scientific reports

OPEN

Exploring the impact of hyperparameter and data augmentation in YOLO V10 for accurate bone fracture detection from X-ray images

Parvathaneni Naga Srinivasu[1,2]✉, Gorli L. Aruna Kumari[3], Sujatha Canavoy Narahari[4], Shakeel Ahmed[5] & Abdulaziz Alhumam[6]

Accurately identifying bone fractures from the X-ray image is essential to prompt timely and appropriate medical treatment. This research explores the impact of hyperparameters and data augmentation techniques on the performance of the You Only Look Once (YOLO) V10 architecture for bone fracture detection. While YOLO architectures have been widely employed in object detection tasks, recognizing bone fractures, which can appear as subtle and complicated patterns in X-ray images, requires rigorous model tuning. Image augmentation was done using the image unsharp masking approach and contrast-limited adaptive histogram equalization before training the model. The augmented images assist in feature identification and contribute to overall performance of the model. The current study has performed extensive experiments to analyze the influence of hyperparameters like the number of epochs and the learning rate, along with the analysis of the data augmentation on the input data. The experimental outcome has proven that particular hyperparameter combinations, when paired with targeted augmentation strategies, improve the accuracy and precision of fracture detection. It is observed that the proposed model yielded an accuracy of 0.964 on evaluation over the augmented data. The statistical analysis of the classification precision across the augmented and raw images is observed as 0.98 and 0.95, respectively. In comparison with other deep learning models, the empirical evaluation of the YOLO V10 model clearly demonstrates its superior performance over conventional approaches for bone fracture detection.

Keywords Hyperparameters, Data augmentation, Bone fracture, Deep learning, YOLO model

The human body consists of 206 bones that vary in size, shape, and complexity. Bone fractures frequently occur due to physical impact and represent a break in the bone's continuity. These fractures can take various forms, such as transverse, oblique, spiral, and comminuted fractures, and are generally categorized as either open or closed[1]. The closed fractures are especially hard to recognize as the bone would be damaged from the inside, and the skin would be intact. Early and precise identification of fractures can significantly influence treatment decisions, ultimately improving patient outcomes[2]. In some cases, bone fractures are also caused by diseases like osteoporosis, where the bones are more prone to damage. Hence, timely identification of the abnormality and appropriate treatment is needed for better recovery of the patient[3].

Bone fracture detection relies on various medical imaging modalities. X-rays are the most commonly used imaging modality for detecting bone fractures because they are available[4]. , ability to rapidly view bone structures. X-ray imaging works by sending radiation through the body, where denser structures, such as bones,

[1]Amrita School of Computing, Amrita Vishwa Vidyapeetham, Amaravati 522503, Andhra Pradesh, India. [2]Department of Teleinformatics Engineering, Federal University of Ceará, Fortaleza 60455-970, Brazil. [3]Department of Computer Science and Engineering, GST, GITAM University, Visakhapatnam, AP, India. [4]Department of Electronics and Communication Engineering, Sreenidhi Institute of Science and Technology, Hyderabad 501301, India. [5]School of Computer Science (SCS), Taylor's University, Subang Jaya 47500, Malaysia. [6]Department of Computer Science, College of Computer Sciences and Information Technology, King Faisal University, Al-Ahsa 31982, Saudi Arabia. ✉email: parvathanenins@gmail.com

absorb more radiation and show white on the image. The X-ray images are sometimes challenging to process due to the variations in the patient anatomy, positioning, luminance, and divergent calibration of the devices. These are some of the common challenges that are faced in fracture detection from X-ray images[5]. Data augmentation techniques are commonly applied before the data is fed to train the machine-intelligent model to mitigate these issues. This helps the model generalize better over the unseen data. The deep learning (DL)[6] models are extensively used in the classification and localization of bone fractures from X-ray images[7,8]. The current study uses YOLO version 10[9] to localize bone fractures from the images. The current study tries to address some of the research questions associated with DL models in bone fracture detection.

RQ 1 To analyze the role of an object detection model like YOLO for the recognition of the type of bone fracture from the X-ray image.

RQ 2 Do the data augmentation approaches like image contrast enhancement and sharpening have any impact on the precision of the object recognition model.

RQ 3 Does the hyperparameter tuning, like the number of epochs and learning rate, impact the object recognition performance.

Machine intelligent models are widely used in the classification of X-ray images to identify abnormalities. Their systematic feature extraction capabilities enable the learning of complex visual patterns, including bone structure, fracture lines, and tissue abnormalities. However, there is still a necessity for models that are faster and precise enough to accurately recognize the subtle fractures from ordinary X-ray images[10]. Among these, the YOLO models are considered to be the most efficient model for abnormality detection with reasonable computational overhead. The current study performs various operations ranging from data augmentation to the class probability for each class of objects, represented in Fig. 1. Moreover, the dataset considered in the current study consists of all categories of the bone fractures include simple fracture, compound fracture, comminuted fractures, greenstick fractures, and stress fractures. The model is trained with divergent data with different categories of the bone fractures across all the body parts. The entire dataset is utilized for both training and validation of the YOLO model. Using the full dataset in experimentation enhances the model's ability to learn and adapt to the dynamic aspects of X-ray images, leading to improved performance.

The YOLO V10 model is comparatively faster in performing the classification, when compared to that of the earlier versions of YOLO. YOLO V10 efficiently handles smaller objects with a better feature pyramid network. The other major changeover is that the proposed model can handle the variable size input and images of divergent natures[11]. Furthermore, the YOLO model is being evaluated concerning data augmentation and divergent hyperparameters. This could ultimately lead to improvised accuracy and precision working with X-ray images. The following are the contributions of the study.

- Data augmentation is performed on the bone X-ray images to scale the input images, sharpen them, and enhance the contrast of greyscale images. The processed images are used with the model for training, and validation.
- The YOLO V10 model is being evaluated with standard hyperparameters to recognize bone fractures from the X-ray images.
- The YOLO model is analyzed concerning divergent hyperparameters in the process of identifying bone fractures.

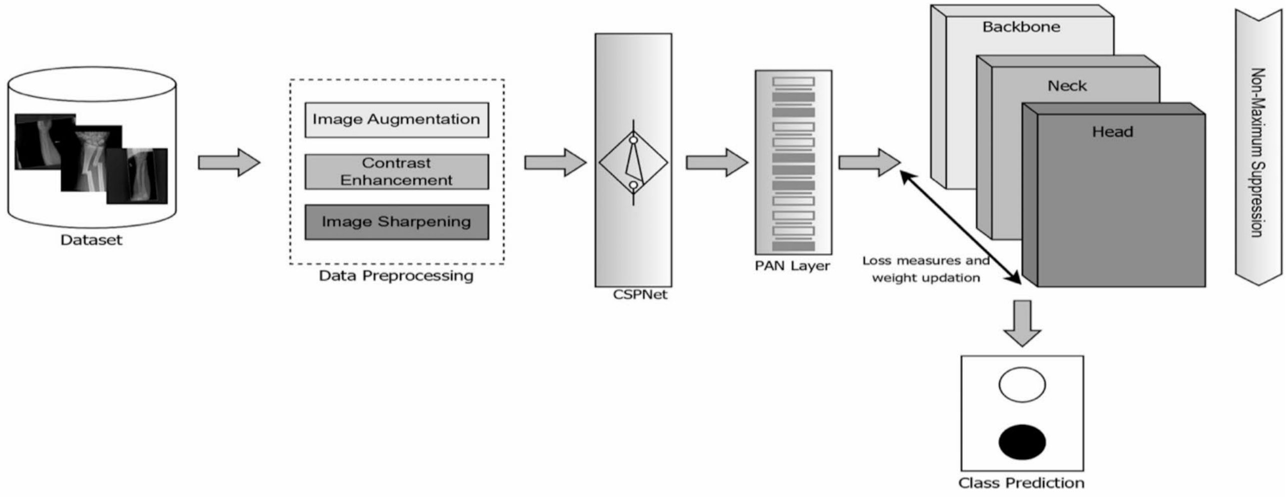


Fig. 1. The block diagram of the YOLO V10-based bone fracture detection module.

- The ablation study is performed on the YOLO V10 model without performing the data augmentation to analyze the robustness of the model in handling the raw data.

The rest of the manuscript is organized in the following manner. Section 2 presents the literature on deep-learning models used in bone fracture detection. Section 3 presents the background section, which outlines the details of the dataset, implementation environment, and the standard hyperparameters used in the evaluation process. Section 4 presents the YOLO V10 model and various components of the architecture. Section 5 outlines the results and the statistical analysis of the resultant outcome. Finally, Sect. 6 summarizes the findings of the study.

Literature review

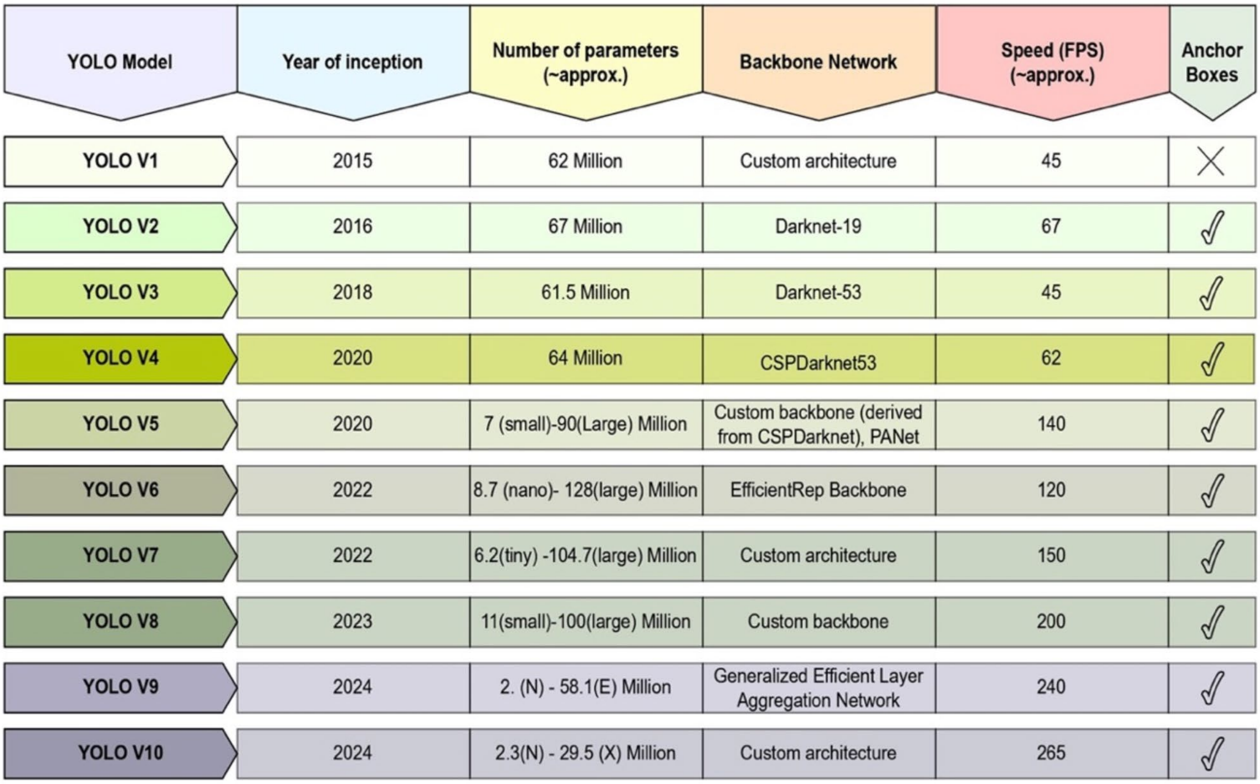
The current section presents the various existing models that are used in bone fracture detection and their challenges in the precise classification of the input data. Various machine-intelligent models are extensively used in classifying X-ray images for identifying bone fractures[12,13]. Some conventional approaches, such as neural networks (NN), Support Vector Machine (SVM), and decision tree (DT)[14], are also being used in the classification process. Later, the deep learning model based on the convolutional neural network (CNN), Region-based Convolutional Neural Networks (R-CNN) was extensively used to identify bone fractures from the X-ray images. Furthermore, several researchers have investigated the use of data augmentation and transfer learning to enhance model performance, particularly in cases with limited labeled data. The rising amount of data demonstrates the potential of machine-intelligent approaches to aid clinical decision-making and enhance the treatment processes.

A study by Dlshad Ahmed, Kosrat, and Roojwan Hawezi[15], analyzing a dataset of 270 X-ray images for identifying bone fracture using the machine learning algorithms like Naïve Bayes, Decision Tree, Nearest Neighbors, Random Forest, and SVM algorithm achieved the highest accuracy, ranging from 0.64 to 0.92 across different methods. Another study by S. Karimunnisa et al.[16] automatic X-ray fracture detection method using a back propagation neural network (BPNN) combined with conservative smoothing filtering and segmentation using the Canny edge approach, achieving an accuracy of 91%, which outperforms SVM and other techniques. A study by A. Y. Yang et al.[17] has evaluated a two line-based fracture detection technique, Standard line-based detection, and Adaptive Differential Parameter Optimized (ADPO) algorithm, which have been developed to identify fractured lines in X-ray images by extracting features that differentiate fractured from non-fractured lines. The ADPO scheme optimizes the Probabilistic Hough Transform parameters for better detection, achieving an average accuracy of 72.89%, compared to 71.57% for the Standard scheme, and both methods utilize an Artificial Neural Network (ANN) for classification.

