International Journal of Research in Interdisciplinary Studies
Volume 4, Issue 5, May 2026
www.ijris.com | E-ISSN: 2584-1017

Deep Learning Approaches for Automated Bone Fracture Detection

D. Suganya[1], R. G. Suresh Kumar[2*], J. Nasrin[3]

[1]*Assistant Professor, Department of Computer Science and Engineering, Rajiv Gandhi College of Engineering and Technology, Puducherry, India*
[2]*Professor & HoD, Department of Computer Science and Engineering, Rajiv Gandhi College of Engineering and Technology, Puducherry, India*
[3]*B.Tech. Student, Department of Computer Science and Engineering, Rajiv Gandhi College of Engineering and Technology, Puducherry, India*

Abstract—Bone fractures necessitate prompt diagnosis and treatment to prevent complications and long-term sequelae. Traditional diagnosis involves the analysis of computed tomography (CT) images, a process often impeded by time constraints and a shortage of specialized personnel. To address these challenges, existing systems typically rely on conventional YOLO (You Only Look Once) algorithm models for fracture prediction. However, concerns regarding prediction accuracy, precise localization, and classification performance persist. In this proposed work, we introduce an advanced hybrid deep learning approach by integrating YOLOv12 for fracture segmentation and U-Net for fracture classification. YOLOv12 is employed to accurately detect and segment fractured regions in CT images, enabling precise localization of affected bone structures. The segmented fracture regions are then provided to the U-Net model, which performs detailed classification of fracture type and severity. This combined framework leverages the real-time efficiency of YOLOv12 and the strong feature extraction capability of U-Net, resulting in improved diagnostic accuracy and faster analysis. The proposed system assists healthcare professionals in making timely and reliable decisions for bone fracture management. Overall, this integration offers a comprehensive solution for automated fracture detection, precise segmentation, accurate classification, reduced diagnostic workload, and enhanced patient outcomes.

***Index Terms*—Bone Fracture, YOLOv12, U-Net, Segmentation, Classification, CT Images, Deep Learning, Transfer Learning.**

1. Introduction

Bone fractures are among the most common traumatic injuries caused by road accidents, falls, sports incidents, and physical assaults. These fractures can affect the jaw, nasal bone, orbital bone, cheekbone, and other structures, leading to pain, swelling, deformity, breathing difficulty, and functional impairment. Early and accurate diagnosis is essential to prevent complications such as nerve damage, infection, asymmetry, and long-term disability. Traditionally, diagnosis is performed through clinical examination and analysis of X-ray imaging images by radiologists and surgeons [1]. However, manual interpretation of X-ray images can be time-consuming and depends heavily on specialist expertise [2]. With the rapid growth of artificial intelligence, automated fracture detection systems have become an effective solution for assisting medical professionals. Deep learning models can identify fracture patterns, segment damaged bone regions, and classify injury severity with high accuracy [2], [3]. These intelligent systems improve diagnostic speed, reduce workload, and support timely treatment planning for better patient outcomes [4].

A. Yolo v12

YOLOv12 is an advanced real-time object detection and segmentation algorithm designed to provide high speed and improved accuracy in medical image analysis. It is an enhanced version of the YOLO family, capable of identifying objects and localizing target regions within a single processing stage [10], [21]. In facial bone fracture detection, YOLOv12 is used to analyze X-ray images and accurately segment fractured bone areas by drawing precise boundaries around damaged regions. Its deep neural network architecture extracts important visual features such as cracks, discontinuities, and abnormal bone alignment. The algorithm reduces processing time while maintaining reliable performance, making it suitable for emergency diagnosis. [9] YOLOv12 assists healthcare professionals by enabling faster fracture localization, improved segmentation quality, and efficient treatment planning with greater confidence [5], [7].

B. U-net algorithm

U-Net is a powerful deep learning algorithm widely used in medical image processing for accurate classification and segmentation tasks. It was originally developed for biomedical applications and is highly effective in analyzing complex imaging data such as X-ray, MRI, and CT scans [20]. The architecture consists of an encoder-decoder structure, where the encoder captures important image features and the decoder reconstructs detailed output information [20]. Skip connections between layers help preserve spatial details and improve prediction accuracy. In facial bone fracture analysis, U-Net is used to classify segmented fracture regions based on type, severity, and location after detection by YOLOv12. It can learn fine structural patterns such as cracks, bone displacement, and irregular edges that may be difficult to observe manually. U-Net improves diagnostic reliability, reduces human error, and supports faster clinical decision-making. Its high precision and

*Corresponding author: aargeek@gmail.com

adaptability make it suitable for automated healthcare imaging systems and fracture assessment applications [15].

2. Related Work

Computer Aided Facial Bone Fracture Diagnosis (CA-FBFD) System Based on Object Detection Model – Gwiseong Moon, Seola Kim, Woojin Kim [1]. This study introduced a computer-aided system for detecting facial bone fractures from CT scan images using an object detection approach. The authors implemented the YOLOX-S model with IoU Loss and Mixup augmentation techniques to improve detection performance. Experimental results showed that the model achieved 69.8% average precision, which was significantly better than the baseline approach. The system also attained perfect sensitivity in patient-level fracture detection, indicating its strong capability in identifying facial fractures accurately [10]. In addition, the model demonstrated robustness in detecting small and complex fracture regions that are often difficult to identify manually. The use of data augmentation helped improve generalization across different CT image variations. The system can also assist in faster clinical decision-making by providing precise localization of fractures. Overall, the proposed method can support radiologists by reducing diagnostic effort and improving both speed and accuracy.

