study utilized a hybrid dataset of 4736 hand bone X-ray images, further classified into 6 classes. The outcomes demonstrated superior accuracy, with challenges in customizing the algorithms deeply. Kassem et al. [17] employed a computer-aided diagnosing-based DL model for detecting pelvic fractures. A dataset comprising 876 X-ray images was utilized with an explainable AI framework.

With high values for evaluation metrics, ResNet50, transfer learning, and GoogleNet models were used for classification. The suggested ResNet50 and transfer learning model failed to predict the pelvic fracture into three classes (normal, fractures, dislocated).

The existing studies on bone fracture detection provide several techniques with significant gaps. There are challenges in pinpointing the exact location of the fracture. The widespread use of CT is limited due to its availability, medical costs, and radiation exposure considerations, as it is not classically utilized as the primary tool in most parts of the world. The amount of data utilized for CT images is larger than that for chest X-rays. These algorithms are not globally applicable due to their complex calculations, high computational power requirements, and challenging integration into the medical examination process. Also, there are no inter-rater reliability assessments, and it suffers from varying inter-observer reliability.

3. Materials and Methods

Accurate and appropriate bone fracture detection is crucial for efficient patient care, potentially minimizes related complications, and enhances recovery times. DL can be integrated to detect bone fractures, reducing the medical staff workload and maximizing resource efficiency. So, this study proposes a DL model for efficient bone fracture detection on X-ray images by processing massive image data. A comprehensive dataset with X-ray images undergoes preprocessing and data augmentation with efficient feature extraction. The deep learning architecture with an attention mechanism is designed for real-time bone fracture detection, integrating different layers. Figure 1 illustrates the schematic block diagram of the suggested model.

3.1. Dataset

The study utilized data from the Kaggle Repository [18] consisting of fractured and non-fractured X-ray images of several joints in the upper extremities. The dataset comprises 9463 X-ray images, and this study utilized 500 images in each category, resulting in a balanced dataset. The fractured and non-fractured X-ray images are labeled as '0' and '1', respectively, for classification. Figure 2 illustrates the sample images in the dataset.

3.2. Data Preprocessing and Augmentation

Preprocessing prepares the raw data suitable for the model training [19]. This study's collected X-ray images undergo preprocessing steps, including standardization, resizing, and orientation adjustments. Resizing is done in the preprocessing step to ensure uniformity across the dataset's images. All the X-ray images are resized into 224 x 224 pixels. Data augmentation is crucial since it enlarges the dataset by applying various transformations. In this study, transformations such as rotation, flip, rescale, zoom, and shear are implemented to get the augmented data to increase the diversity and generalizability of the dataset and avoid the risk of overfitting. Both testing and training data undergo preprocessing and data augmentation.