Ma, Yangling, and Yixin Luo[18] have proposed a Crack-Sensitive Convolutional Neural Network (CrackNet), which utilizes a two-stage system combining Faster R-CNN to detect 20 different bone regions in X-ray images, followed by CrackNet to determine if these regions are fractured. Tested on 1,052 images, including 526 fractured cases, the system achieved an accuracy of 90.11% and an F-measure of 90.14%, outperforming other two-stage systems. A transfer learning approach utilizing a Faster R-CNN[19] DL model with a Region Proposal Network (RPN) was implemented for fracture detection and classification. The model, retrained using Inception v2 architecture on 50 X-ray images over 40,000 steps, achieved an overall accuracy of 94% in classifying images as fractured or non-fractured and accurately locating fractures with rectangular bounding boxes. Another study has deployed the Deep Neural Network (DNN)[20] model, developed to classify fractured and healthy bones, with data augmentation techniques applied to address overfitting on a small dataset. Using Softmax and Adam optimizers, the model achieved a classification accuracy of 92.44% using 5-fold cross-validation and over 95% and 93% accuracy on 10% and 20% test data. In another study on Combining a Growing Neural Gas (GNG) network[21] with eight VGG-inspired models, the method achieves superior sensitivity (95.86%) and specificity (99.63%) for bone classification, and 92.50% and 92.12%, respectively, for abnormality detection, while significantly reducing computation time compared to state-of-the-art models. Another study employs a MobileNet CNN model[22] to classify bone fractures with 98% accuracy using a dataset of 4906 X-ray images, demonstrating the effectiveness of deep learning in early and accurate fracture detection, which is crucial for improving patient outcomes.

Some of the most recent advancements in object detection using the DL models, like YOLO models, are used in bone fracture detection. A study on the YOLO Model compares one-stage and two-stage deep learning architectures for detecting four types of fractures using well-annotated datasets, focusing on YOLO variants, SSD, Faster-RCNN, and Mask-RCNN models. The customized YOLOv7-ATT model[23], enhanced with an attention mechanism and an Enhanced Intersection of Unions (EIoU) loss function, achieved superior performance, with a mAP of 86.2% on the FracAtlas dataset, demonstrating its potential for clinical fracture detection. In another study on YOLO, A hybrid model combining YOLO-Neural Architecture Search(NAS)[24], EfficientDet, and DETR3 was developed to detect hand bone and joint fractures in X-ray images using a dataset of 4,736 images classified into six types. The model's performance was evaluated by comparing it with existing algorithms, demonstrating its effectiveness in accurate fracture detection. The proposed YOLO-NAS model has outperformed the other DL models with an accuracy of 98.1%. The details of various YOLO versions concerning the number of parameters, backbone network, and speed are summarized in Fig. 2.

There is a significant demand for a DL approach that can precisely classify the abnormality from the low resolution and raw X-ray images[25], where the YOLO version 10 is considerably superior over the earlier versions of YOLO[26]. YOLO V10 model would no longer use the non-maximum suppression (NMS) during post-processing. Rather, it uses dual label assignments to provide a harmonic combination of accuracy and speed by guaranteeing that the model stays computationally efficient while collecting crucial features for precise classification.



YOLO Model	Year of inception	Number of parameters (~approx.)	Backbone Network	Speed (FPS) (~approx.)	Anchor Boxes
YOLO V1	2015	62 Million	Custom architecture	45	✗
YOLO V2	2016	67 Million	Darknet-19	67	✓
YOLO V3	2018	61.5 Million	Darknet-53	45	✓
YOLO V4	2020	64 Million	CSPDarknet53	62	✓
YOLO V5	2020	7 (small)-90(Large) Million	Custom backbone (derived from CSPDarknet), PANet	140	✓
YOLO V6	2022	8.7 (nano)- 128(large) Million	EfficientRep Backbone	120	✓
YOLO V7	2022	6.2(tiny) -104.7(large) Million	Custom architecture	150	✓
YOLO V8	2023	11(small)-100(large) Million	Custom backbone	200	✓
YOLO V9	2024	2. (N) - 58.1(E) Million	Generalized Efficient Layer Aggregation Network	240	✓
YOLO V10	2024	2.3(N) - 29.5 (X) Million	Custom architecture	265	✓

Fig. 2. A summary of various versions of the YOLO model is provided.

Approach	Dataset	Observations	Remarks
Faster R-CNN[27]	Gazi University Hospital's dataset of wrist X-ray images	The model has yielded an average precision of 0.61 and an average recall of 0.36	Faster R-CNN is computationally expensive due to its two-stage detection process, which includes the region identification followed by an object detection, which makes it slower for real-time applications.
Dynamic R-CNN[27]	Gazi University Hospital's dataset of wrist X-ray images dataset of 2075 images	The model has yielded an average precision of 0.77 and an average recall of 0.41	The dynamic R-CNN model is also computationally expensive and susceptible to overfitting of the model. Lesser generalizability in case of smaller-size datasets.
Ensembled DL (MobileNetV2, Vgg16, InceptionV3, and ResNet50)[28]	Mura-v1.1 dataset has 6,542 radiographs images.	The model yielded an accuracy of 0. 92 and a precision of 0.91	The combination of all models makes it challenging to interpret the decision process and complex hyperparameter tuning process. Moreover, the ensemble models need larger and diverse datasets for desired performance.
MobileNet V2[29]	Bone fracture dataset, Mendeley Data V1.	The model has yielded an accuracy of 93.5%	MobileNet V2 model is a lightweight model, it finds it difficult to capture complex features and subtle details in high-resolution X-ray images
VarifocalNet Feature Pyramid Network[30]	Dataset with 823 hip radiographs images across 150 subjects.	The model has yielded an accuracy of 0.94, sensitivity of 0.95, and specificity of 0.94	The VarifocalNet fails to handle the class imbalance and is susceptible to overlapping objects in complex scenes.
MobLG-Net[31]	Fracture detection using an x-ray images dataset that consists of 9,463 X-ray images of fractured and normal bones	The model has attained an accuracy of 0.97 when evaluated with random forest.	MobLG-Net struggles with balancing speed and accuracy, especially for efficiently detecting small or subtle objects.
CrackNet[32]	Radiopaedia dataset with 3053 X-ray images.	The model has attained an accuracy of 0.883 accuracy and a precision of 0.890	CrackNet struggles with detecting fine or irregular cracks, and requires large datasets. The model performs poorly on noisy images.

Table 1. Various existing models are used in bone fracture detection.

Furthermore, various studies used in bone fracture detection, along with their observations are summarized in Table 1.

Background

The current section of the manuscript discusses the various aspects, like the dataset used in the current study and the data augmentation techniques like sharpening and contrast enhancement of greyscale images. The details of the implementation environment are also discussed in the current section of the manuscript. The components of the image pre-processing are illustrated in Fig. 3.

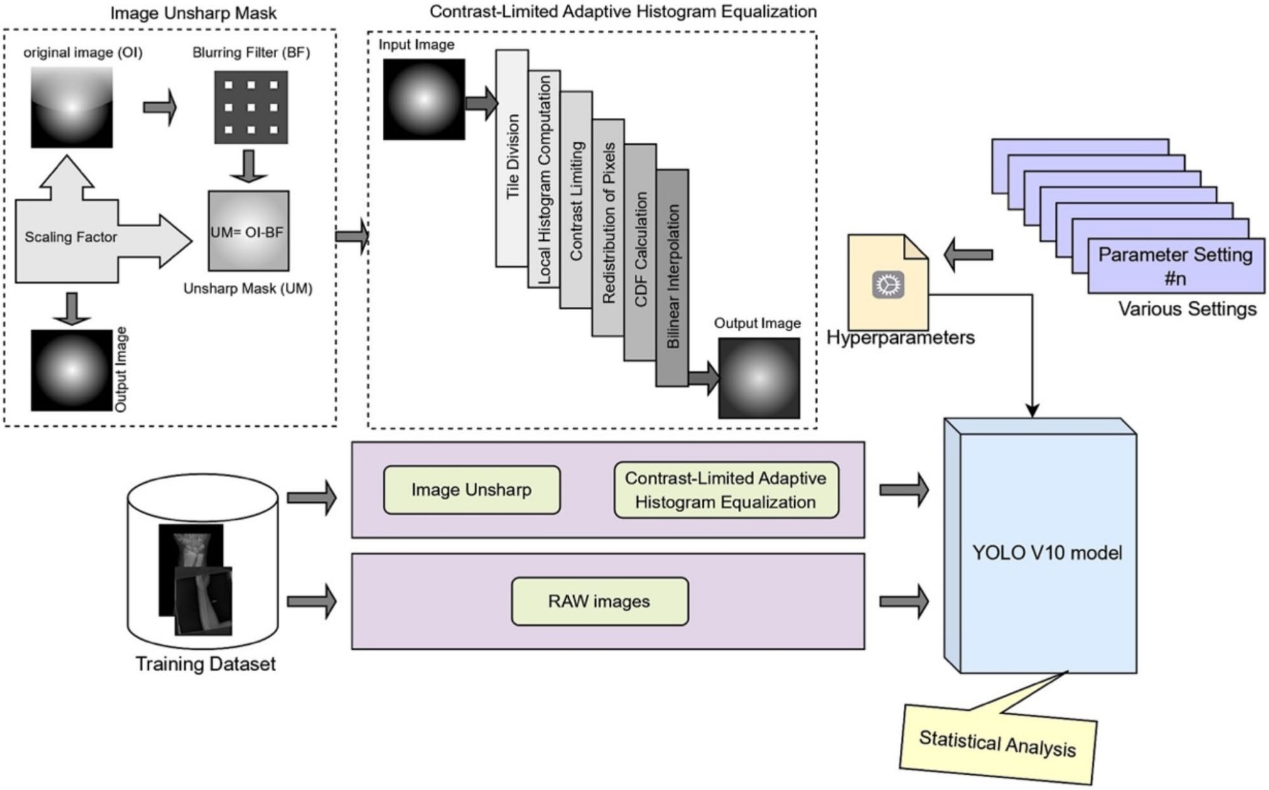


Fig. 3. The workflow of the YOLO V10 model for bone fracture detection.

Dataset description

The current study uses the Bone Fracture Detection: Computer Vision Project dataset[33], which is the fourth version of the bone fracture detection daoon dataset, an open-access dataset by roboflow universe search. The complete dataset consists of 4148 images of various body parts that includes wrist, arm, finger, leg, and ankle fracture images distributed across seven classes that include Elbow Positive, Fingers Positive, Forearm Fracture, Humerus Fracture, Shoulder Fracture, and Wrist Positive. The dataset is further splitted as the training, validation, and testing datasets in a ratio of 88,8 and 4 percentages respectively, by the dataset owner. Resultantly, 3631 images are associated with the training set, 348 images are used to validate the model, and 169 for testing the model's efficiency. The images in the dataset are raw greyscale images, where no additional data augmentation or pre-processing is performed on them. The images are annotated with the ground truth, i.e., with the corresponding class label. The split of the dataset is not being modified in the current study, the same split is considered for evaluation. The distribution of the classes is shown in Fig. 4, where it can be observed that there are more instances for the fingers positive class and significantly less samples in wrist positive class, which indicates the dataset is imbalanced. The sample images in the dataset are presented in Fig. 5.

Data augmentation

The original data in the dataset is being processed to enhance the contrast and edge-related information by sharpening the image. The edge-related enhancements are done through the Unsharp Masking mechanism[34], and contrast enhancement is done through the Contrast-Limited Adaptive Histogram Equalization (CLAHE) approach[35]. A detailed explanation of each model is presented in the sub-section discussed below. Histogram equalization is performed across the entire dataset, which would assist in better classification by suppressing the noise and improvising the local contrast of the image regions. Affine transforms are performed across 40% of an overall dataset by transforming 40% in each portion of the dataset. The same has been presented in Fig. 6, showing the number of instances being processed. Affine transforms would improve the model generalization by dynamic feature selection by transforming the image.