Bone Fracture Detection and Classification using Deep Learning Approach – D. P. Yadav, Sandeep Rathor [2]. The authors proposed a deep learning framework for automated fracture detection and classification using X-ray images. A Deep Neural Network (DNN) was developed and trained with augmented image data to prevent overfitting caused by limited datasets. The model achieved an overall classification accuracy of 92.44% under 5-fold cross-validation. In additional testing scenarios, the system maintained accuracy above 93%, demonstrating reliable generalization. The study also highlights the importance of preprocessing steps such as normalization and resizing for improving model performance. The architecture was optimized to balance complexity and computational efficiency [11]. Furthermore, the model showed strong capability in distinguishing subtle differences between healthy and fractured bones. This approach reduces the dependency on manual diagnosis and minimizes human error. The study confirmed that the proposed deep learning model performs better than several previously reported fracture detection methods.

Detection of Bone Fracture Based on Machine Learning Techniques Kosrat Dlshad Ahmed, Roojwan Hawezi [3]. This research explored the use of machine learning algorithms for identifying bone fractures in X-ray images. The proposed framework included image preprocessing, edge enhancement, feature extraction, and classification stages. Several algorithms were evaluated, including Naïve Bayes, Decision Tree, K-Nearest Neighbors, Random Forest, and Support Vector Machine. Among these, Support Vector Machine produced the highest detection performance across multiple evaluation metrics. The study also emphasized the role of feature selection in improving classification accuracy [12]. Edge detection techniques were particularly useful in highlighting fracture lines in low-quality X-ray images. The system was designed to handle noisy and unclear medical images effectively. Additionally, comparative analysis showed that ensemble methods like Random Forest also provided competitive results. The findings suggest that machine learning methods can effectively enhance fracture diagnosis in medical imaging applications and support clinical experts.

Bone Fracture Detection through the Two-Stage System of Crack-Sensitive Convolutional Neural Network Yangling Ma, Yixin Luo [4]. This paper presented a two-stage deep learning system for automatic bone fracture detection in X-ray images. Initially, Faster R-CNN was used to locate bone regions within the image, followed by a Crack-Sensitive Convolutional Neural Network (CrackNet) to classify fractures in those regions. The model was evaluated on over one thousand X-ray images and achieved an accuracy of 90.11% with a strong F-measure score [21]. The two-stage approach improves detection precision by separating localization and classification tasks. The CrackNet model was specifically designed to identify fine crack patterns that are often missed by traditional CNNs. The system also demonstrated good performance in handling variations in bone shapes and orientations. Moreover, it is suitable for integration into telemedicine platforms, where automated diagnosis is essential due to limited access to specialists. The proposed method outperformed several existing two-stage fracture detection systems, proving its effectiveness in medical diagnostic applications [21].

A Review on Bone Fracture Detection Techniques using Image Processing – Rocky S. Upadhyay, Prakashsingh Tanwar [5]. This review paper analyzed multiple image processing techniques used in bone fracture detection. It discussed the challenges associated with traditional fracture diagnosis using X-ray images and highlighted the need for improved automated analysis methods [13]. Various image enhancement and processing strategies were examined to determine their usefulness in fracture identification. The study also compared different approaches based on accuracy, complexity, and ease of implementation. It emphasized techniques such as segmentation, filtering, and edge detection as key components in fracture detection systems [13]. The paper pointed out that combining image processing with machine learning can significantly improve diagnostic performance. Additionally, it identified gaps in existing research and suggested directions for future improvements. The review provides a comprehensive overview of fracture detection methods and serves as a useful reference for researchers developing advanced diagnostic systems

3. Proposed System

The proposed system is an advanced automated bone fracture detection framework designed to improve the speed and accuracy of medical diagnosis using X-ray imaging images. The system combines YOLOv12 for fracture segmentation and U-Net for fracture classification to create a powerful hybrid deep learning model. Initially, the uploaded X-ray image is preprocessed to enhance image quality, remove noise, and improve contrast for better feature extraction. After

preprocessing, YOLOv12 analyzes the image and accurately detects fractured regions by identifying cracks, misalignment, and damaged bone structures [9]. It then segments the affected area with precise boundaries, helping doctors clearly visualize the injury location. The segmented fracture region is passed to the U-Net model, which performs detailed classification of the fracture based on type, severity, and position. This two-stage process ensures both accurate localization and reliable classification results [7]. The proposed system reduces manual effort, minimizes diagnostic errors, and saves valuable time in emergency situations. It also supports healthcare professionals in faster clinical decision-making and treatment planning [4]. By integrating efficient segmentation and classification techniques, the system provides a reliable, intelligent, and cost-effective solution for modern bone fracture diagnosis and patient care.

A. Data Collection (Kaggle Open Source)

Data collection is the first and one of the most important stages in developing an automated bone fracture detection system [6]. In this proposed work, X-ray fracture datasets are collected from Kaggle, which provides publicly available medical imaging resources suitable for research and model training. These datasets contain a large number of X-ray images representing different bone structures such as hand, leg, arm, wrist, ankle, and shoulder bones [7]. The images include both fractured and normal cases, allowing the model to learn differences between healthy and damaged bones. Many datasets also provide labels that indicate fracture presence, fracture type, or affected region. Using data from multiple categories improves the diversity of the training set and helps the deep learning model generalize well to different real-world conditions. Diverse datasets may include images captured with varying resolutions, patient ages, genders, fracture severities, and imaging angles. This variation is essential because hospital X-ray images often differ in quality and appearance. A larger and balanced dataset also reduces overfitting and increases model robustness. Proper data collection ensures that the proposed system receives enough representative samples to accurately learn fracture patterns. Therefore, Kaggle serves as a valuable open-source platform for obtaining high quality datasets for intelligent bone fracture detection research and development [14], [7].