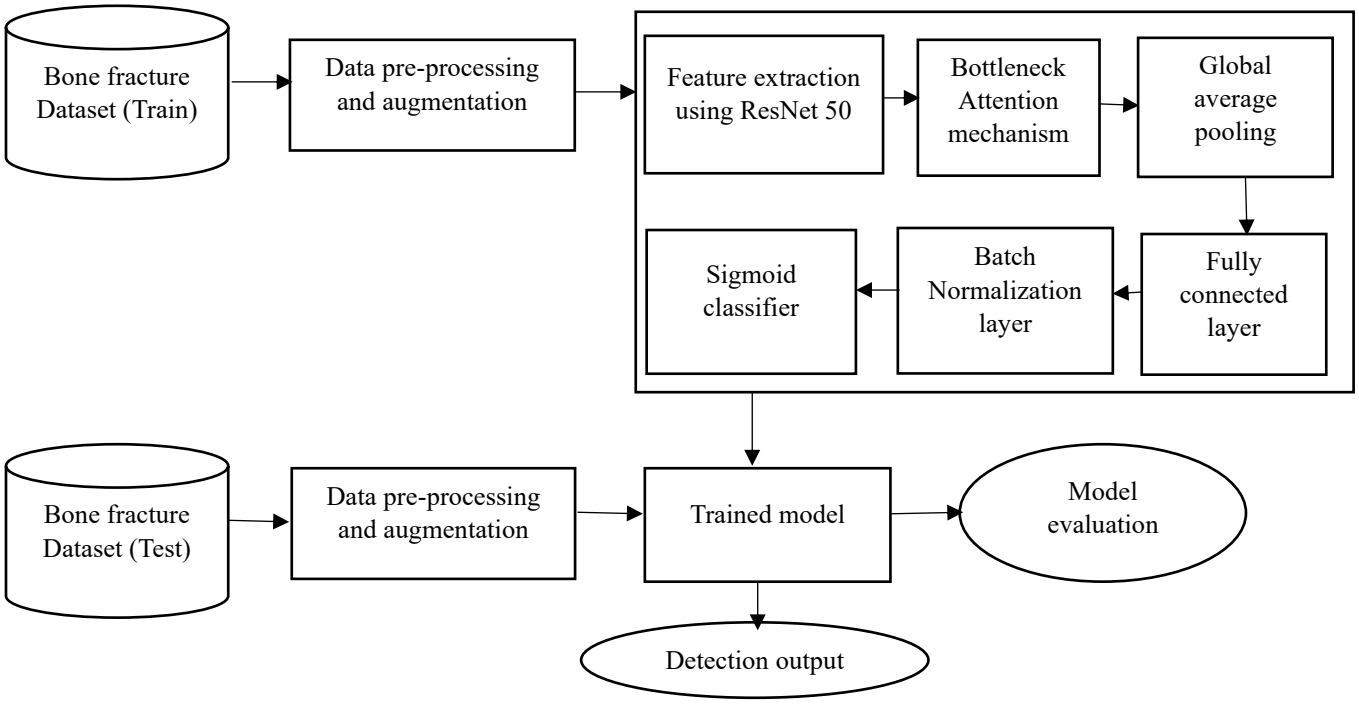

Fig. 1 Block illustration of the proposed model

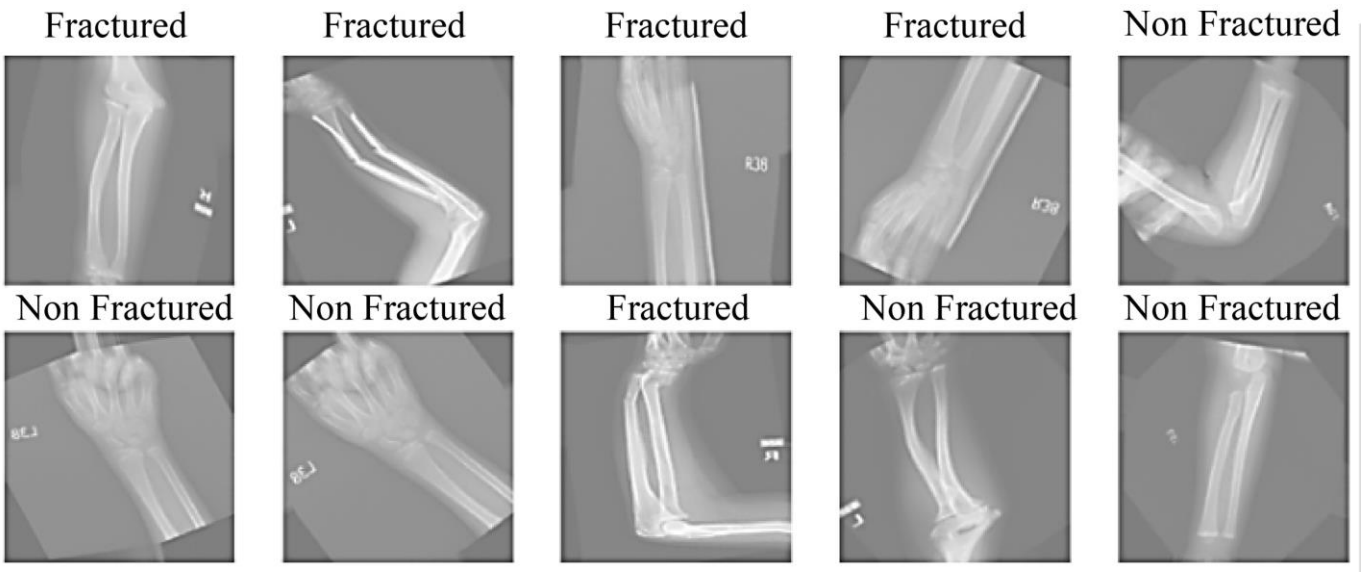

Fig. 2 Sample images in the dataset

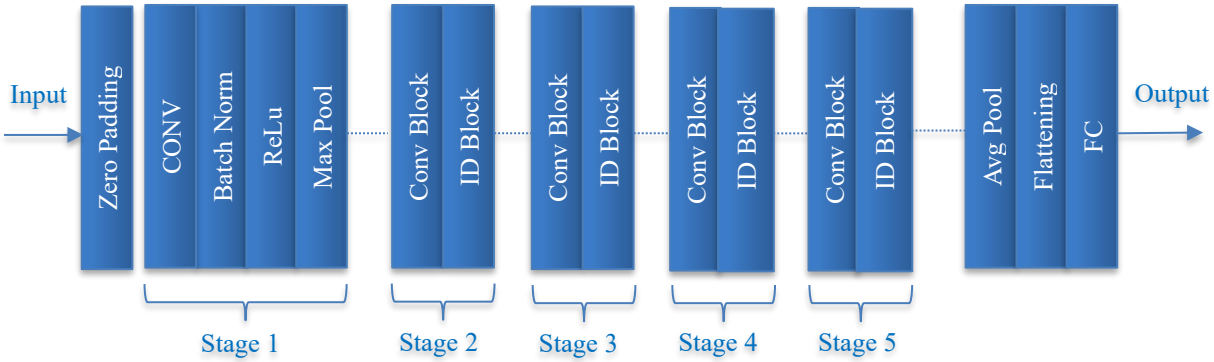

Fig. 3 Basic architecture of ResNet 50

3.3. Proposed Methodology

The study proposes a DL model with an attention mechanism for detecting bone fractures from X-ray images. ResNet 50 is employed for the feature extraction by excluding the top layers, augmented with a Bottleneck Attention Module (BAM).

This base model processes the input image of 224 x 224 x 3, and the feature map output is then fed to the BAM block. This combination enhances the model performance by focusing on important features with the addition of channel and spatial attention mechanisms.