Image unsharp masking approach

Image unsharp masking is one of the most widely used image sharpening techniques. The approach adds a scaled version of the image's high-frequency components, like the edge, back to the original image. The high-frequency data is derived by subtracting a blurred image variant from the original. The outcome highlights regions of intensity change in the image, resulting the edges looking sharper[36]. The output image denoted by O_s by updating the image intensity value I is shown in Eq. (1).

$$O_s\left(x,y\right)=I\left(x,y\right)+\gamma\ \left(I\left(x,y\right)-I_b\left(x,y\right)\right) \tag{1}$$

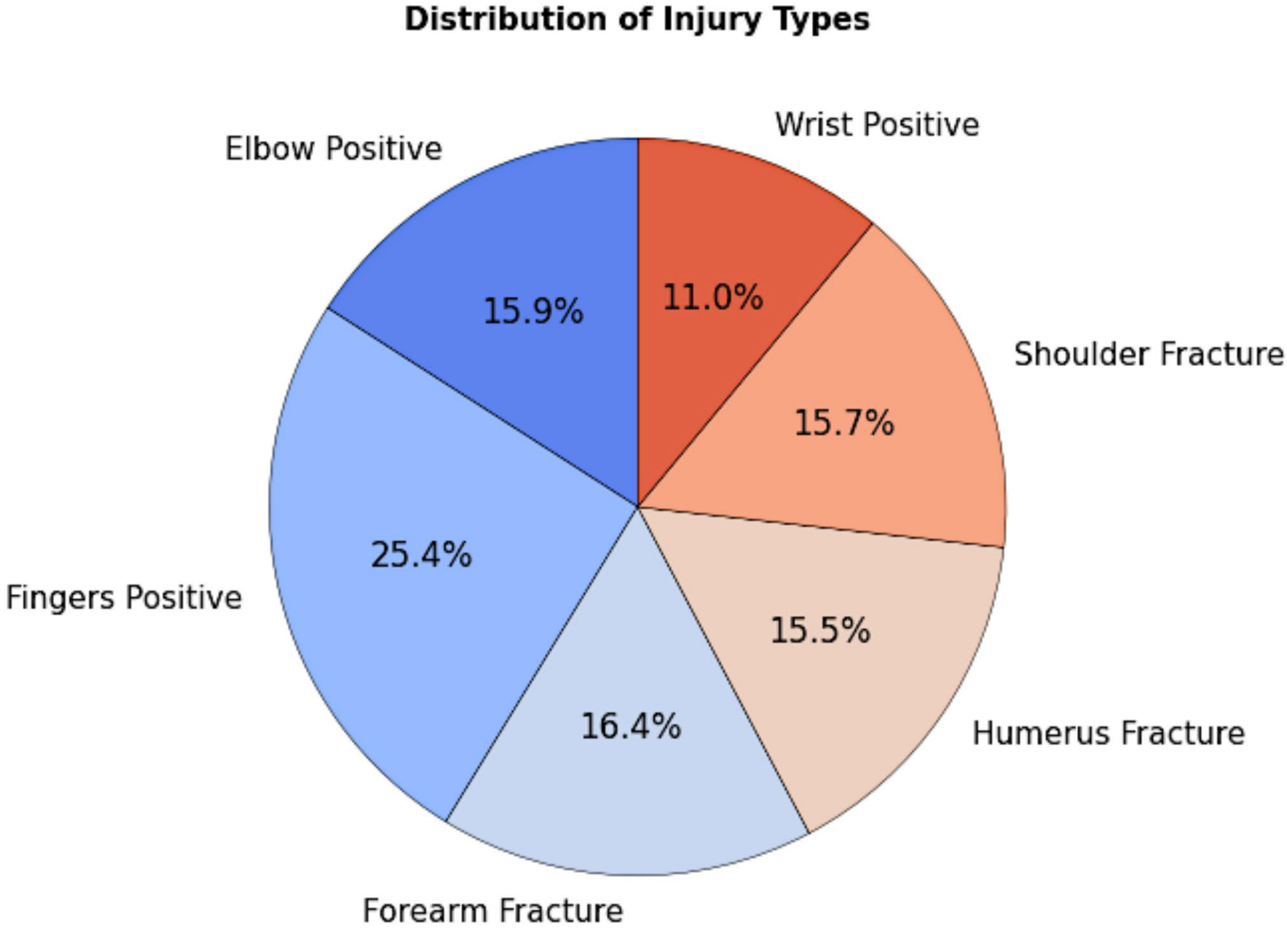


Fig. 4. Graph representing the distribution of classes in the dataset.

From the equation, the notation $I\left(x,y\right)$ designates the intensity of the original greyscale image at the coordinated $\left(x,y\right)$. The $I_b\left(x,y\right)$ designates the intensity of the blurred image at the coordinates $\left(x,y\right)$, and the notation γ is the scaling factor that determines the magnitude of the sharpening. The corresponding input image and the output image, along with the histogram values of intensities, are presented in Fig. 7. In this approach, a blurred version of the original image is removed to highlight the edges, and the result is then returned to the original image.

Contrast-Limited adaptive histogram equalization

The low-contrast nature of X-ray images makes it difficult to recognize minor fractures and inconsistencies. So, contrast augmentation is required. Traditional histogram equalization changes the contrast of a whole image by distributing intensity values to obtain a uniform histogram. But, this approach may result in over-enhancement and noise amplification, particularly in areas with uniform brightness. Adaptive Histogram Equalization (AHE)[37] improves image contrast by partitioning the image into smaller segments and performing histogram equalization to each segment independently. This technique enhances local contrast much better than a simple histogram equalization approach but may considerably increase noise in areas with poor contrast. Contrast limiting (CL) will restrict the contrast improvement for each tile to prevent noise amplification. When a histogram bin's height exceeds the specified threshold, extra pixels are reallocated across the histogram, limiting the contrast. The contrast-limited adaptive histogram equalization approach would limit the excessive noise enhancement in the lower-intensity regions.

Let the AHE be applied over the image intensity value designated by I, which is divided into $p \times q$ non-overlapping tiles in the image. The local histogram is calculated for every region, and the intensity values are adjusted, as shown in Eq. (2).

$$I'\ \left(x,y\right) = CD\left(I\left(X,Y\right)\right) \times\ \left(N-1\right) \tag{2}$$

From the equation, the function $CD\left(\right)$ is the cumulative distribution function over the local histogram, the $\left(x,y\right)$is the pixel coordinates and $I\left(X,Y\right)$ is the intensity of the original image, and $I'\ \left(x,y\right)$ is the calculated intensity value. CLAHE restricts contrast enhancement to avoid excessive noise amplification. The maximum height of the histogram bins is restricted to a set clip limit M_T. Excess pixels are redistributed throughout all

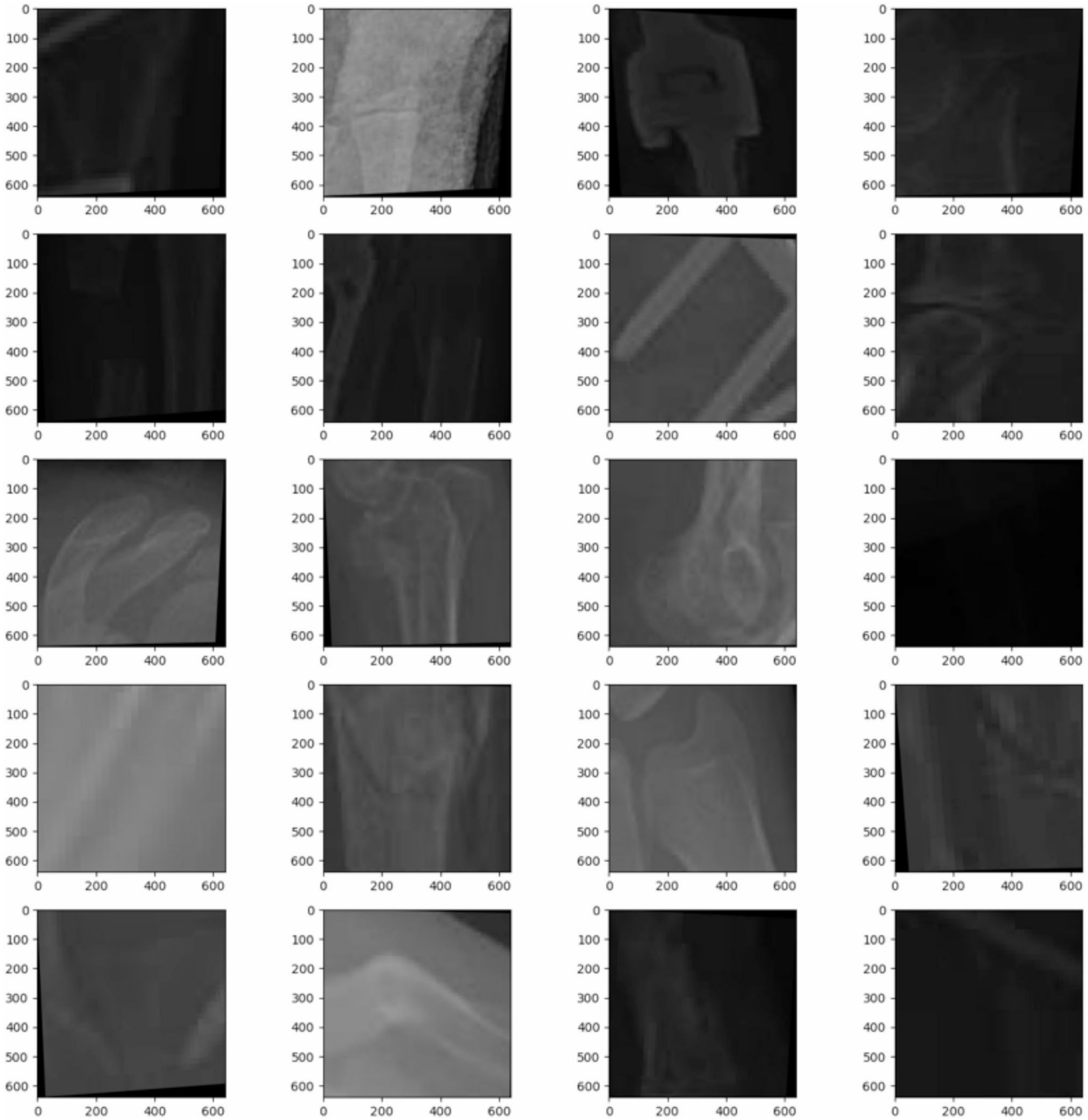


Fig. 5. Sample images that are obtained from the dataset.

bins, preserving the histogram level across the image pixels. The pixel adjustment is done based on the following criteria as shown in Eq. (3).

$$H\prime_{b} = \begin{cases} M_T & if\ H_v > M_T \\ H_b & Otherwise \end{cases} \tag{3}$$

From the equation, the H_b actual histogram of the bin b and $H\prime_{b}$ designates the updated histogram after the clipping. The CLAHE approach uses bilinear interpolation (BI) to deal with the discontinuities at the tile boundaries. BI combines the histograms of the adjacent tiles. Let us assume the pixel p at the coordinates (x, y) , the measure of interpolated intensity $II(x, y)$ is calculated as shown in Eq. (4).

$$II(x, y) = (1 - \Delta x)(1 - \Delta y) I_a + \Delta x (1 - \Delta y) I_b + (1 - \Delta x) \Delta y I_c + \Delta x \Delta y I_d \tag{4}$$

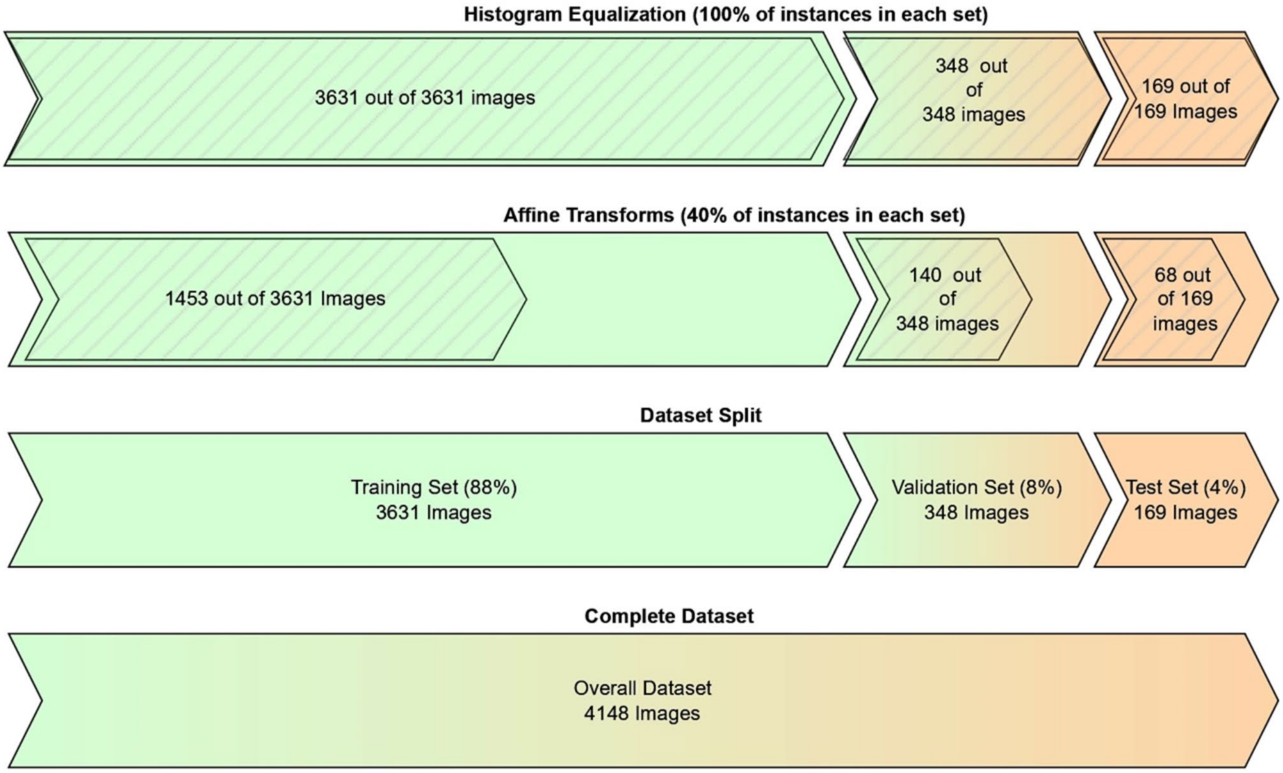

Histogram Equalization (100% of instances in each set)
3631 out of 3631 images
348 out of 348 images
169 out of 169 Images
Affine Transforms (40% of instances in each set)
1453 out of 3631 Images
140 out of 348 images
68 out of 169 images
Dataset Split
Training Set (88%) 3631 Images
Validation Set (8%) 348 Images
Test Set (4%) 169 Images
Complete Dataset
Overall Dataset 4148 Images

Fig. 6. Dataset Samples and number of instances used in data augmentation.

From the equation, the $\varDelta x$ and $\varDelta y$ values designate the relative distances of the pixel from the tile boundaries. The notations I_a, I_b, I_c and I_d are the equalized value of their four surrounding tiles. The resultant outcome and the associated histogram values of the pixels are shown in Fig. 8.

Experimental setup

The current section of the manuscript outlines the experimental setup used in the current study. The execution is performed on the online platform based on Kaggle for training and evaluation of the model. The details of the standalone computer, the packages used, and the details of the graphics processing unit (GPU) used in the current study are in Table 2. All the experimentation analysis is done over the same platform in the Kaggle using the same GPU configuration.