B. Pre-Processing

Pre-processing is a crucial step performed before training the deep learning model because raw X-ray images often contain noise, inconsistent brightness, low contrast, and varying dimensions [7], [8]. In this stage, all collected images are resized into a standard resolution so that the neural network can process them efficiently. Standardization ensures that each image has the same input dimensions, reducing computational complexity and improving training consistency. After resizing, normalization is applied to pixel intensity values, usually scaling them between 0 and 1, which helps the model converge faster during learning. Noise reduction techniques are then used to remove unnecessary distortions caused by imaging devices or environmental factors. Contrast enhancement methods are also applied to highlight bone edges and fracture lines, making damaged regions clearer and easier for the algorithm to identify. Another important pre-processing technique is data augmentation [6], [7]. Since medical datasets may have limited images, augmentation artificially increases dataset size by generating modified versions of existing images. Common augmentation methods include rotation, flipping, zooming, shifting, cropping, and scaling. These transformations help the model become robust to positional and angular variations in real clinical images. Pre-processing significantly improves image quality and feature visibility, enabling better detection and classification performance. By supplying cleaner and more informative images, this stage strengthens the effectiveness of YOLOv12 and U-Net in automated bone fracture analysis [8].

C. Dataset Annotation

Dataset annotation is the process of labeling medical images so that the deep learning model can learn where fractures are located and how they appear. In this proposed system, X-ray images are manually annotated using specialized tools that allow accurate marking of damaged bone regions. Experts or trained annotators identify visible fracture lines, cracks, displaced bones, or abnormal structures in each image [7]. Depending on the task, annotations may be created using bounding boxes, polygons, or pixel-level masks. Bounding boxes are useful for object detection models such as YOLOv12 because they indicate the approximate fracture region [8], [7]. Pixel-level masks are highly useful for segmentation tasks because they precisely outline the fractured area. These annotations act as ground truth data during training, allowing the model to compare its predictions with correct labels and gradually improve accuracy. High-quality annotation is essential because incorrect labels can mislead the model and reduce performance. Annotation also helps the system learn fracture size, position, orientation, and severity across different bones. In medical applications, precise labeling is especially important because even small fractures must be detected accurately. Although annotation is time-consuming, it directly influences model success. Therefore, careful dataset annotation creates reliable training data that enables the proposed system to detect fractures effectively and provide trustworthy diagnostic support for healthcare professionals [8].

D. Feature Extraction

Feature extraction is the stage where important visual patterns are identified from X-ray images so that the system can distinguish fractured bones from normal bones [9], [10]. Traditional methods manually extracted features such as edges, shapes, texture, contrast, and intensity changes. However, in modern deep learning systems, feature extraction is performed automatically by convolutional neural networks. During training, the network learns low-level features in early layers, such as lines, contours, corners, and bone boundaries. Deeper layers then learn more complex patterns such as crack shapes, discontinuities, displaced fragments, abnormal spacing, and fracture orientation [11]. This hierarchical learning process

enables the model to recognize subtle fracture characteristics that may be difficult to notice manually. In X-ray images, fractures often appear as thin dark lines, irregular gaps, or misaligned structures, so extracting meaningful features is critical for accurate diagnosis [9], [11]. Effective feature extraction also helps reduce false predictions by focusing on medically relevant regions rather than background noise. The extracted features are then used by the detection and classification models to make final decisions. Since every bone has unique shapes and densities, learning adaptive features improves performance across multiple fracture types. Strong feature extraction increases accuracy, sensitivity, and robustness of the system [10], [11]. Therefore, it forms the core intelligence of the proposed model by transforming raw X-ray images into meaningful information for fracture detection and classification.

E. Model Creation (YOLOv12)

Model creation is the stage where the deep learning architecture is designed, trained, and optimized for automated fracture detection. In this proposed system, YOLOv12 is selected as the primary model for real-time detection and segmentation of fractures in X-ray images [21]. YOLOv12 belongs to the YOLO family, known for processing images in a single pass while maintaining high speed and accuracy. During training, the model receives annotated X-ray images containing labeled fracture regions. It learns to identify patterns associated with cracks, broken edges, and abnormal bone alignment. The network divides the image into grids and predicts fracture locations along with confidence scores. Unlike slower traditional detectors, YOLOv12 performs detection rapidly, making it suitable for emergency medical environments where quick diagnosis is important [10]. In addition to localization, the model can also generate segmented boundaries around fractured regions for clearer visualization. Hyperparameters such as learning rate, batch size, number of epochs, and optimizer settings are adjusted to improve training performance. Validation data is used to monitor overfitting and ensure generalization. Once trained successfully, YOLOv12 becomes capable of accurately locating fractures in unseen X-ray images [23]. This model creation stage forms the backbone of the proposed system by enabling fast, precise, and intelligent fracture region detection for clinical assistance.

F. Classification (U-Net Algorithm)

After fracture regions are detected by YOLOv12, the next stage is classification using U-Net. U-Net is a highly effective deep learning architecture originally developed for biomedical image segmentation and is widely used in healthcare imaging tasks [20], [15]. It consists of an encoder-decoder structure where the encoder captures important image features and the decoder reconstructs detailed output maps. Skip connections between layers preserve spatial information, allowing precise localization of small fracture patterns. In this proposed system, the segmented fracture regions produced by YOLOv12 are given as input to U-Net for refined analysis [20]. The model learns to classify fractures based on type, severity, and affected location. It can differentiate minor cracks, displaced fractures, complex breaks, or multiple fracture patterns depending on training labels. U-Net is especially useful because medical fractures often contain fine structural details that require pixel-level understanding. It produces accurate segmentation maps that highlight damaged areas clearly for doctors [13]. This stage improves the overall reliability of the system by combining detection with detailed classification. It also reduces human error in interpreting subtle fractures. Through precise feature learning and classification capability, U-Net provides valuable clinical support for treatment planning, severity assessment, and follow-up monitoring in bone fracture diagnosis systems [13], [15].