3.3.1. Feature Extraction Using ResNet 50

ResNet 50, a CNN architecture, is a residual network (ResNet) family. This residual network is known for addressing the vanishing gradient problem [20]. The model learns the residual function to the input layers by introducing residual learning through skip connections. The basic architecture of ResNet 50 is illustrated in Figure 3. ResNet 50 comprises 50 layers, including convolution, batch normalization, ReLU activation, and fully connected layers. The main architecture is divided into four stages, each consisting of several residual blocks, and each block comprises two convolution layers followed by batch normalization and ReLU.

In the convolution block, with the convolution layer, a skip connection (identity connections) is matched with the input and output dimensions using a 1 x 1 convolution. Similarly, identity blocks directly add the input to the output without the dimension-matching convolution by preserving the information from earlier layers.

A max-pooling layer that follows downsamples the convolutional layer output. The convolution operation is mathematically expressed as in Equation 1.

$$y_{i,j,k} = \sum_{x=1}^{X} \sum_{y=1}^{Y} \sum_{z=1}^{Z} x_{i+x-1,j+y-1,z} \cdot W_{x,y,z,k} + b_k \quad (1)$$

Where, is the output at position *(i, j)* for output channel *k.* is the input at position for the input channel *l.* is the weight of the filter at position (m, n) between the input channel *l* and the output channel *k.* represents the bias factor for output channel *k* and, *M* is the height, and *N* denotes the filter's width with *L* number of input channels.

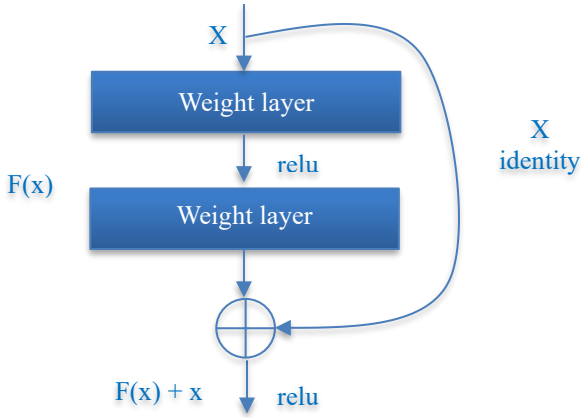

Fig. 4 Residual block in ResNet 50 model

The max-pooling layer output is then passed through a sequence of residual blocks. Figure 4 represents the residual block in ResNet 50.

The residual function is expressed by Equation 2.

$$y = \mathcal{F}(x, \{W_i\}) + x \tag{2}$$

Where y represents the block's output and x its input. is the residual function in connection with the stacked convolution layers, batch normalization, and ReLU activation. If the input and residual function have different dimensions, a 1 x 1 convolution is applied to the x to match the dimension as in Equation 3.

$$y = \mathcal{F}(x, \{W_i\}) + W_s x \tag{3}$$

Where is the weight matrix of 1 x 1 convolution for matching dimensions. Inputs to each layer are normalized to stabilize and accelerate the training process. For an input X having mini batch mean and variance, the batch normalized output is given by Equation 4.

$$\hat{x} = \frac{x - \mu_B}{\sqrt{\sigma_B^2 + \epsilon}} \tag{4}$$

Where ϵ is a constant to avoid zero division, and the learnable parameters are given by Equation 5.

$$Y = \gamma \hat{x} + \beta \tag{5}$$

Where γ and β are learnable parameters that allow the model to scale and shift the normalized output. The input of the residual block is summed with the output of the second convolutional layer to be fed through another ReLU activation function, as expressed in Equation 6.

$$ReLU(z) = max(0, z) \tag{6}$$

Where z is the input to the activation function. The last residual block output is connected to a fully connected layer, the final layer of the network, to map the output classes. The number of neurons is the same for fully connected layer and output classes. The mathematical expression of the fully connected layer for lowering the feature map to a single value is given by Equation 7.

$$y_k = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} x_{i,j,k} \tag{7}$$

Where y_k is the pooled output for channel *k*.

3.3.2. Bottleneck Attention Mechanism

The bottleneck attention mechanism operates through a dual attention strategy composed of channel and spatial attention with a bottleneck architecture, as illustrated in Figure 5, resulting in improved feature representation without significant computational overhead [21]. The channel attention mechanism emphasizes informative channels (features) by utilizing global information across the entire spatial dimension. In contrast, the spatial attention mechanism focuses on important spatial regions within the feature map.

Consider the input feature map where W, H, and C symbolize the width, height, and number of channels. The output feature map is generated by input processing using the attention module. The input feature F is initially subjected to global average pooling, resulting in a channel vector with reduced spatial dimensions, as given by Equation 8.