Methodology and experimental design

The current section of the manuscript discusses the YOLO v10 model for classifying the abnormalities in the X-ray images to localize the fractures in the bones. The model has various components, like the backbone and neck.

Backbone component

The backbone component of the YOLO V10 model used the Cross Stage Partial Network (CSPNet)[38], enhancing the model's computational efficiency and accuracy. CSPNet minimizes redundancy in feature maps by facilitating a more efficient gradient flow, which is especially beneficial in deep learning models aimed at object detection. This framework is particularly efficient in balancing computing efficiency by retaining knowledge at more profound levels of the model. CSPNet reduces the number of parameters and processes to just a feature map segment, resulting in a lightweight network[39]. CSPNet integrates features from both shallow and deep layers, enhancing the model's capacity for effective object detection via more complex and diversified feature representations. The corresponding figure of CSPNet architecture is shown in Fig. 9.

Neck component

The neck component of the YOLO V10 model uses the Path Aggregation Network (PAN)[40] for the purpose of a multi-scale feature fusion technique. PAN is an upgrade of feature pyramid networks (FPN) that incorporates a bottom-up approach with the conventional top-down pathway used in FPNs. This framework facilitates a bidirectional exchange of information, integrating high-level semantic attributes from deeper levels with low-level spatial data from shallower layers. The multiscale feature aggregation (MSFA) is used to detect objects of varying sizes, it records both low-level spatial information and high-level semantic information. The PAN model consists of bottom-up and top-down approaches, allowing cross-scale connections between features. The

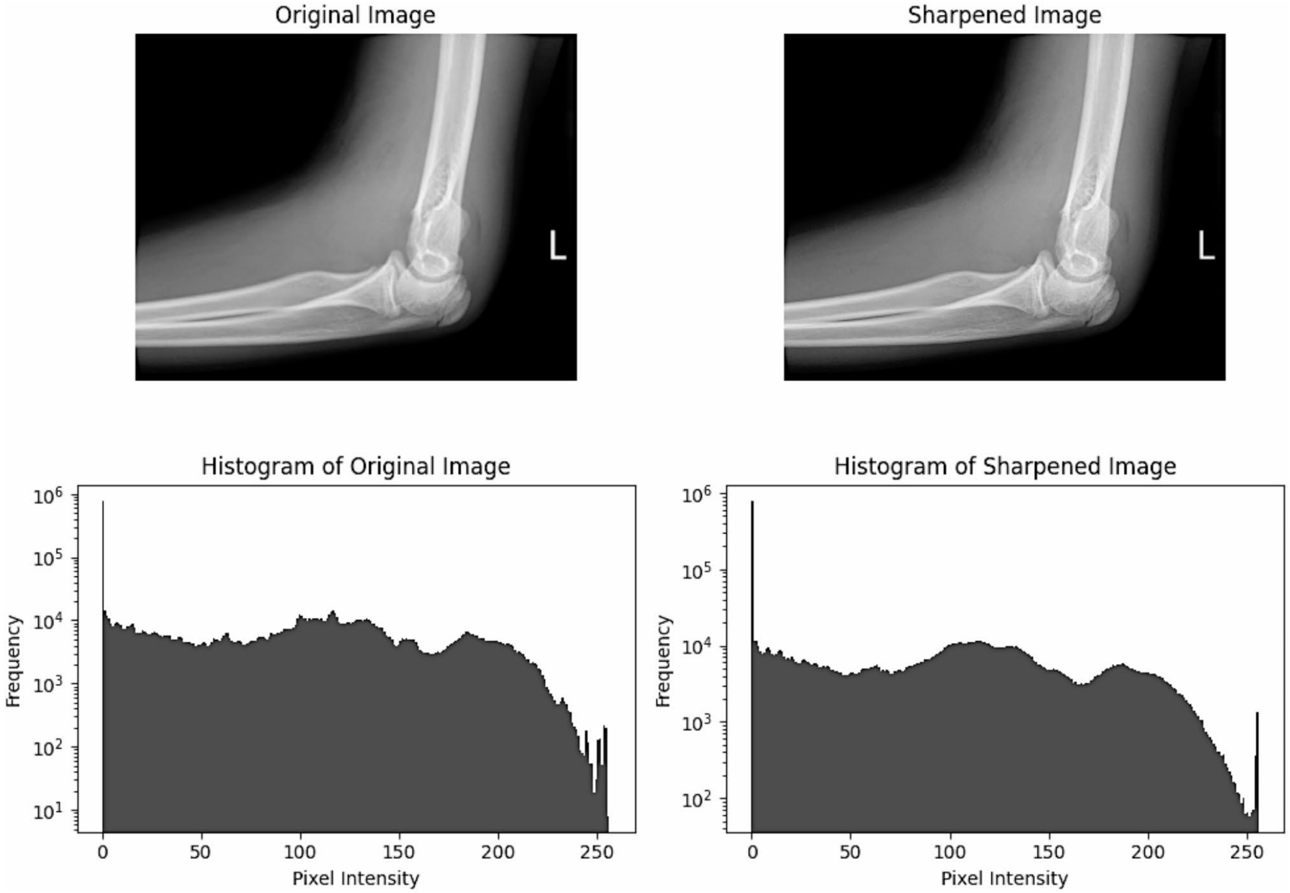


Fig. 7. The input and output images and histogram values of image unsharp masking approach.

top-down approach would localize the features from the deeper layer to shallow layers. Similarly, the bottom-up approach would localize the features from the shallow layers that are propagated upwards to recognize the semantic features. The neck components of YOLO V10 are illustrated in Fig. 10.

The top-down feature map is represented by f_x^{td} over the layer x evaluated by up-sampling the feature map from the previous layer f_{x+1}^{td}, which is combined with the current layer as shown in Eq. (5).

$$f_x^{td} = Bilinear\_Upsample\left(f_{x+1}^{td}\right) + f_x \tag{5}$$

Similarly, the bottom-up stage improves feature propagation by aggregating information in the opposite way. The corresponding formula is presented in Eq. (6), over the feature next stage identified by f_{x-1}^{bu}.

$$f_x^{bu} = Pooling\_downsample\left(f_{x-1}^{bu}\right) + f_x^{td} \tag{6}$$

The features of both the top-down feature map and the bottom-up stage are being concatenated to retain both fine details and semantic context, as shown in Eq. (7).

$$f\prime_{x} = Concat(f_x^{td}, f_x^{bu}) \tag{7}$$

After concatenation, 1 × 1 convolution is used to reduce dimensionality and improve the combined feature maps. PAN architecture reduces redundancy, allowing for computational efficiency without losing the accuracy.

Head component

The head component is the place where the boundary box and class classification and approximates the confidence score[41]. This head component consists of the one-to-one head and the one-to-many head. The corresponding figure of the head component is presented in Fig. 11.

The one-to-one head would assist in assigning the anchor box to each unique ground truth object in the image. The one-to-one head is efficient in real-time object detection as it minimizes the overlap of the bounded region, reducing the computing cost of managing redundant detections. The one-to-one head assigns each anchor box directly to a single bounding box to the object. The mathematical formulation for each anchor box

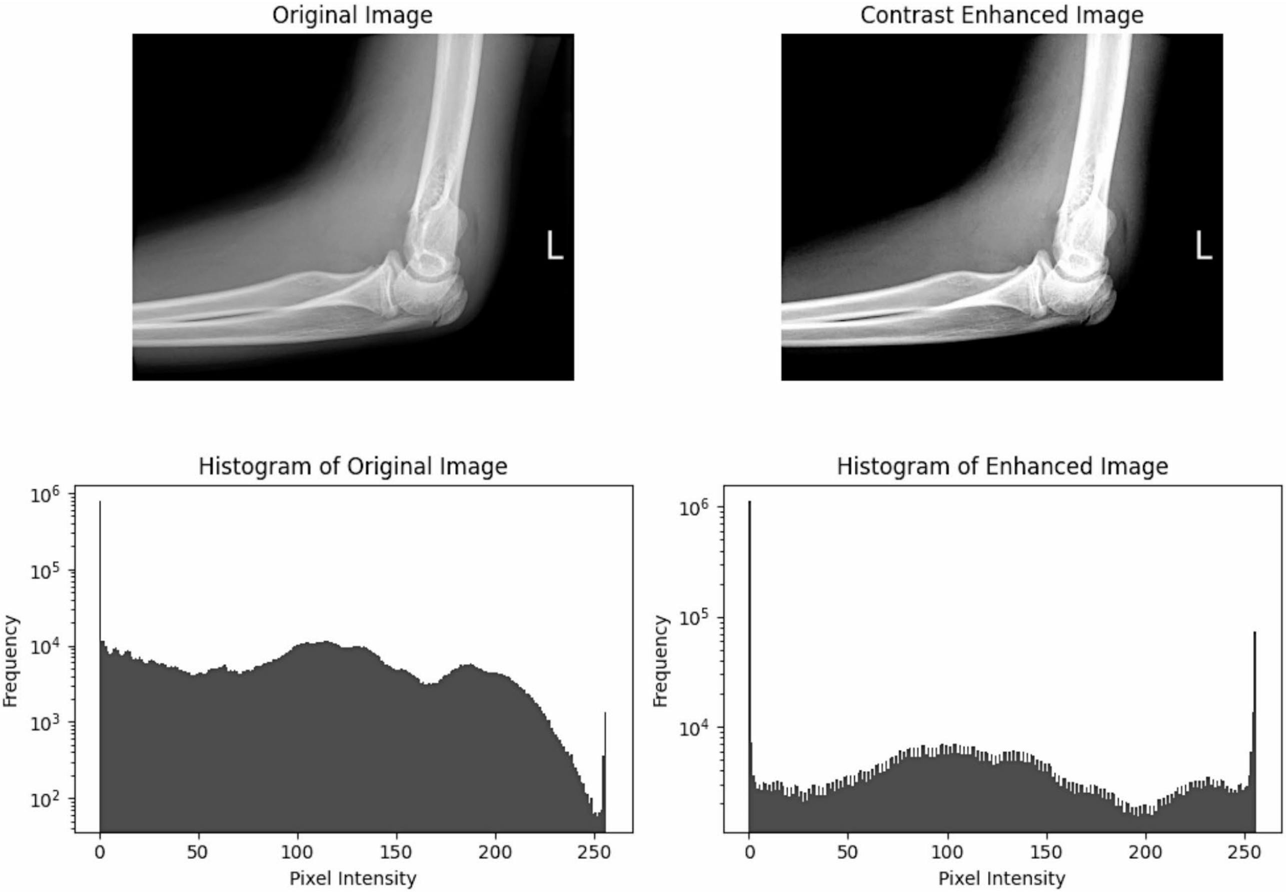


Fig. 8. The input and output images and histogram values of image unsharp masking approach.

Environment Details	**Specifications**
Machine	Laptop
Make	Lenovo
Model	ThinkPad T16 Gen 2
Processor	AMD Ryzen 5 PRO 7545U Processor
Online Platform	Kaggle
GPU	GPU T4 × 2
Coding Language	Python: V 3.10.2
Libraries Used	PyTorch: V 1.9.0 Torchvision: V 0.10.0 Ultralytics: V 8.3.49 Pandas: V 1.3.5, NumPy: V 1.21.4, Matplotlib: V 3.4.3

Table 2. Details of the implementation environment.

A_x for each corresponding nearest bounded box B_j, where the corresponding assignment function is shown in Eq. (8).

$$\mathcal{B}\left(A_x, B_j\right) = \arg\_min_j\ IOU(A_x, B_j) \tag{8}$$

From the above equation, the notation $\mathcal{B}(\cdot)$ represents the bounded box assignment function, and IOU represents the intersection over the union metric. The confidence score C_i for each anchor is assessed based on the chance that A_x relates to an actual object, and the same was approximated using the sigmoid function, as shown in Eq. (9).

$$C_x = \sigma\ \left(lin\left(f_x\right)\right) \tag{9}$$

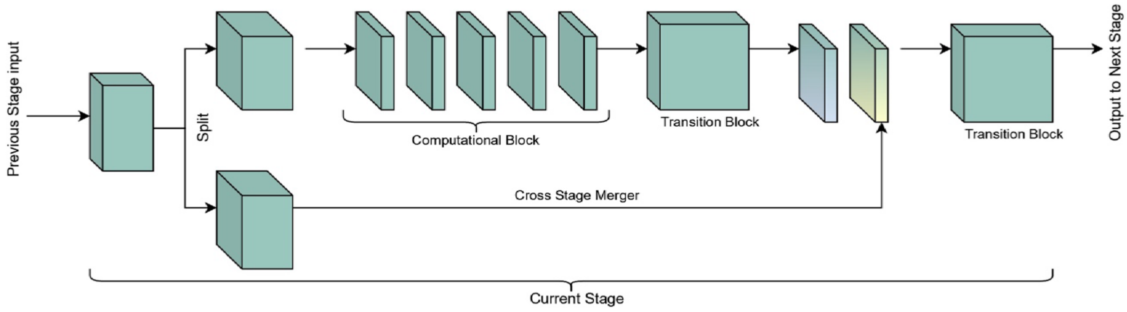


Fig. 9. The architecture diagram of CSPNet.