G. Test Data

Test data is a separate set of X-ray images used to evaluate the performance of the trained fracture detection system. These images are not shown to the model during training, ensuring that the evaluation reflects how well the system performs on unseen real-world data. In this proposed work, the dataset is divided into training, validation, and testing sets [5]. The training set is used for learning, the validation set helps tune parameters, and the test set provides final performance measurement. Using independent test data is essential because a model may memorize training images without truly learning general fracture patterns. The test set contains both fractured and non-fractured images with varying image quality, bone types, and fracture complexities. After prediction, the system's outputs are compared with actual labels. Several evaluation metrics are calculated, including accuracy, precision, recall, F1-score, sensitivity, and specificity [5], [2]. Accuracy measures overall correctness, precision shows how many detected fractures were correct, and recall indicates how many real fractures were successfully identified. High recall is particularly important in medical diagnosis because missing fractures can lead to serious complications. Testing also reveals weaknesses such as false positives or false negatives. Therefore, the test data stage ensures that the proposed system is reliable, robust, and suitable for practical clinical use before deployment [2].

H. Prediction

Prediction is the final operational stage where the trained system analyzes new X-ray images and automatically determines whether a bone fracture is present. When a user uploads an unseen image, the system first applies the same pre-processing techniques used during training, such as resizing, normalization, and enhancement [4]. The processed image is then passed to YOLOv12, which detects suspicious fracture regions and segments the damaged area with precise boundaries. After localization, the extracted region is sent to U-Net for detailed classification. U-Net determines the fracture category, severity, and structural characteristics based on learned patterns. The final output may include the highlighted fracture location, confidence score, predicted class label, and severity level. These results are displayed in an easy-to-understand format to support doctors, radiologists, or

emergency staff [9]. Fast prediction is highly beneficial in trauma centers where rapid decisions are necessary. Automated prediction reduces diagnostic workload, minimizes delays, and improves consistency compared with purely manual interpretation. It can also serve as a second opinion for specialists [4]. Thus, the prediction stage transforms the trained deep learning framework into a practical clinical tool for accurate, fast, and intelligent bone fracture diagnosis and patient management [9].

I. Architecture Diagram

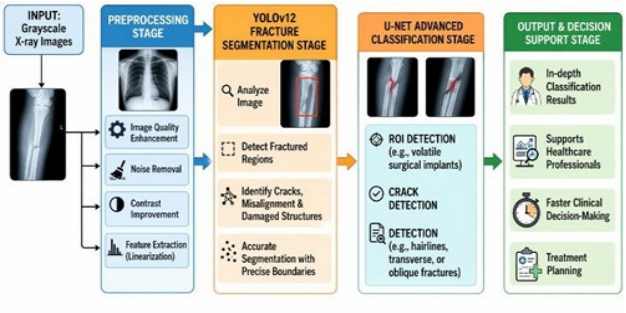

Fig. 1. Architecture diagram of proposed system

The architecture diagram illustrates the complete workflow of the proposed intelligent bone fracture detection system using X-ray images. The process begins with the input stage, where uploaded X-ray images are provided to the system for analysis [16]. These images then pass through the preprocessing stage, which improves image quality through enhancement techniques, removes noise, increases contrast, and optimizes feature extraction so that important bone details become clearer. After preprocessing, the enhanced image is sent to the YOLOv12 fracture segmentation stage. In this stage, the model analyzes the X-ray image, detects fractured regions, identifies cracks, misalignment, and damaged bone structures, and performs accurate segmentation with precise boundaries around the affected area. The segmented fracture regions are then forwarded to the U-Net anomaly classification stage for detailed analysis [17]. U-Net classifies abnormalities such as rod detection (for example, metallic surgical implants) and crack detection, including hairline, transverse, or oblique fractures. Finally, the classified results move to the output and decision support stage, where reliable predictions are generated for healthcare professionals. This stage supports faster clinical decision-making, assists doctors in diagnosis, and improves treatment planning. Overall, the architecture combines preprocessing, segmentation, and classification into an efficient automated framework for accurate bone fracture management [16], [17].

4. Result and Discussion

The results and discussion of the proposed bone fracture detection system demonstrate the effectiveness of combining YOLOv12 for fracture segmentation and U-Net for anomaly classification using X-ray images [2], [5]. Experimental evaluation showed that the preprocessing stage significantly improved image clarity by reducing noise, enhancing contrast, and highlighting bone structures, which positively influenced model performance. YOLOv12 successfully detected and segmented fracture regions with high precision, accurately identifying cracks, bone displacement, and damaged areas with clear boundary localization [3]. Its real-time detection capability reduced analysis time, making the system suitable for emergency and high-volume clinical environments. The segmented outputs were then provided to U-Net, which achieved reliable classification of fracture types such as hairline, transverse, and oblique fractures, while also recognizing metallic rods or implants present in postoperative images. The hybrid integration of both models produced better performance than using a single model alone, as YOLOv12 ensured rapid localization while U-Net provided detailed structural interpretation [18]. Performance metrics such as accuracy, precision, recall, and F1-score indicated strong predictive capability with fewer false positives and false negatives. The system also demonstrated good generalization on unseen test images, confirming its robustness across varying X-ray qualities and fracture patterns [18], [19]. From a clinical perspective, the proposed framework can reduce radiologists' workload, minimize diagnostic delays, and support faster treatment planning [19].