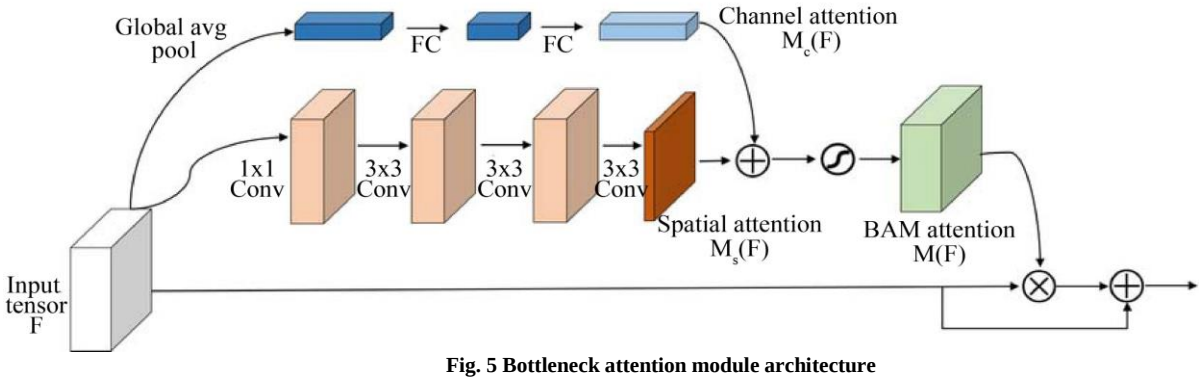

Fig. 5 Bottleneck attention module architecture

$$z_c = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} F_{i,j,c} \quad (8)$$

Where denotes the average activation for each channel c, by effectively concluding the global information for that channel. This channel descriptor is then passed through a bottleneck structure, designed with one hidden layer as a Multi-Layer Perceptron (MLP) to reduce the dimensionality. The MLP with two fully connected layers is denoted by Equation 9, where BN is the batch normalization layer to match the scale of the spatial attention output.

$$M_c(F) = BN(FC_2 RELU(FC_1(z))) \quad (9)$$

Where FC_1 reduces the channel dimension to C/r, with r being the reduction ratio and restores the channel dimension to C. Channel attention is computed using Equation 10.

$$M_c(F) = BN\left(MLP\left(AvgPool(F)\right)\right)$$

$$= BN(W_1(W_0 AvgPool(F) + b_0) + b_1) \quad (10)$$

Where $W_0 \in \mathbb{R}^{C/r \times C}, W_1 \in \mathbb{R}^{C/r \times C}, b_0 \in \mathbb{R}^{C/r}, b_1 \in \mathbb{R}^{C}$ and $M_c(F)$ having shape $C \times 1 \times 1$.

Algorithm 1: AI Enhanced Bone Fracture Detection

Input: *Xray images (fractured and non-fractured)*
Output: *Efficient bone fracture detection and classification Model.*
Begin:
Load and preprocess data:

1. *Collect dataset: D= {(), where is an X-ray image and ∈{0,1} i∈{0,1} (1: Non-fractured, 0: Fractured).*
2. *Preprocess:*
   - *Resize: →∈*
   - *Normalize:*
   - *Data Augmentation: (Rotate, Shear, Zoom, Flip, Rescaling)*

Define Base Models:
Load Models: ResNet 50 model architecture with bottleneck attention mechanism

1. ***ResNet 50****:*
   - *Input image →∈ to ResNet 50.*
   - *Feature map output:*
2. ***Bottle Attention Mechanism****:*
   - *Channel attention: Global average pooling by Equation 8, Multi-layer perceptron by Equation 9.*
     - *Final channel attention:*
   - *Spatial attention: Convolution layers with varying kernel sizes:*
   - *Combine Channel and Spatial Attention:*
     - *Attention map: +*
     - *Refined Feature Map:*
3. ***Global Average Pooling:*** *GlobalAvgPooling2D ()*
4. ***Fully Connected Layers:***
   - *Dense (1024, activation='relu') (x)*
5. ***Output layer***
   - *predictions = Dense (1, activation='sigmoid') (x)*

Model Compilation and Training:

1. *Compile each model M:*
   loss=binary_crossentropy
   optimizer=Adam ()
   metrics=[accuracy]
2. *Train: M.fit (, ,validation_data= (,))*

Model Evaluation and Comparison:

1. *Evaluate:*
   metrics=M.evaluate(,), where metrics include accuracy, precision, recall.
   Save the Model:

End

The spatial attention mechanism, on the other hand, applies a convolutional layer having a large kernel size and a specified dilation rate to capture spatial dependencies in the feature map as Equation 11.

$$M_s(F) = BN\left(f_3^{1\times1}\left(f_2^{3\times3}\left(f_1^{3\times3}\left(f_0^{1\times1}(F)\right)\right)\right)\right) \quad (11)$$

Where, $f_3^{1\times1}$, $f_2^{3\times3}$, $f_1^{3\times3}$, $f_0^{1\times1}$ are the convolution layers with varying kernel size. The output is a spatial attention map $M_s(F)$ of size $H\times W$, emphasizing or suppressing features in different spatial locations.

Thus, for a given input, BAM provides a 3D attention map $M(F) \in \mathbb{R}^{C\times H\times W}$. The refined feature map is computed as in Equation 12.

$$F' = F + F \otimes M(F) \quad (12)$$

Where $\otimes$ denotes the element wise multiplication. The computed $M_c(F) \in \mathbb{R}^C$ and $M_s(F) \in \mathbb{R}^{H\times W}$ from two separate modules are combined to evaluate the attention map $M(F)$ as in Equation 13. Since the two attention maps have different shapes, the attention maps are expanded to the size of $C \times H \times W$ before combining them.

$$M(F) = \sigma(M_c(F) + M_s(F)) \quad (13)$$

Where σ is the sigmoid activation function. Thus, BAM can highlight or suppress features in both spatial and channel dimensions and enlighten the representational power. After refining the feature map, the BAM block is fed through a global average pooling layer to reduce the spatial dimensions, effectively summarizing the information from the refined feature map into a 1D vector.