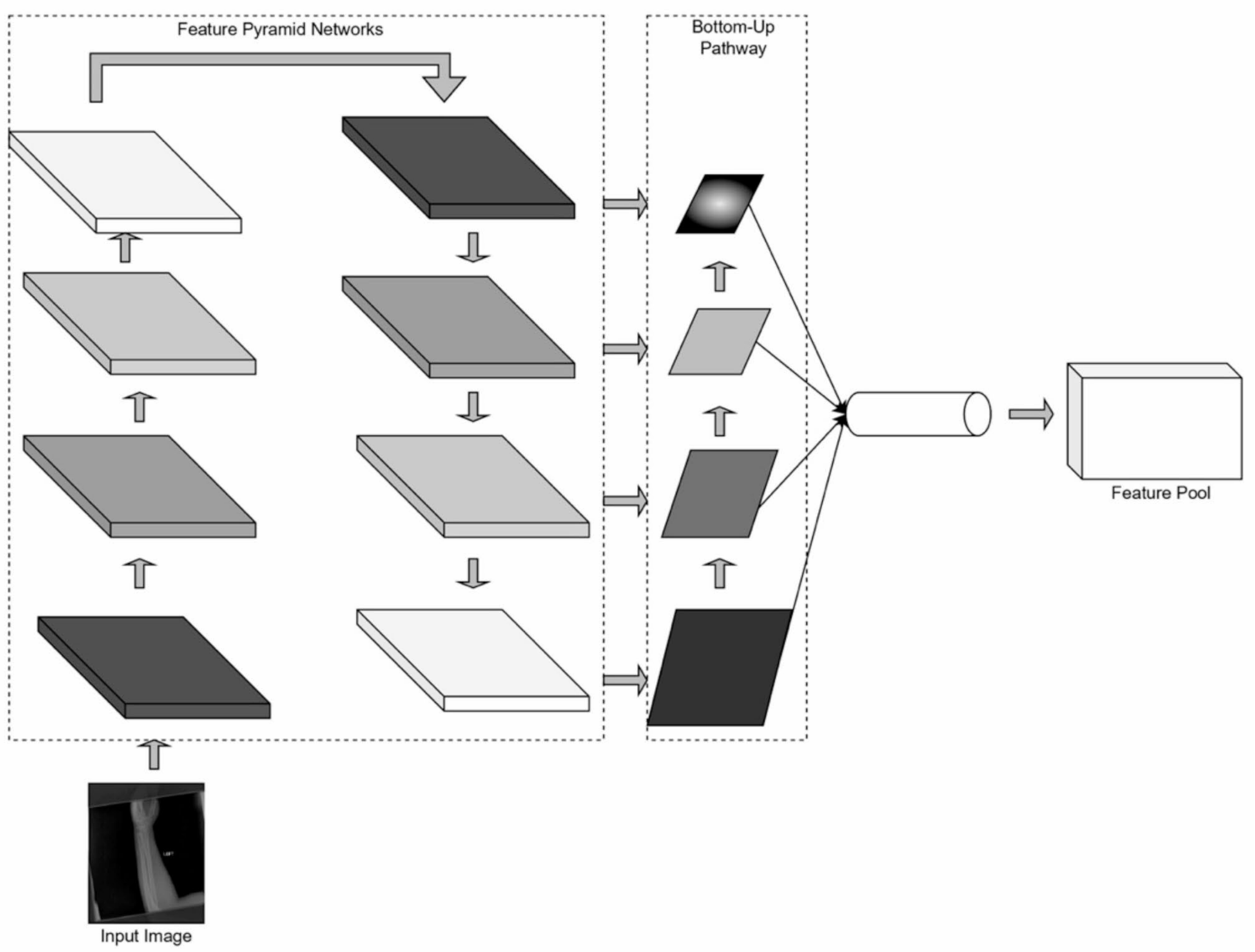


Fig. 10. The architecture diagram of the Path Aggregation Network.

From the above equation, the notation f_i designates the feature vector for the anchor x. The bounded box regression for the anchor x in the one-to-one head is represented over the coordinates (p_x, q_y) and dimensions (s_x, t_y) of A_x. The four learned offsets adjustments $(\Delta\, p_x,\ \Delta\, q_x,\ \Delta\, s_x, \Delta\, t_x)$ to the assigned objects, which are shown in the Equations (10) through (13).

$$p_x' = p_x + \Delta p_x \tag{10}$$

$$q\prime\ _x = q_x + \Delta\, q_x \tag{11}$$

$$s\prime\ _x = s_x \cdot\ exp(\Delta\, s_x) \tag{12}$$

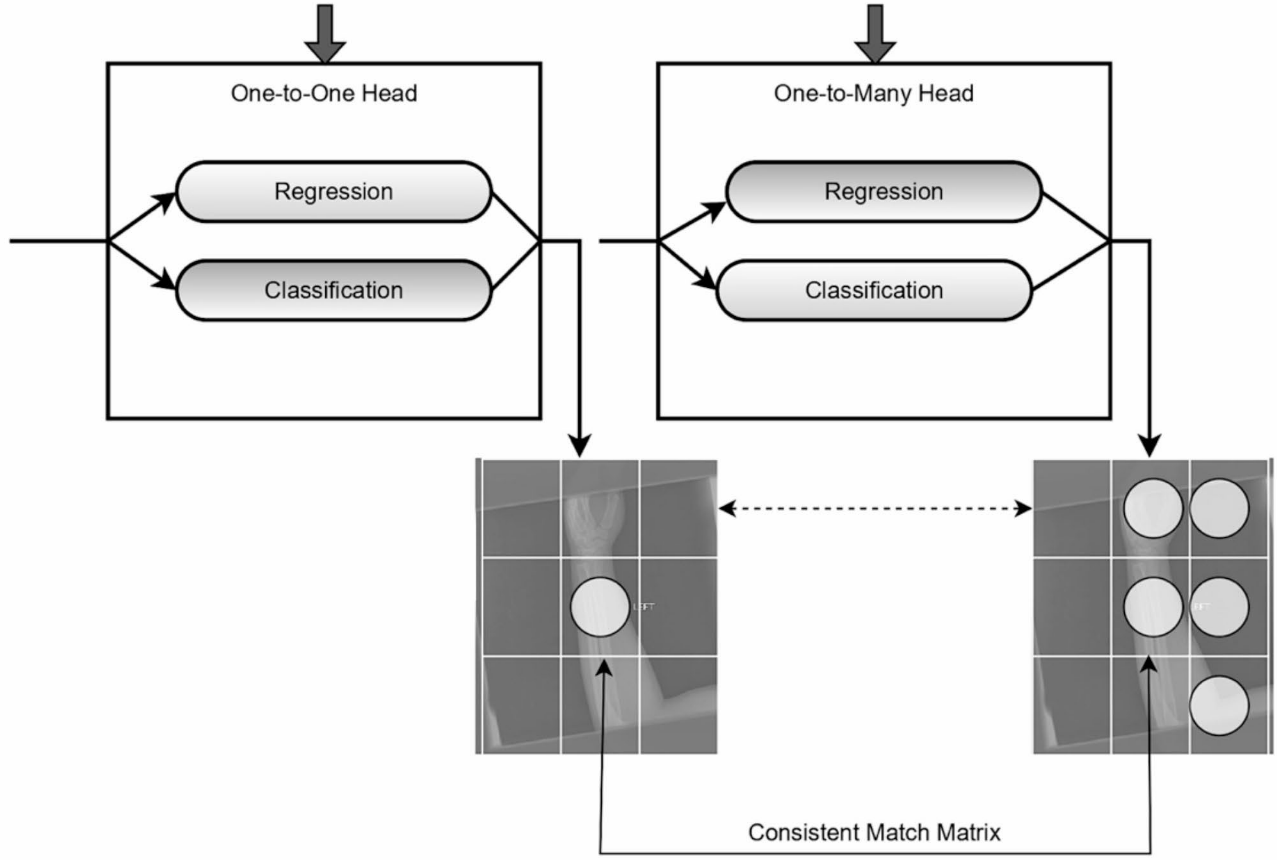


Fig. 11. The architecture diagram of the head component of the YOLO V10 model.

$$t\prime_{\,x} = t_x \cdot \; exp(\varDelta\, t_x) \tag{13}$$

The one-to-many head assigns an anchor box to different bounding boxes, allowing numerous detections per anchor. One-to-many head is useful in recognizing smaller or overlapping objects and is often employed in dense object identification tasks. The mathematical formulation for each anchor box A_x for each corresponding nearest bounded box B_j across multiple ground truths concerning the threshold T for Intersection over the Union (IOU) operation is presented in Eq. (14).

$$\mathcal{B}\,(A_x, B_j) = \{ j : IOU(A_x, B_j) \geq \; T\} \tag{14}$$

The class and confidence scores are assessed for each anchor-object match, and the formula for the confidence score C_{xj} is shown in Eq. (15) and class score k_{xj} is shown in Eq. (16).

$$C_{xj} = \sigma \;(lin\,(f_{xj})) \tag{15}$$

$$k_{xj} = softmax\,(l\omega_{\,c} \cdot \; f_{xj}) \tag{16}$$

Furthermore, the bounded box regression for each bounded box, over the offsets $\varDelta\, p_{xi}, \varDelta\, q_{xi}, \varDelta\, s_{xi}, \varDelta\, t_{xi}$ for each anchor box assignment is shown in Equations (17) through (20).

$$p'_{xi} = p_x + \varDelta p_{xi} \tag{17}$$

$$q\prime_{\,xi} = q_x + \varDelta\, q_{xi} \tag{18}$$

$$s\prime_{\,xi} = s_x \cdot \; exp(\varDelta\, s_{xi}) \tag{19}$$

$$t\prime_{\,xi} = t_x \cdot \; exp(\varDelta\, t_{xi}) \tag{20}$$

Hyperparameters and performance analysis

The current section discusses the standard hyperparameters that are being used to implement the proposed YOLO V10 model for bone fracture detection. The corresponding baseline parameters are discussed in Table 3. The hyperparameters are then fine-tuned to analyze the model's performance over divergent settings, each of which is recognized as configuration #1, configuration #2, and so on for all possible combinations.

Environment Details	Specifications
Learning Rate	0.01
Batch Size	64
Number of Epochs	100
Momentum	Stochastic Gradient Descent
Weight Decay	0.005
Optimizer	AdamW
Activation Function	SiLU
Warmup Epochs:	1.7
Gradient Clipping:	Yes

Table 3. Details of the standard hyperparameters used in the current study.

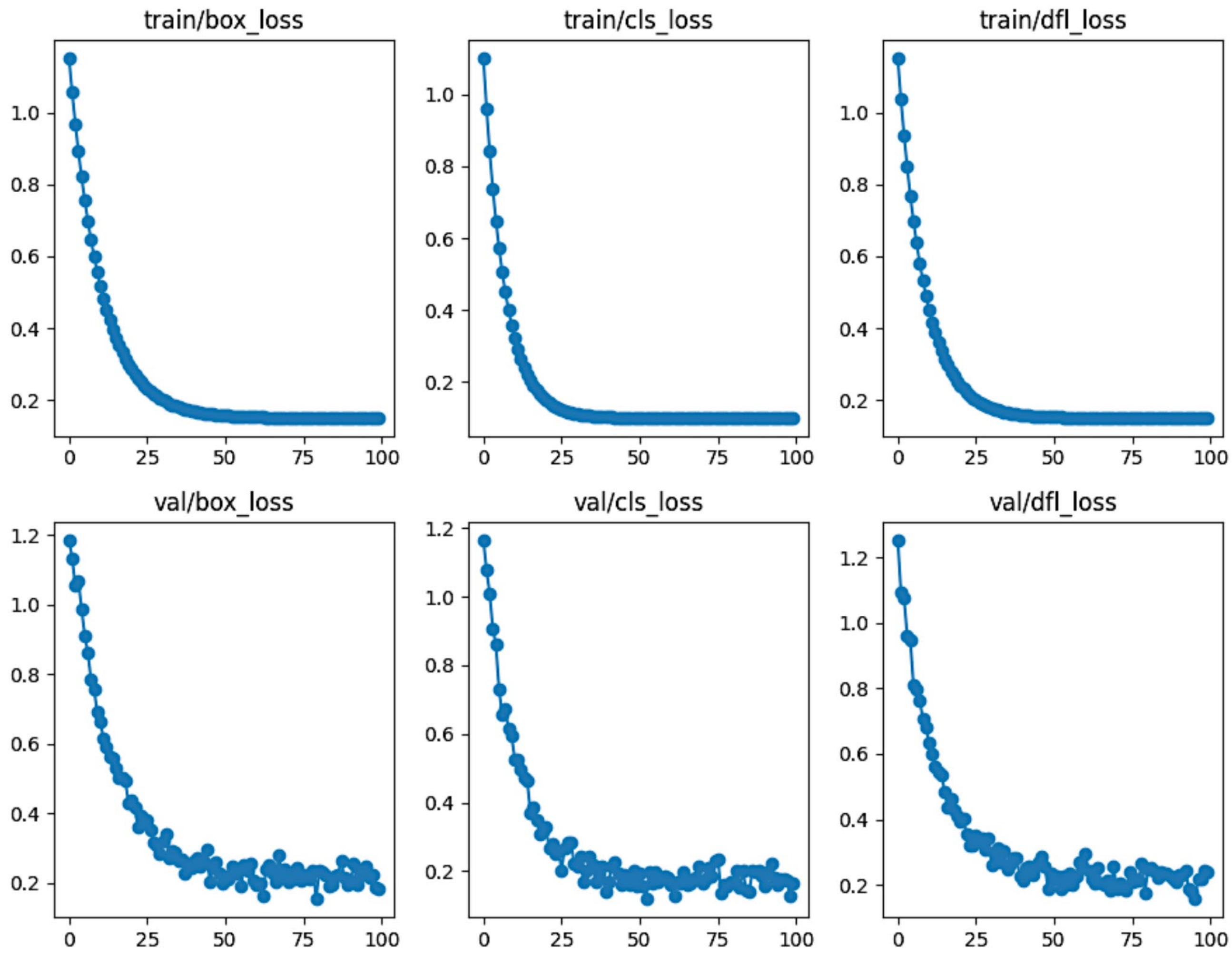


Fig. 12. The training and validation loss metrics were obtained on standard hyperparameters.