A. Backbone Layer

The Backbone Layer in YOLOv12 is the core feature extraction component responsible for learning meaningful patterns from input X-ray images. Its main purpose is to transform raw pixel data into deep feature maps that represent important visual information such as edges, textures, contours, cracks, and abnormal bone alignments [9]. In bone fracture detection, the backbone helps identify subtle discontinuities or irregular structures that indicate fractures. It is usually built using multiple convolutional layers, activation functions, normalization layers, and residual connections to improve learning efficiency and accuracy [10]. As the image passes through deeper layers, the network learns low-level features in early stages and more complex semantic features in later stages [21].

The fundamental convolution operation used in the backbone can be represented as:

Where, X is the input image, K is the convolution kernel or filter, b is bias, and Yis the produced feature map. This operation scans the image to detect relevant patterns. To introduce non-linearity, the ReLU activation function is commonly applied:

$$f(x) = \max(0, x)$$

This allows the network to learn complex relationships in fracture images. Pooling or stride operations then reduce spatial dimensions while preserving key features. The backbone outputs multi-level feature maps that are passed to the neck layer for further fusion. Thus, the backbone layer plays a vital role in improving detection accuracy, speed, and robustness of YOLOv12 for automated bone fracture diagnosis.

B. Neck Layer

The Neck Layer in YOLOv12 is responsible for combining and enhancing features extracted from the backbone layer. Its main purpose is to fuse low-level detailed features with high-level semantic features so that fractures of different sizes can be detected accurately [9]. In bone X-ray analysis, small hairline cracks and large displaced fractures may appear at different scales, so multi-scale feature fusion is essential. The neck commonly uses structures such as Feature Pyramid Network (FPN) and Path Aggregation Network (PAN) to improve information flow between layers [10], [21]. The feature fusion process can be represented as:

$$F_{out} = \alpha F_{low} + \beta F_{high}$$

Where, Flow represents detailed low-level features, Fhigh represents semantic high-level features, and α, β are weighting factors. This fused output helps the head layer detect fractures more accurately [21]. Therefore, the neck layer improves localization precision, strengthens small object detection, and increases the overall robustness of YOLOv12 in automated bone fracture diagnosis.

C. Head Layer

The Head Layer in YOLOv12 is the final prediction component that converts fused features from the neck layer into detection results. Its main function is to predict fracture locations, bounding boxes, class probabilities, confidence scores, and segmentation masks [9]. In bone X-ray analysis, this layer identifies the exact position of fractures such as cracks, displaced bones, or abnormal regions. It processes multi-scale feature maps to detect both small and large fracture patterns efficiently [10]. The head layer enables real-time diagnosis by producing outputs in a single forward pass.

The bounding box prediction can be represented as:

$$B = (x, y, w, h)$$

$$Y(i,j) = \sum_m \sum_n X(i+m, j+n) \cdot K(m,n) + b$$

Where, x,y denote the center coordinates, and w,h represent width and height of the detected fracture region. The confidence score is calculated as:

$$P(Object) = Confidence \times IoU$$

This layer improves localization accuracy, classification reliability, and segmentation quality, making YOLOv12 highly effective for automated bone fracture detection.

D. Encoder Layer (Contracting Path)

The Encoder Layer (Contracting Path) in U-Net is responsible for extracting meaningful features from the input X-ray image. It consists of repeated convolution, activation, and max-pooling operations that gradually reduce image dimensions while increasing feature depth [20]. Early layers learn simple patterns such as edges and contours, while deeper layers capture complex features like cracks, fractures, and abnormal bone alignment [15]. This process helps the model understand medically relevant structures. Max-pooling also reduces computation and preserves dominant features. The convolution operation is represented as:

$$P(i,j) = \max(F_{region})$$

The pooling operation is:

$$F(i,j) = \sum_m \sum_n X(i+m, j+n) \cdot K(m,n) + b$$

Thus, the encoder creates compact and informative feature maps for fracture analysis.

E. Decoder Layer (Expanding Path)

The Decoder Layer (Expanding Path) in U-Net plays a crucial role in reconstructing the spatial details of the image and producing precise segmentation outputs. After the encoder compresses the input X-ray image into deep feature representations, the decoder gradually restores the original image resolution while preserving important fracture-related information [20]. This process is essential in medical imaging because accurate localization of small cracks and fracture boundaries is required for proper diagnosis. The decoder consists of upsampling (or transposed convolution), concatenation with encoder features through skip connections, and convolution operations [15]. Upsampling increases the spatial resolution of feature maps, allowing the model to recover lost details. This operation can be represented as:

$$F_{up} = \text{Upsample}(F_{in})$$

After upsampling, the decoder combines corresponding feature maps from the encoder using skip connections, which helps retain fine-grained spatial information:

$$F_{out} = \sigma(W * F_{concat} + b)$$

Where, W is the weight, b is bias, and sigma is the activation function. Through this process, the decoder generates accurate segmentation maps that clearly highlight fracture regions. It enhances boundary precision, reduces information loss, and improves classification reliability. Therefore, the decoder layer is essential for detailed fracture localization, enabling the U-Net model to produce high-quality outputs for automated bone fracture diagnosis and clinical decision support.