This vector is then processed through a dense layer with 1024 units and ReLU activation, followed by batch normalization to stabilize and accelerate training. Finally, a dense layer makes the output layer, which is developed for a binary classification task, including the 'Fractured' and 'Non-Fractured' classes. The algorithm for the proposed model is illustrated above.

3.4. Hardware and Software Setup

The study utilized a comprehensive setup consisting of a NVIDIA GeForce GTX 1080Ti GPU, an Intel Core i7 processor, 32GB of RAM, and the Python-based Keras library integrated with the TensorFlow framework.

This intelligent detection framework is trained and tested on the Google Collaboratory platform, offering a free and cloud-based environment for model development with GPU acceleration. Unlike model parameters, hyperparameters are set before the training process, which is crucial as they directly influence the model's performance. Proper hyperparameter tuning significantly impacts the development of robust and effective models. Table 1 illustrates the hyperparameter specification used in this study.

Table 1. Hyperparameter specifications

Hyperparameters	Values
Activation function	Sigmoid
Batch size	16
Epochs	40
Loss function	Binary cross entropy
Learning rate	0.001

4. Results and Discussion

Understanding the performance and learning patterns of the suggested model requires comprehending the accuracy and loss plots. The accuracy plot graphically represents the model's capacity to consistently predict data labels during training iterations on both the training and validation datasets.

The accuracy and loss plot of the suggested model are shown in Figure 6. The model's performance is evaluated throughout training by tracking how well its predictions match the labels. A loss plot depicts the model's loss function trend across training iterations or epochs.

This declining trend shows that the model improves with time in reducing prediction errors. Better alignment of the model's predictions and the real labels in the training data is shown by lower loss values. The model's response to changes in the training data and optimization procedure is reflected in the loss values that fluctuate throughout epochs, much like accuracy.

A confusion matrix is an excellent tool for evaluating the accuracy of the proposed model. The matrix compares the model's predictions with the actual labels for several classes to provide a methodical overview of the model's performance. It presents the findings in a tabular format, where the predicted labels are shown in the columns, and the actual labels are shown in the rows. Each cell in the matrix represents the instances where the model's predictions match or deviate from the actual labels.

The confusion matrix is divided into four quadrants, with those of the diagonal signifying misclassification and items on the diagonal representing accurate predictions. The suggested model's confusion matrix with actual and predicted results is shown in Figure 7. A high number of FN is more crucial than FP, as a missed fracture causes serious complications.

In this case, 144 fractured labels were correctly predicted under 'fractured', while 143 non-fractured true labels were correctly predicted under 'non-fractured'.