During the experimentation, the YOLO V10 model of bone fracture detection, the corresponding loss measures associated with box loss, class loss, and distribution of Focal Loss for both the training and testing phases are evaluated. The corresponding graphs obtained for the standard metrics are presented in Fig. 12.

However, the hyperparameters are being further tuned across multiple configurations to analyze the model's performance. The configurations are as follows, Configuration #1 the standard hyperparameters, remained the same except for the learning rate, where the learning rate is set to 0.001. Configuration #2 includes the standard hyperparameter values, except the learning rate, where the learning rate is set to 0.1. Configuration #3 includes the standard hyperparameter values, except the number of epochs. The number of epochs is set to 50. Configuration #3 includes the standard hyperparameter values, except the number of epochs, where the number of epochs is set to 150. The loss and accuracy graphs for each of these configurations are presented as follows, the graphs of configuration #1 are presented in Fig. 13, The graphs of configuration #2 are presented in Fig. 14, the graphs of configuration #3 is presented in Fig. 15, and the graphs of configuration #4 is presented in Fig. 16.

It can be observed from the loss and accuracy graphs that the model performed reasonably better in the standard hyperparameters discussed in Table 2. This is followed by Configuration #4, which has performed well, but the model has performed exceptionally in the training phase and struggled to generalize in the validation phase. In configuration #1, the accuracy has dropped due to a lower learning pace. The lesser accuracy in configuration #2 might be due to instability in the model. Configuration #3 needs more epochs to converge. Furthermore, the model is also being evaluated over the raw data without performing the data augmentation, the obtained loss graphs are being presented in Fig. 17.

It can be observed from Figs. 12 and 17 that the model performed slightly better on the pre-processed images than the raw image. However, there is no significant change in the performance of the model. It can be understood that the YOLO V10 model is performing well on all forms of data, as the model is able to perform reasonably well on the raw data.

Results and discussion

The current section outlines the experimental observations of the proposed YOLO V10 model for bone fracture detection. The baseline metrics like true positive, true negative, false positive, and false negative are being observed across multiple instances to determine the number of successful instances, where the model has identified the fractured images correctly, and also the number of unsuccessful instances, where the model has misinterpreted normal images as fractured instance and vice versa. The model has been evaluated across divergent metrics like precision, recall, F1-Score, and mean Average Precision (mAP)[42]. The performances are being analyzed for each configuration, and the corresponding values are presented in Table 4.

Furthermore, the model is evaluated on the raw images, i.e., without any data augmentation. The raw and pre-processed images are being processed using the same standard configuration. The observed results on the raw and augmented images over the standard metrics are presented as a confusion matrix[43] in Fig. 18. The observed results are presented in Table 5.

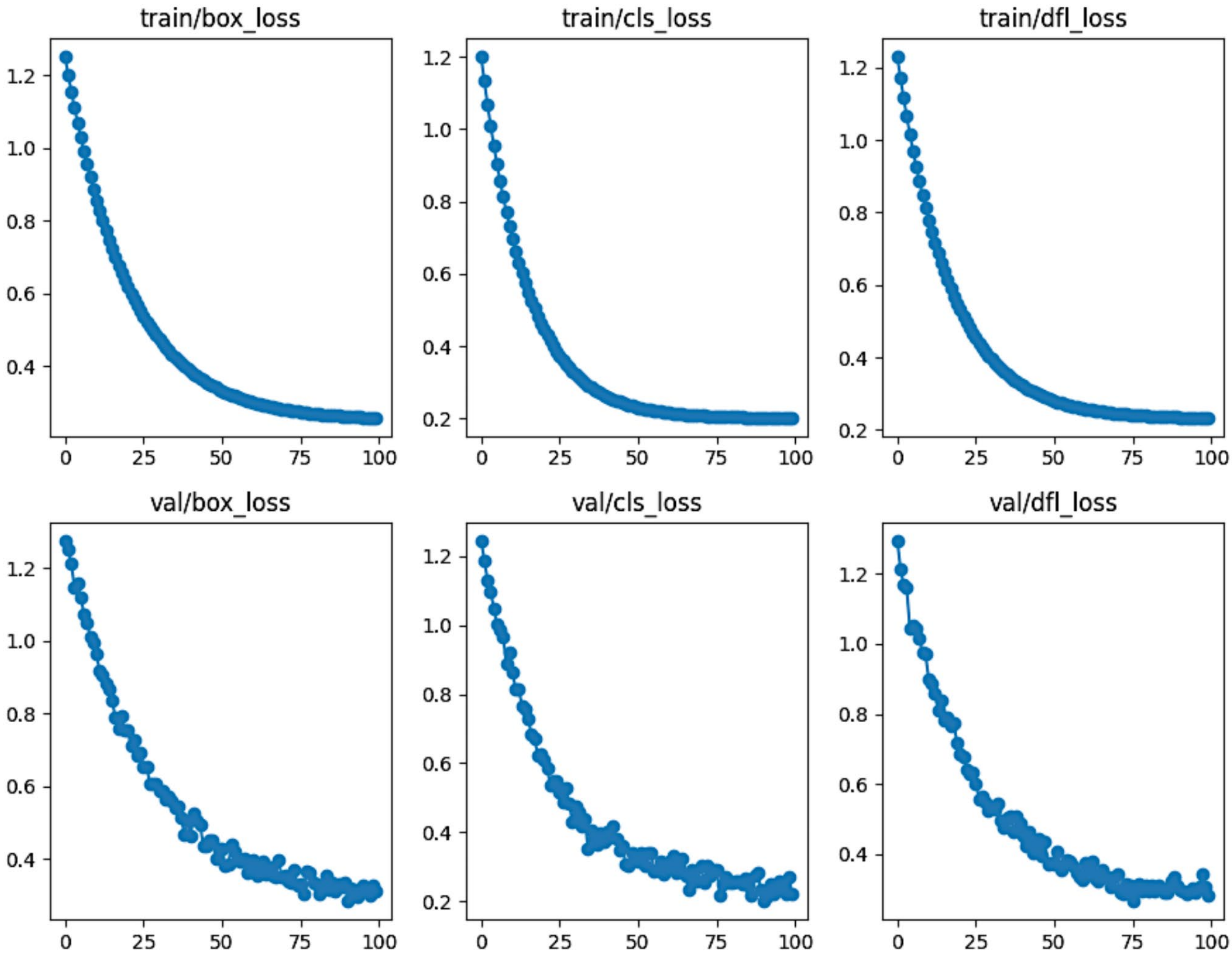


Fig. 13. The training and validation loss metrics were obtained over configuration #1.

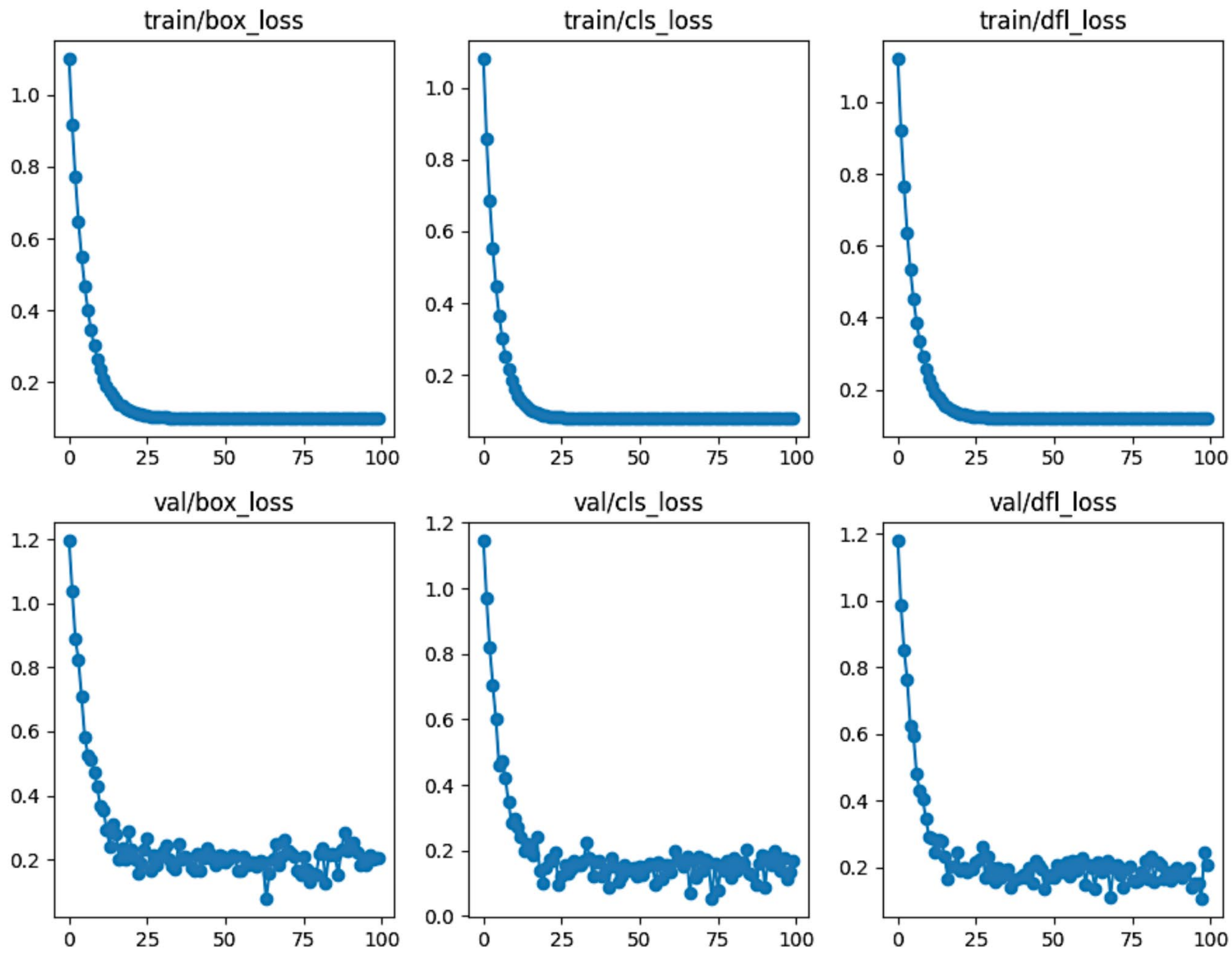


Fig. 14. The training and validation loss metrics were obtained over configuration #2.

It can be observed from the experimental results shown in the above table, that the YOLO-based object detection model has performed well on both the raw and augmented results. Despite feeding the pre-processed images, the model managed to yield reasonable accuracy. On the other hand, considerable efforts are needed to pre-process the images concerning both the contrast and the sharpness of the objects in the image.

Furthermore, the impact of data augmentation is evaluated across three different configurations, Configuration#5 where only the CLAHE is performed on the data samples, Configuration #6 is performed CLAHE along with affine transforms over 30% of the dataset across all the partitions. Configure #7 is performed CLAHE along with affine transforms over 35% of the dataset across all the partitions. The obtained results across all three configurations are presented in Table 6.

The experimental results on evaluating over the standard parameters across the raw and augmented data are presented in Figs. 19 and 20. The bounded box across the region of interest shows the associated class label. The results obtained from the experimenting over the augmented data are presented in Fig. 19.

This can be observed from the experimental results shown in the above figure. The region of interest is enclosed in the box, and the label indicates the identified object class. The value that is enclosed in the label part is the confidence score that is associated with the class that is being identified. The confidence score is significant in assessing the likelihood of the region of interest being enclosed in the bounded box and the precision of that box's location in the image. The experimental outcome of the raw images is shown in Fig. 20.

It can be observed from the experimental outcome that the confidence values are comparatively lower than the raw images due to the deviation of the bounded box localization over the region of interest[44]. When performing data augmentation over the raw images, contrast enhancement and image sharpening will assist in the precise localization of the region of interest. The statistical analysis of bone fracture recognition with the other existing models is presented in Table 7 to summarize the performance of the YOLO V10 model.

It can be observed from the tabulated values that the YOLO V10 model outperforms the other state-of-the-art model in precisely identifying bone fractures and the associated classes. The results of the current study are across six different classes, and upon considering the classification for two classes, the YOLO V10 model would perform much better[49].