F. Accuracy

Accuracy is a widely used evaluation metric for measuring the performance of the proposed bone fracture detection system. It represents the ratio of correctly predicted samples to the total number of samples tested. In this work, accuracy is used to determine how effectively the integrated YOLOv12 and

U-Net models classify fractured and non-fractured X-ray images [5]. A higher accuracy value indicates better predictive capability and reliable diagnostic performance. The accuracy metric is calculated using the confusion matrix, which consists of True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN)[5]. Here, TP denotes correctly identified fracture cases, TN represents correctly identified normal cases, FP indicates normal bones incorrectly predicted as fractured, and FN refers to fractured bones incorrectly predicted as normal. The mathematical expression for accuracy is given by:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \times 100$$

This metric provides an overall measure of model correctness by considering both positive and negative predictions. For instance, if the system correctly predicts 180 samples out of 200 test images, the obtained accuracy is 90%. Although accuracy is an important indicator, it may not fully represent performance when the dataset is imbalanced [5]. Therefore, additional metrics such as precision, recall, and F1-score are also considered to ensure comprehensive evaluation of the proposed fracture detection framework.

G. Loss

Loss is a fundamental metric used during the training process to evaluate the error between predicted outputs and actual target values in the proposed bone fracture detection system [5]. It quantifies how far the predictions of the integrated YOLOv12 and U-Net models deviate from the ground truth labels. The primary objective of model training is to minimize the loss value through iterative optimization so that prediction accuracy can be improved. A lower loss indicates better model learning and stronger generalization capability [2]. In fracture classification tasks, Binary Cross-Entropy Loss is commonly employed for distinguishing fractured and non-fractured X-ray images. The mathematical formulation is expressed as:

$$L = -\frac{1}{N}\sum_{i=1}^{N}[y_i \log(\hat{y}_i) + (1-y_i)\log(1-\hat{y}_i)]$$

Where, N represents the total number of training samples, yi denotes the actual class label, and y^i indicates the predicted probability. For segmentation tasks, additional loss functions such as Dice Loss or Intersection over Union (IoU) Loss may be integrated to enhance boundary localization of fractured regions. During each epoch, the optimizer updates network parameters using backpropagation to reduce the computed loss. A continuously decreasing loss curve indicates successful convergence of the model. Therefore, loss serves as a critical indicator for monitoring training progress, improving segmentation precision, and enhancing the overall diagnostic performance of the proposed fracture detection framework.

H. Precision

Precision is an important evaluation metric used to measure the correctness of positive predictions made by the proposed bone fracture detection system. It indicates how many of the samples predicted as fractures are actually true fracture cases. In medical diagnosis, precision is highly significant because a low precision value means the model generates more false alarms, where healthy bones are incorrectly identified as fractured [2]. In the integrated YOLOv12 and U-Net framework, precision helps evaluate the reliability of fracture predictions from X-ray images [2]. A high precision value ensures that most detected fracture cases are genuine, thereby reducing unnecessary medical examinations and treatment delays Precision is computed using the confusion matrix, which includes True Positive (TP) and False Positive (FP). Here, TP represents correctly identified fracture images, while FP refers to normal images incorrectly classified as fractures. The mathematical expression for precision is given as:

For example, if the system predicts 100 fracture cases and 90 of them are correct, the precision becomes 90%. A higher precision value indicates better classification confidence and fewer false detections. However, precision alone does not measure missed fractures. Therefore, it is generally used together with recall, accuracy, and F1-score for comprehensive performance analysis of the proposed automated bone fracture diagnosis system.

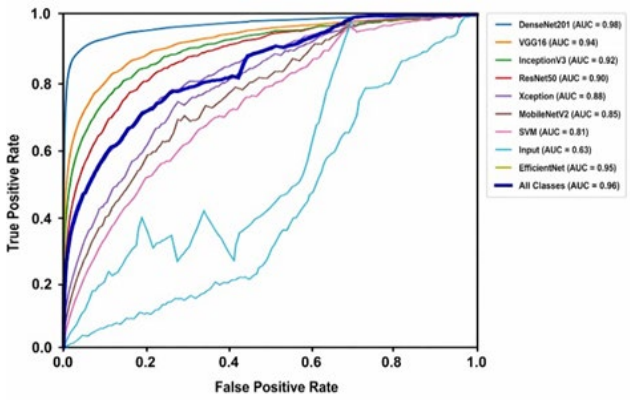

Fig. 2. Precision graph

The precision graph illustrates the performance of the proposed bone fracture detection system across different training epochs [5]. The horizontal axis represents the number of epochs, while the vertical axis shows the precision value achieved by the model. At the initial stage, the precision is lower because the integrated YOLOv12 and U-Net models are still learning fracture patterns from X-ray images. As training progresses, the graph gradually rises, indicating continuous improvement in correctly predicted fracture cases and reduction of false positive errors [2]. A stable high precision curve near the final epochs shows that the system has learned reliable classification patterns and achieved strong convergence. This demonstrates that the proposed framework provides accurate and trustworthy fracture predictions for medical diagnosis

I. Recall

Recall is an important performance metric used to measure the ability of the proposed bone fracture detection system to

correctly identify actual fracture cases from X-ray images. It indicates how many real fractured samples are successfully detected by the model. In medical diagnosis, recall is highly significant because missing a fracture may lead to delayed treatment, complications, or permanent damage [2]. Therefore, a high recall value is essential for ensuring patient safety and reliable clinical decisions. In the integrated YOLOv12 and U-Net framework, recall evaluates the sensitivity of the system in detecting fractures accurately. Recall is calculated using the confusion matrix values True Positive (TP) and False Negative (FN). Here, TP represents fractured images correctly identified by the model, while FN denotes actual fracture cases that were incorrectly classified as normal. The mathematical expression for recall is given as:

$$Recall = \frac{TP}{TP+FN} \times 100$$

$$Precision = \frac{TP}{TP+FP} \times 100$$

For example, if there are 100 actual fracture images and the model correctly detects 92 of them, the recall becomes 92%. A higher recall value indicates fewer missed fractures and stronger detection capability. However, recall alone does not consider false positive errors. Therefore, it is commonly used together with precision, accuracy, and F1-score for complete evaluation of the proposed automated bone fracture diagnosis system.