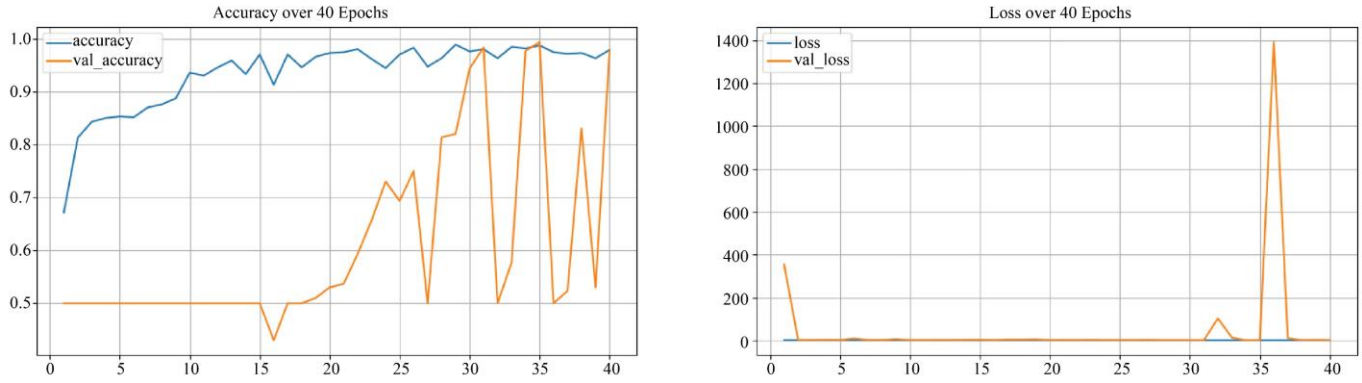

Fig. 6 Accuracy and loss plot of the proposed model

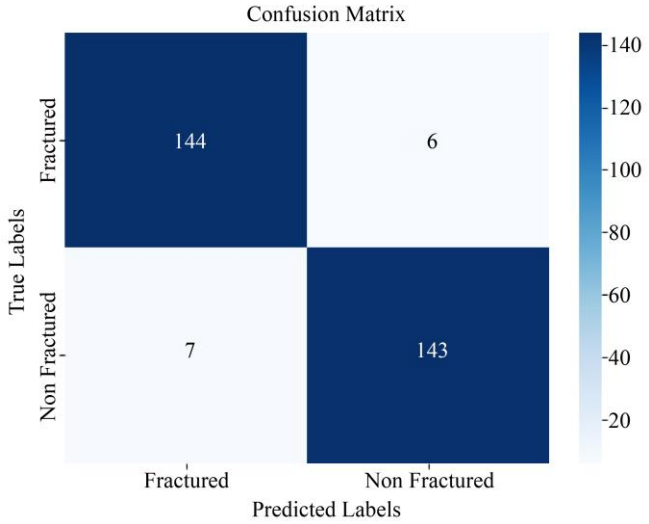

Fig. 7 Confusion matrix of the suggested model

The confusion matrix's performance metrics comprehensively assess the suggested model's effectiveness. The F1-score, accuracy, precision, and recall are the four main metrics used to fully assess the effectiveness and operational efficiency of the suggested model. For evaluating the model's performance, these metrics—which are founded on the ideas of False Positive (FP), False Negative (FN), True Negative (TN), and True Positive (TP)—are crucial. Equations (14), (15), (16), and (17) provide mathematical definitions of these performance parameters.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (14)$$

$$Precision = \frac{TP}{TP+FP} \quad (15)$$

$$Recall = \frac{TP}{TP+FN} \quad (16)$$

$$F1-score = 2 \times \frac{precision \times Recall}{Precision+Recall} \quad (17)$$

The acquired performance metrics demonstrate the developed model's remarkable effectiveness, as illustrated in Figure 8. With an accuracy of 97%, the proposed model correctly identified bone fractures, demonstrating the model's strong capability to make correct predictions. 96.12% precision denotes that the model predicts positive classes with a low false positive rate. This high precision reduces the likelihood of falsely diagnosing a fracture. A high recall value is crucial because failing to detect a fracture will cause serious health complications. The proposed model provides 96.70% recall, demonstrating the model's success in identifying the fracture case. The F1-score of 96.38% shows that the framework upholds a good balance between precision and recall, making precise positive predictions. These high values across the metrics suggest the robustness of the model, making it reliable for efficient fracture diagnosis.

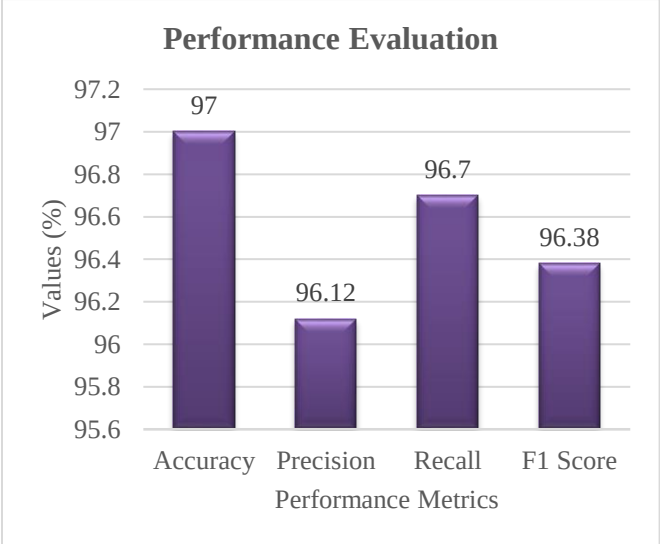

Fig. 8 Performance metrics

The Receiver Operating Characteristic (ROC) demonstrates the trade-off between the TP and FP rates across various threshold conditions. The area under the curve (AUC) signifies the model's overall performance, where an AUC of 1.0 specifies a perfect model with an FP rate of '0' and a TP rate of '1'. The ROC-AUC of the proposed model is 0.96, indicating better classification, as illustrated in Figure 9.

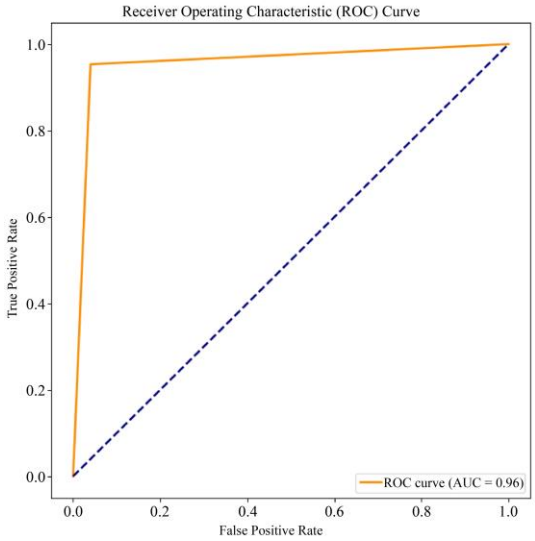

Fig. 9 ROC curve

Figure 10 illustrates the predicted output showing a fractured label using the proposed bone fracture detection framework.

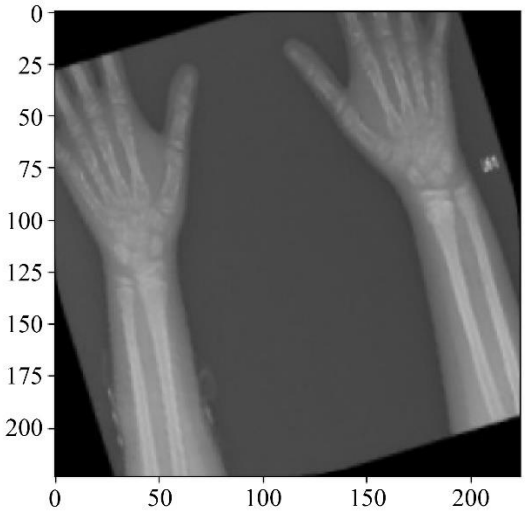

Fig. 10 Predicted 'Fractured' output

The proposed model for bone fracture detection is compared with existing methods for evaluating its effectiveness. Table 2 compares the model, and its graphical representation is given in Figure 11.