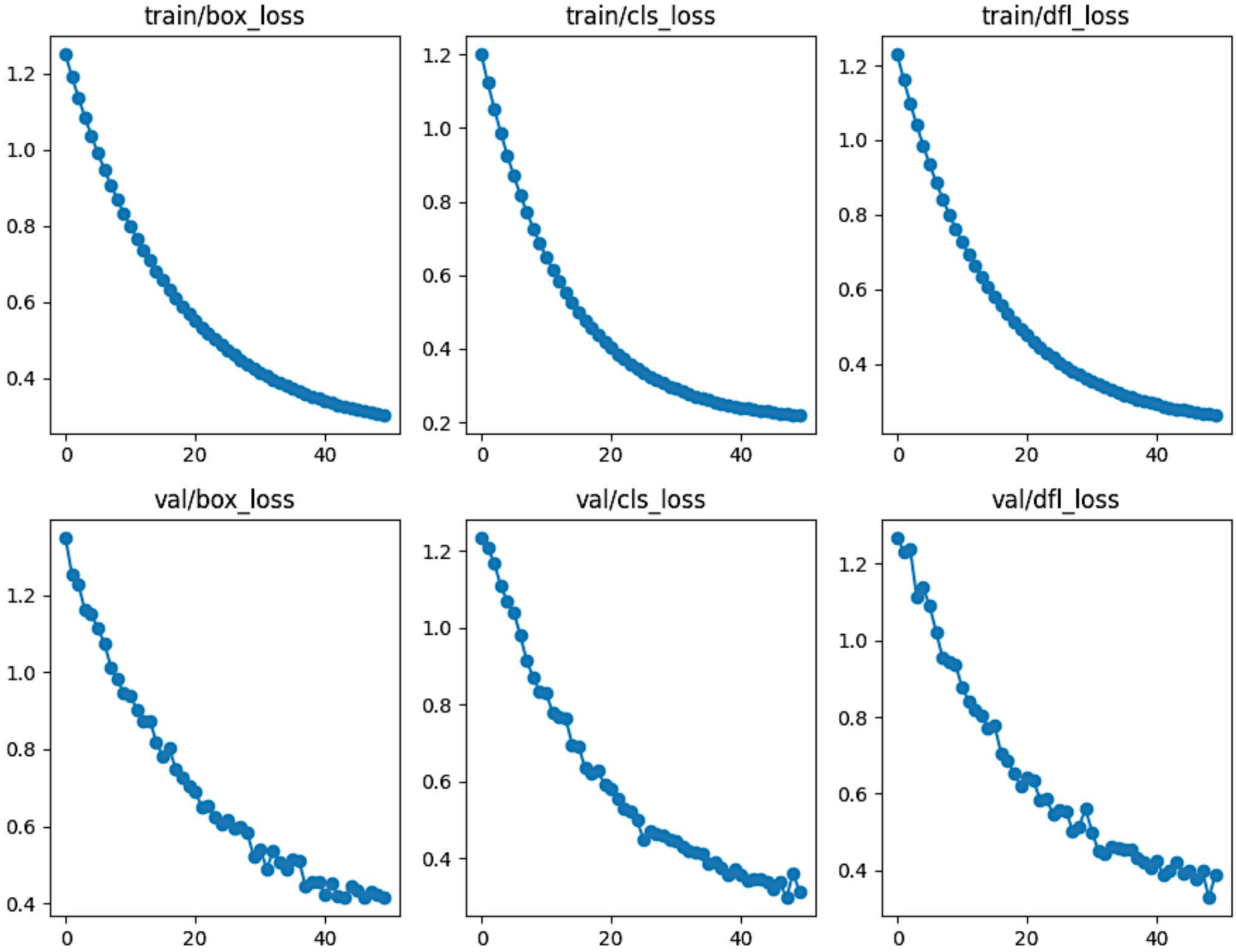


Fig. 15. The training and validation loss metrics were obtained over configuration #3.

Threats to the validity of the statement

The types of fractures that are considered in the current study are limited to the availability of the data that is being considered in the current study. The images in the dataset are annotated to the corresponding class, and the YOLO V10 model is trained on the ground facts associated with the dataset. The experimentation is done only on the data samples that are available as part of the dataset, hence the model is not evaluated with divergent instances. The model performance is evaluated in concerning to the ground truths, and they are not clinically evaluated.

Potential limitations

The current study has evaluated the YOLO V10 model concerning the learning rate and the number of epochs. The other hyperparameters, like the batch size, activation function, and optimizer, can be evaluated for better statistical analysis of the model[50,51]. The other existing models are not considered. However, they could be considered when analyzing the trade-off among the performances. The data balancing is not performed in the current study; however, it would significantly impact the model's performance. The various versions of YOLO V10 can be analyzed to determine the trade-off among the performances. The ablation study is not performed in the current study as the YOLO V10 model is directly used, and no additional components are being used to optimize the performances, which is considered one of the potential limitations of the current study. The impact of various hyperparameters are being analyzed, but hyperparameter tuning can also be performed for better optimistic outcomes. The current study have not taken the data leakage by data augmentation into consideration.

Conclusion

The current study identifies bone fractures from X-ray images using the YOLO V10 model. The YOLO V10 model is being analyzed concerning various hyperparameters by changing the number of epochs and the learning rate. However, with minimal change, the model can perform well in almost all the configurations. The model exhibited better accuracy when the number of epochs increased, where the model performed well for training data and struggled to analyze the validation data. Furthermore, data augmentation, like sharpening and contrast enhancements, is performed over the images, where the model has shown slightly better performance

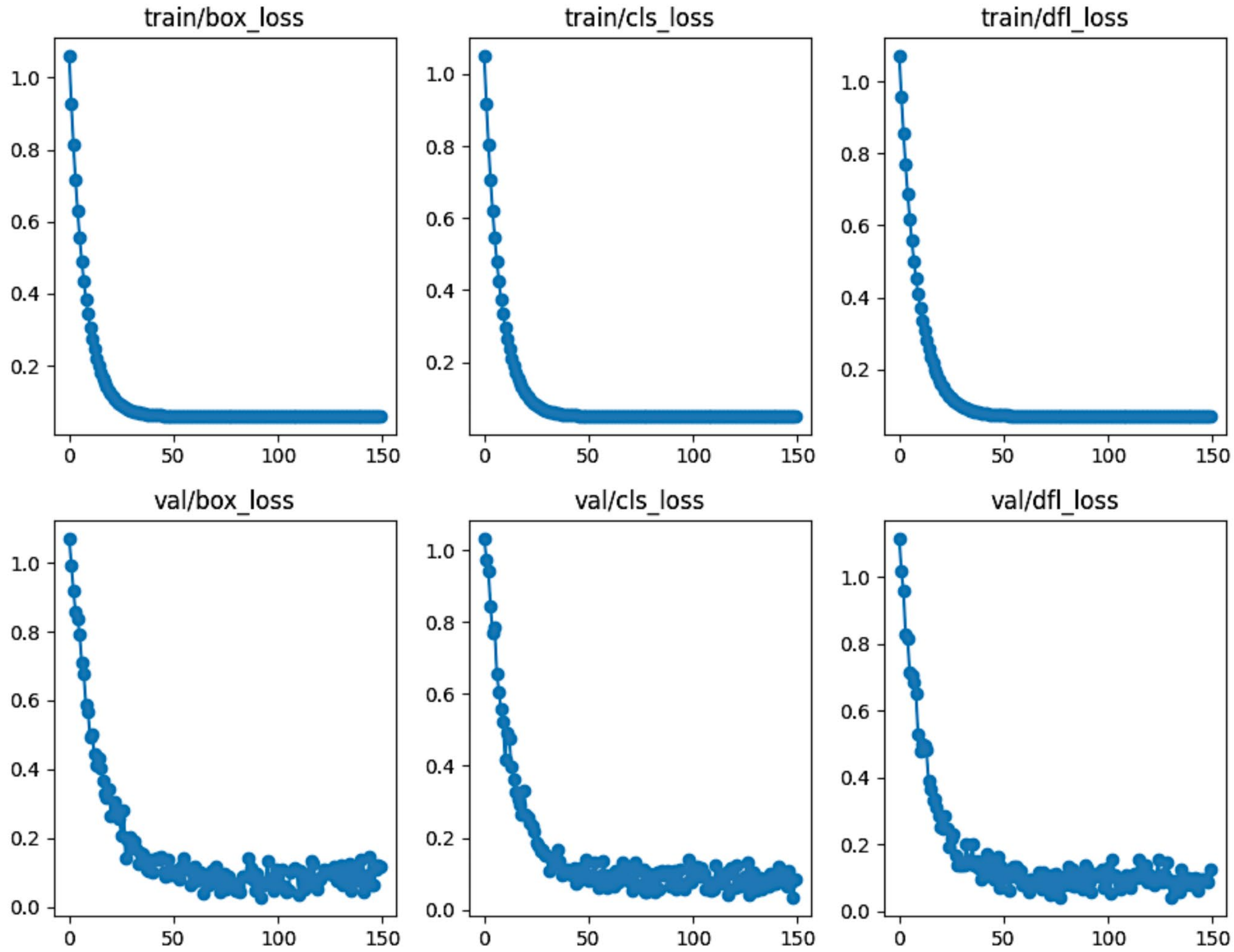


Fig. 16. The training and validation loss metrics were obtained over configuration #4.

than the raw data. It can be observed from the experimental results that the YOLO V10 model is more robust than the augmented data in comparison to the raw data. The current dataset consists of seven different classes, and the model has achieved an accuracy of 96.4% when working with multi-class identification. The model can be expected to perform much better than the binary class. The confidence score was observed to be fair enough in localizing the object in the image. Using a transformer to capture the contextual information would assist in better performance and precise localization of the abnormality from the images.

The current study can be further extended over various versions of YOLO V10, and analyze the trade-off between their accuracies and computational complexities. The optimizer and regularizer are not being analyzed in the current study. They can be evaluated as they are exceptionally significant in model training and generalization. A deep survey of all the YOLO versions from YOLO V1 to V10 can also be performed to analyze the evolution of the yolo model in object identification and assess its impact on the healthcare domain. In the current study, the impact of hyperparameters is analyzed upto a limited extent, furthermore, more hyperparameters can be considered, and optimization of hyperparameters can be done using the platforms like Optuna. The impact of data leakage could also be taken in to consideration for better statistical analysis of the model.

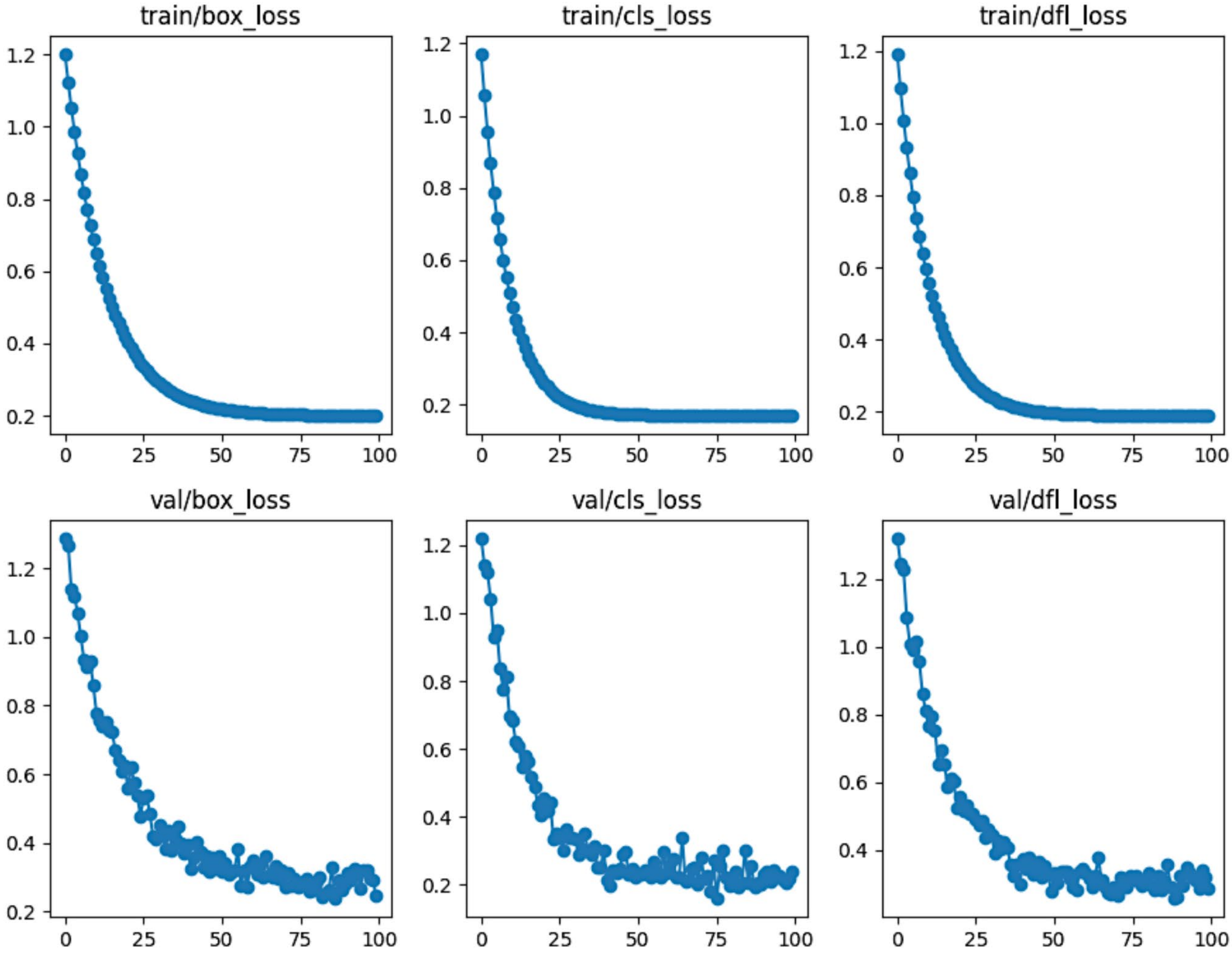
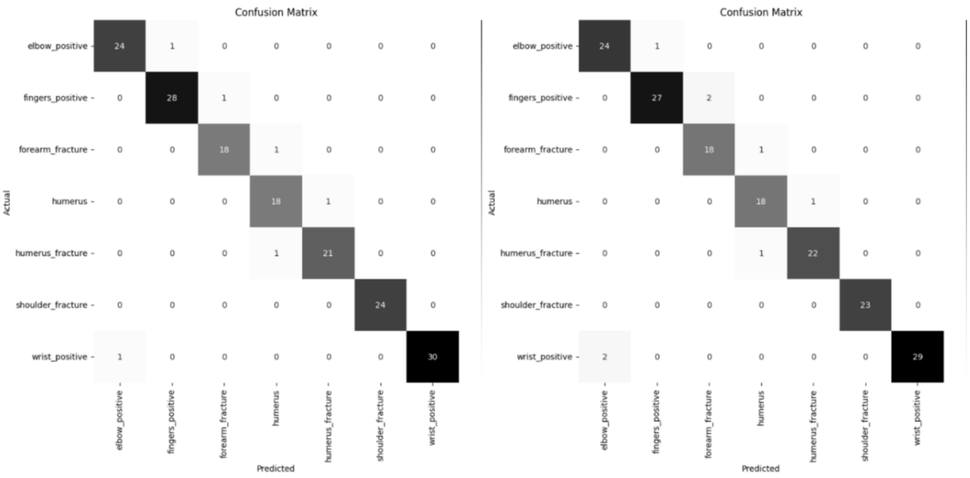


Fig. 17. The training and validation loss metrics were obtained on standard hyperparameters without data augmentation.