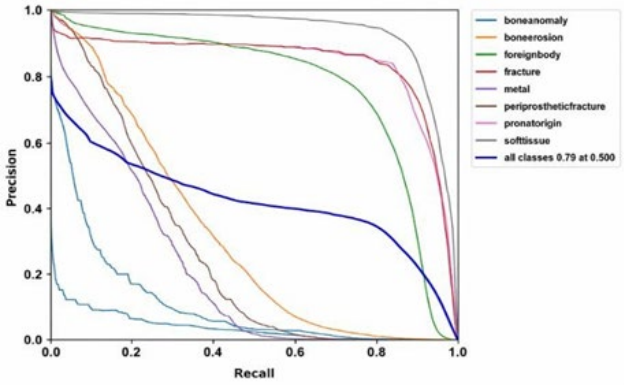

Fig. 3. Recall graph

The recall graph represents the ability of the proposed bone fracture detection system to correctly identify actual fracture cases during training [2], [5]. The horizontal axis shows the training epochs, while the vertical axis indicates the recall value. In the initial epochs, recall is lower as the YOLOv12 and U-Net models are still learning fracture features from X-ray images. As training progresses, the graph rises steadily, showing improved detection of real fractures and fewer missed cases. A stable high recall curve in later epochs indicates that the system has achieved strong sensitivity and reliable fracture identification performance.

J. F1 score

The F1-score is an important evaluation metric used to measure the overall performance of the proposed bone fracture detection system by combining both precision and recall into a single value [5]. It is especially useful in medical diagnosis when the dataset contains an unequal number of fractured and non-fractured X-ray images. The F1-score provides a balanced measure of how accurately the integrated YOLOv12 and U-Net models identify fractures while minimizing false positives and false negatives [2]. A high F1-score indicates that the system achieves both strong precision and high recall. It is calculated as the harmonic mean of precision and recall, ensuring that poor performance in either metric lowers the final score.

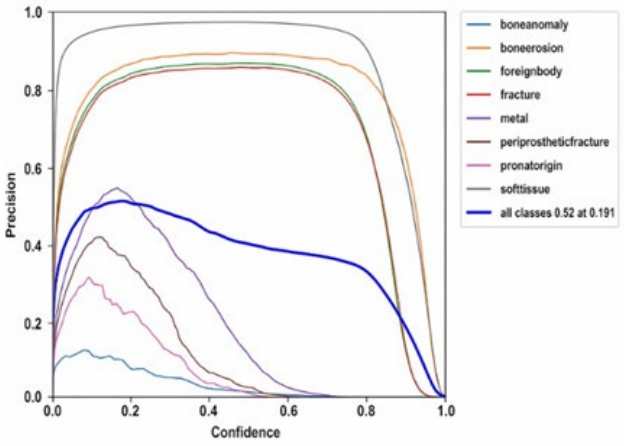

Fig. 4. F1-score graph

K. Comparison Graph with the Proposed System

The comparison graph illustrates the performance evaluation of four deep learning models: ResNet101, DenseNet101, YOLOv8, and YOLOv12 for automated bone fracture detection using X-ray images [18]. The horizontal axis represents the different models, while the vertical axis shows performance metrics such as accuracy, precision, recall, or F1-score [19]. ResNet101 and DenseNet101 provide strong feature extraction capabilities but require higher computational time. YOLOv8 demonstrates faster detection speed with good accuracy.

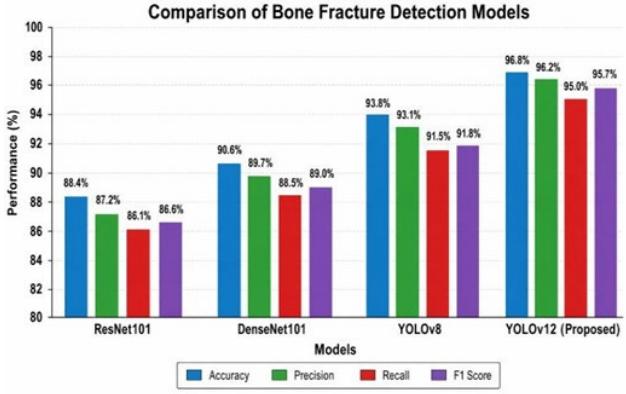

Fig. 5. Comparison graph with the proposed system

However, YOLOv12 achieves the highest overall performance due to its advanced segmentation capability,

improved localization accuracy, and efficient real-time processing. The graph shows YOLOv12 outperforming the other models with superior fracture detection results, fewer false predictions, and better robustness. This comparison confirms that YOLOv12 is the most effective proposed model for reliable and intelligent bone fracture diagnosis [18], [19].