Table 2. Performance comparison of the suggested model with existing methods

Methodology	Dataset	No. of images	Accuracy (%)
YOLOv8 [7]	FracAtlas	4083	81.00
Ensemble Model [8]	MURA-v1.1	6542	92.00
CrackNet [10]	Radiopaedia + data collected from the hospital	1052	90.11
ParallelNet [11]	Thigh Fracture dataset	3842	87.80
CNN [13]	Bone Fractures Dataset	9103	91.00
ResNet50-DenseNet 121[14]	MURA dataset	1266	93.41
Proposed Model	**Bone Fracture Dataset**	**9463**	**97.00**

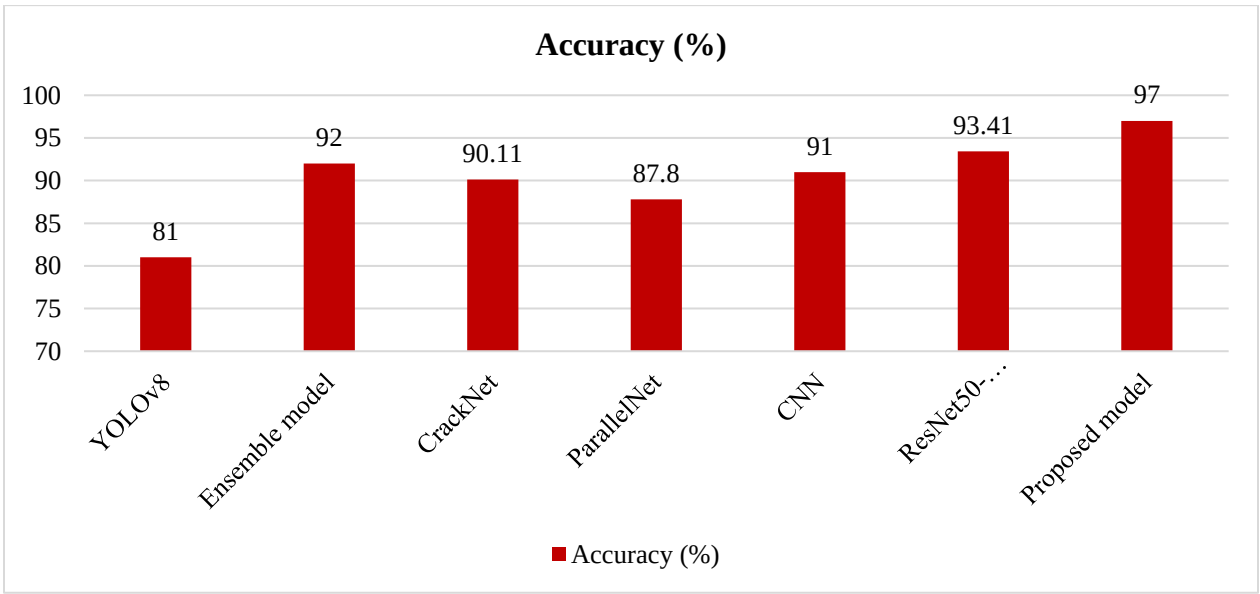

Fig. 11 Accuracy comparison

Table 2 compares the range of accuracies achieved across different methodologies for bone fracture detection. The efficiency of the proposed model, with an accuracy of 97%, using a bone fracture dataset with 9463, outperformed other models like YOLOv8, Ensemble Model, ParallelNet, ResNet50-DenseNet 121, and CNN. ResNet50-DenseNet 121, with an accuracy of 93.41%, used the MURA dataset with 1266 images, which closely indicates the strength of the hybrid model. The robust performance of an ensemble model with the MURA-v1.1 dataset having 92% accuracy highlights the effectiveness of multiple models. Other models like CNN, YOLOv8 and CrackNet show competitive performances. However, the proposed deep learning model with attention mechanism underscores its effectiveness in bone fracture detection, making it reliable for medical diagnosis.

5. Conclusion

The timely and accurate detection of bone fractures significantly impacts patient outcomes and prevents further complications. This study proposed an AI-enhanced framework incorporating a deep learning model enhanced by an attention mechanism. ResNet 50 architecture enhanced with BAM at both spatial and channel levels utilized a bone fracture dataset comprising X-ray images of fractured and non-fractured cases for bone fracture detection. Different preprocessing and data augmentation techniques played a major role in improving the dataset diversity. The model's ability to effectively detect bone fractures in real-time scenarios by minimizing FP and FN cases is demonstrated with an accuracy of 97%, precision of 96.12%, recall of 96.70%, and F1 score of 96.38%. The ROC-AUC of 0.96 specifies the superior classification performance of the DL model. The suggested model outperformed existing models such as YOLOv8, Ensemble Model, ResNet50-DenseNet 121, and CNN, making it suitable for medical diagnostics with reduced workload on healthcare professionals and better patient care.