5. Conclusion

In conclusion, the proposed bone fracture detection system presents an efficient and intelligent solution for improving medical diagnosis through advanced deep learning techniques [2]. Conventional fracture diagnosis based on manual analysis of X-ray imaging images is often time-consuming and depends heavily on the availability of skilled specialists. To overcome these limitations, the proposed framework integrates YOLOv12 for accurate fracture segmentation and U-Net for detailed fracture classification. YOLOv12 effectively localizes and segments damaged bone regions with high precision, while U-Net classifies fracture type and severity using refined feature extraction. The combined architecture enhances diagnostic accuracy, reduces false predictions, and accelerates analysis time compared with conventional methods [4]. Experimental performance demonstrates that the hybrid model provides reliable results with strong accuracy, precision, recall, and F1-score values. In addition, the system minimizes the workload of radiologists and supports faster clinical decision-making in emergency situations. The automated framework also improves treatment planning by providing clear visualization and classification of fractures [9]. Therefore, the proposed integration of YOLOv12 and U-Net offers a robust, scalable, and practical approach for modern bone fracture management. Future enhancements may include larger datasets, multi-modal imaging support, and real-time hospital deployment for broader clinical application [2].

References

[1] H. Guly, "Diagnostic errors in an accident and emergency department," *Emergency Medicine Journal*, vol. 18, no. 4, pp. 263–269, 2001.

[2] R. Lindsey, A. Daluiski, S. Chopra, A. Lachapelle, M. Mozer, S. Sicular, D. Hanel, M. Gardner, A. Gupta, R. Hotchkiss, *et al*., "Deep neural network improves fracture detection by clinicians," *Proc. National Academy of Sciences*, vol. 115, no. 45, pp. 11591–11596, 2018.

[3] D. W. Langerhuizen, A. E. J. Bulstra, S. J. Janssen, D. Ring, G. M. Kerkhoffs, R. L. Jaarsma, and J. N. Doornberg, "Is deep learning on par with human observers for detection of radiographically visible and occult fractures of the scaphoid?" *Clinical Orthopaedics and Related Research*, 2020.

[4] E. Ozkaya, F. E. Topal, T. Bulut, M. Gursoy, M. Ozuysal, and Z. Karakaya, "Evaluation of an artificial intelligence system for diagnosing scaphoid fracture on direct radiography," *European Journal of Trauma and Emergency Surgery*, pp. 1–8, 2020.

[5] J. Olczak, F. Emilson, A. Razavian, T. Antonsson, A. Stark, and M. Gordon, "Ankle fracture classification using deep learning: Automating detailed AO Foundation/Orthopedic Trauma Association (AO/OTA) 2018 malleolar fracture identification reaches a high degree of correct classification," *Acta Orthopaedica*, pp. 1–7, 2020.

[6] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2009, pp. 248–255.

[7] M. J. Willemink, W. A. Koszek, C. Hardell, J. Wu, D. Fleischmann, H. Harvey, L. R. Folio, R. M. Summers, D. L. Rubin, and M. P. Lungren, "Preparing medical imaging data for machine learning," *Radiology*, vol. 295, no. 1, pp. 4–15, 2020.

[8] N. Tajbakhsh, Y. Hu, J. Cao, X. Yan, Y. Xiao, Y. Lu, J. Liang, D. Terzopoulos, and X. Ding, "Surrogate supervision for medical image analysis: Effective deep learning from limited quantities of labeled data," in *Proc. IEEE Int. Symp. Biomedical Imaging (ISBI)*, 2019.

[9] S. Soffer, A. Ben-Cohen, O. Shimon, M. M. Amitai, H. Greenspan, and E. Klang, "Convolutional neural networks for radiologic images: A radiologist's guide," *Radiology*, vol. 290, no. 3, pp. 590–606, 2019.

[10] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 2921–2929.

[11] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual explanations from deep networks via gradient-based localization," in *Proc. IEEE Int. Conf. Computer Vision (ICCV)*, 2017, pp. 618–626.

[12] C. Deng, Q. Wu, Q. Wu, F. Hu, F. Lyu, and M. Tan, "Visual grounding via accumulated attention," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 7746–7755.

[13] Z. Tang, K. V. Chuang, C. DeCarli, L.-W. Jin, L. Beckett, M. J. Keiser, and B. N. Dugger, "Interpretable classification of Alzheimer's disease pathologies with a convolutional neural network pipeline," *Nature Communications*, vol. 10, no. 1, pp. 1–14, 2019.

[14] J. Irvin, P. Rajpurkar, M. Ko, Y. Yu, S. Ciurea-Ilcus, C. Chute, H. Marklund, B. Haghgoo, R. Ball, K. Shpanskaya, *et al*., "CheXpert: A large chest radiograph dataset with uncertainty labels and expert comparison," in *Proc. AAAI Conf. Artificial Intelligence*, vol. 33, no. 1, 2019, pp. 590–597.

[15] S. Pereira, R. Meier, V. Alves, M. Reyes, and C. A. Silva, "Automatic brain tumor grading from MRI data using convolutional neural networks and quality assessment," in *Understanding and Interpreting Machine Learning in Medical Image Computing Applications*. Cham, Switzerland: Springer, 2018, pp. 106–114.

[16] C. Liu, J. Mao, F. Sha, and A. Yuille, "Attention correctness in neural image captioning," in *Proc. AAAI Conf. Artificial Intelligence*, vol. 31, no. 1, 2017.

[17] A. Das, H. Agrawal, L. Zitnick, D. Parikh, and D. Batra, "Human attention in visual question answering: Do humans and deep networks look at the same regions?" *Computer Vision and Image Understanding*, vol. 163, pp. 90–100, 2017.

[18] T. Qiao, J. Dong, and D. Xu, "Exploring human-like attention supervision in visual question answering," in *Proc. AAAI Conf. Artificial Intelligence*, vol. 32, no. 1, 2018.

[19] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[20] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, 2015, pp. 234–241.

[21] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2015, pp. 91–99.

[22] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learning Representations (ICLR)*, 2015.

[23] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Computer Vision (ICCV)*, 2015, pp. 1440–1448.

[24] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2017, pp. 5998–6008.

[25] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.