Acknowledgments

The author profoundly appreciates the supervisor's guidance and unwavering support throughout this study.

References

[1] Christopher Jerome, Benjamin Hoch, and Cathy S. Carlson, *5 - Skeletal System*, Comparative Anatomy and Histology: A Mouse, Rat, and Human Atlas, 2nd ed, Academic Press, pp. 67-88, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[2] Vassiliki Tzelepi et al., *Bone Anatomy, Physiology and Function*, Bone Metastases: A Translational and Clinical Approach, 1st ed, Springer Netherlands, pp. 3-30. 2009. [CrossRef] [Google Scholar] [Publisher Link]

[3] N. L. Fazzalari, "Bone Fracture and Bone Fracture Repair," *Osteoporosis International*, vol. 22, pp. 2003-2006, 2011. [CrossRef] [Google Scholar] [Publisher Link]

[4] Nancy Johari, and Natthan Singh, "Bone Fracture Detection Using Edge Detection Technique", *Soft Computing: Theories and Applications: Proceedings of SoCTA*, Springer Singapore, vol. 2, pp. 11-19, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[5] Dhahir Basim Mohammed, Hameed Imad Hadi, and Jaber Ahmed Radee, "Prospective and Retrospective Study of Fractures According to Trauma Mechanism and Type of Bone Fracture," *Research Journal of Pharmacy and Technology*, vol. 10, no. 11, pp. 3810-3818, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[6] Muhammet Emin Sahin, "Image Processing and Machine Learning-Based Bone Fracture Detection and Classification using X-Ray Images," *International Journal of Imaging Systems and Technology*, vol. 33, no. 3, pp. 853-865, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[7] Amal Alshahrani, and Alaa Alsairafi, "Bone Fracture Classification Using Convolutional Neural Networks from X-Ray Images," *Engineering, Technology & Applied Science Research*, vol. 14, no. 5, pp. 16640-16645, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[8] A. Tahir et al., "Enhancing Diagnosis: Ensemble Deep Learning Model for Fracture Detection Using X-Ray Images," *Clinical Radiology*, vol. 79, no. 11, pp. e1394-e1402, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[9] M Abhijeeth Thaarakaraam, M Manideep Reddy, and Vaidehi Vijayakumar, "CNN Based Bone Fracture Detection for Medical Imaging Using Resnet-50, *International Journal of Technical Research & Science*, pp. 216-22, 2024. [Google Scholar] [Publisher Link]

[10] Yangling Ma, and Yixin Luo, "Bone Fracture Detection through the Two-Stage System of Crack-Sensitive Convolutional Neural Network," *Informatics in Medicine Unlocked*, vol. 22, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[11] Mengxuan Wang et al., "Parallelnet: Multiple Backbone Network for Detection Tasks on Thigh Bone Fracture," *Multimedia Systems*, vol. 27, pp. 1091-1100, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[12] Shahnaj Parvin, and Abdur Rahman, "A Real-Time Human Bone Fracture Detection and Classification from Multi-Modal Images Using Deep Learning Technique," *Applied Intelligence*, vol. 54, pp. 9269-9285, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[13] Aymen Saad, Usman Ullah Sheikh, and Mortada Sabri Moslim, "Developing Convolutional Neural Network for Recognition of Bone Fractures in X-ray Images," *Advances in Science and Technology Research Journal*, vol. 18, no, 4, pp. 228-237, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[14] Puja Dey et al., "Hybrid Deep Transfer Learning Framework for Humerus Fracture Detection and Classification from X-ray Images," *2024 4th International Conference on Intelligent Technologies (CONIT)*, Bangalore, India, pp. 1-6, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[15] Sai Charan Medaramatla et al., "Detection of Hand Bone Fractures in X-ray Images Using Hybrid Yolo Nas," *IEEE Access*, vol. 12, no. 57661-57673, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[16] Santoshachandra Rao Karanam, Y. Srinivas, and S. Chakravarty, "A Supervised Approach to Musculoskeletal Imaging Fracture Detection and Classification Using Deep Learning Algorithms," *Computer Assisted Methods in Engineering and Science*, vol. 30, no. 3, pp. 369-385, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[17] Mohamed A. Kassem et al., "Explainable Transfer Learning-Based Deep Learning Model for Pelvis Fracture Detection," *International Journal of Intelligent Systems*, vol. 2023, no. 1, pp. 1-10, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[18] Madona B. Sahaai et al., "ResNet-50 Based Deep Neural Network Using Transfer Learning for Brain Tumor Classification," *AIP Conference Proceedings*, vol. 2463, no. 1, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[19] Kosrat Dlshad Ahmed, and Roojwan Hawezi, "Detection of Bone Fracture Based on Machine Learning Techniques," *Measurement: Sensors*, vol. 27, pp. 1-6, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[20] Brett Koonce, *Convolutional Neural Networks with Swift for Tensorflow: Image Recognition and Dataset Categorization*, Apress, pp. 1-245, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[21] Fanyi Wang, Haotian Hu, and Cheng Shen, "BAM: A Balanced Attention Mechanism for Single Image Super Resolution," *Electrical Engineering and Systems Science: Image and Video Processing*, arXiv, pp. 1-8, 2021. [CrossRef] [Google Scholar] [Publisher Link]