Model	Precision	Recall	F1-Score	mAP@0.50
YOLO V10 (Standard)	0.98	0.97	0.97	0.96
YOLO V10 (Conf #1)	0.96	0.98	0.96	0.91
YOLO V10 (Conf #2)	0.92	0.93	0.90	0.86
YOLO V10 (Conf #3)	0.90	0.91	0.90	0.84
YOLO V10 (Conf #4)	0.97	0.98	0.98	0.93

Table 4. Details of the observed experimental outcomes of various configurations.



(a) (b)

Fig. 18. The confusion matrix on experimenting over the YOLO V10, (**a**) Augmented images, (**b**) raw X-Ray images.

Model	Precision	Recall	F1-Score	mAP@0.50
YOLO V10 (Augmented data)	0.98	0.97	0.97	0.961
YOLO V10 (Raw data)	0.95	0.94	0.95	0.948

Table 5. Experimental outcome over the Raw and augmented images.

Model	Precision	Recall	F1-Score	mAP@0.50
YOLO V10 (Conf #4)	0.96	0.95	0.96	0.94
YOLO V10 (Conf #5)	0.96	0.95	0.95	0.95
YOLO V10 (Conf #6)	0.97	0.96	0.97	0.95
YOLO V10 (Standard)	0.98	0.97	0.97	0.96

Table 6. Experimental outcomes across various data augmentation settings.

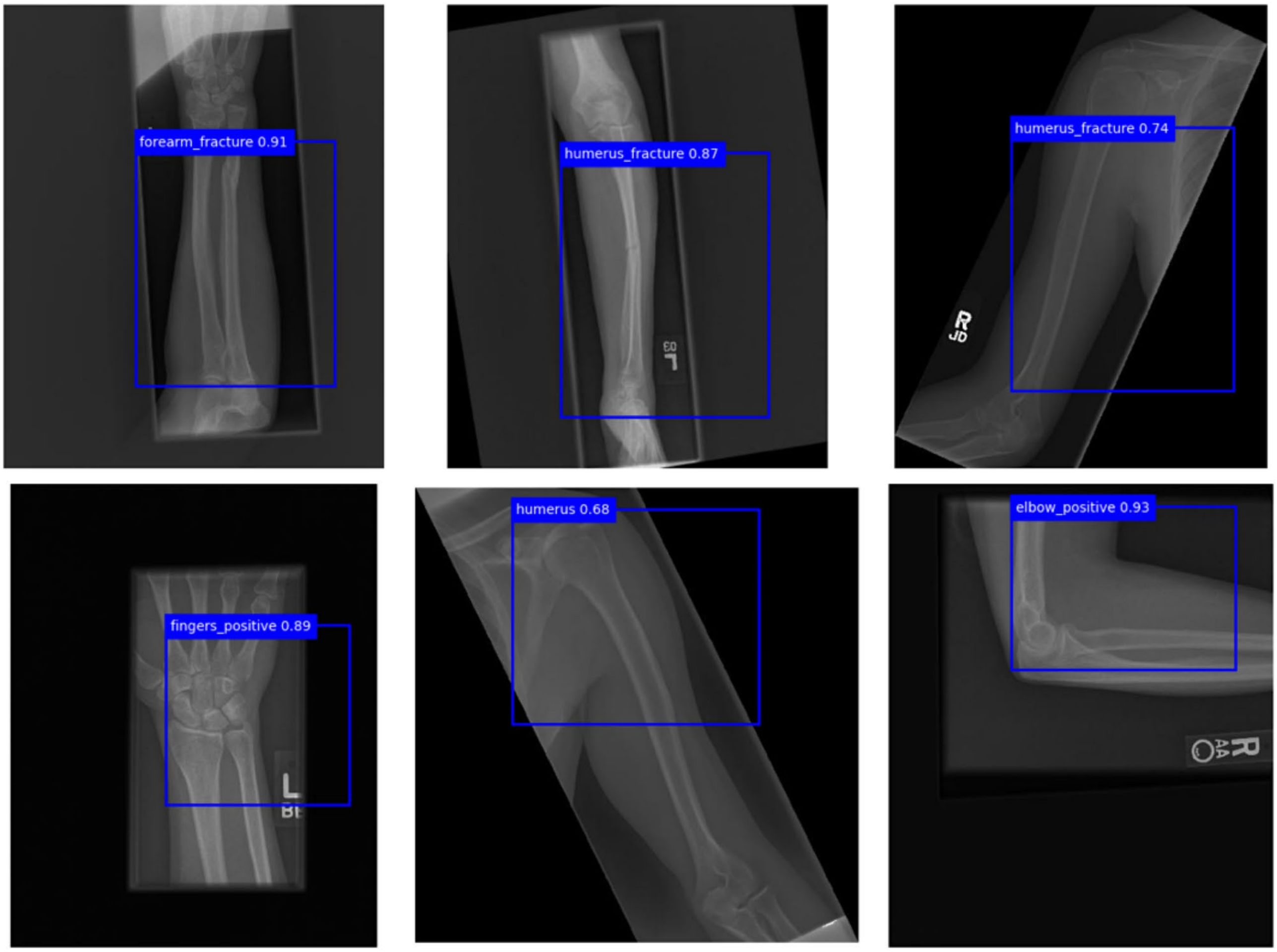


Fig. 19. The experimental outcome of the YOLO V10 model on augmented X-ray images.

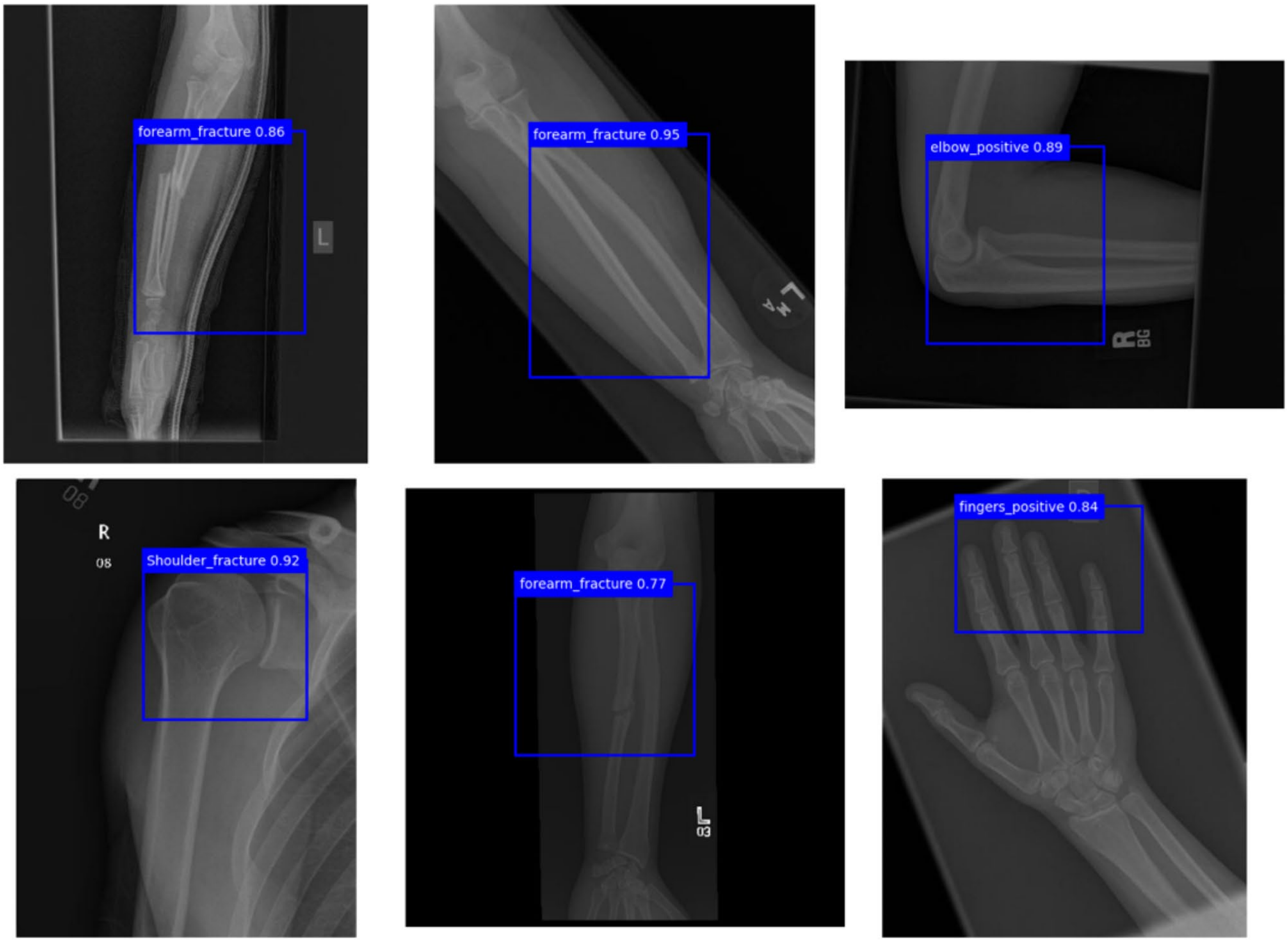


Fig. 20. The experimental outcome of the YOLO V10 model on raw X-ray images.

Model	Accuracy	Precision	Recall	F1-Score
VGG-16[45]	0.78	0.77	0.77	0.78
Inception V3[46]	0.76	0.74	0.71	0.72
ResNet50[46]	0.77	0.68	0.66	0.67
VGG19[46]	0.84	0.74	0.68	0.72
VGG-16[47]	0.78	0.80	0.78	0.77
ResNet-50[47]	0.78	0.80	0.79	0.78
DenseNet-169[47]	0.80	0.81	0.80	0.79
InceptionV3[47]	0.86	0.87	0.86	0.85
Xception[47]	0.86	0.86	0.86	0.86
VGG19[48]	0.95	0.93	0.93	0.93
Densely CNN[49]	0.83	n/a	0.96	n/a
YOLO V8[46]	0.91	0.89	0.88	0.90
YOLO NAS[46]	0.95	0.94	0.92	0.93
YOLO V10 (Current Study)	0.96	0.98	0.97	0.97

Table 7. Performance analysis with existing studies.

Data availability

The dataset analysed during the current study is available in the Bone Fracture Detection: Computer Vision Project repository, Link: https://www.kaggle.com/datasets/pkdarabi/bone-fracture-detection-computer-vision-project.

Received: 6 January 2025; Accepted: 7 March 2025

Published online: 21 March 2025

References

1. Dlshad Ahmed & Kosrat and Roojwan Hawezi. Detection of Bone Fracture Based on Machine Learning Techniques. Measurement: Sensors, vol. 27, p. 100723. https://doi.org/10.1016/j.measen.2023.100723 (2023).
2. Joshi, D. & Singh, T. P. A survey of fracture detection techniques in bone X-ray images. *Artif. Intell. Rev.* **53**, 4475–4517. https://doi.org/10.1007/s10462-019-09799-0 (2020).
3. Srinivasu, N. et al. Enhancing medical image classification via federated learning and Pre-trained model. *Egypt. Inf. J.* **27**, 100530. https://doi.org/10.1016/j.eij.2024.100530 (2024).
4. Ju, R. Y. & Cai, W. Fracture detection in pediatric wrist trauma X-ray images using YOLOv8 algorithm. *Sci. Rep.* **13**, 20077. https://doi.org/10.1038/s41598-023-47460-7 (2023).
5. Abedeen, I. et al. FracAtlas: A dataset for fracture classification, localization and segmentation of musculoskeletal radiographs. *Sci. Data* **10**, 521. https://doi.org/10.1038/s41597-023-02432-4 (2023).
6. Srinivasu, P. N., Sravya, V. S., Tagore, V. D., Vinoothna, V. & Mokshada, Y. CBCD: Comprehensive Analysis for Bone Cancer Diagnosis through MRI Imaging. 2024 1st International Conference on Cognitive, Green and Ubiquitous Computing (IC-CGU), Bhubaneswar, India, 2024. 1–6. https://doi.org/10.1109/IC-CGU58078.2024.10530830.
7. Su, Z., Adam, A., Nasrudin, M. F., Ayob, M. & Punganan, G. Skeletal fracture detection with deep learning: A comprehensive review. *Diagnostics* **13**, 3245. https://doi.org/10.3390/diagnostics13203245 (2023).
8. Eswara Rao, G. V., Naga Srinivasu, R. B. P., Ijaz, M. F. & Woźniak, M. Hybrid framework for respiratory lung diseases detection based on classical CNN and quantum classifiers from chest X-rays. *Biomed. Signal Process. Control.* **88**, 105567. https://doi.org/10.1016/j.bspc.2023.105567 (2024).
9. Ao Wang, H. et al. Guiguang Ding, YOLOv10: Real-Time End-to-End Object Detection. ArXiv, /abs/2405.14458. (2024).
10. Oh, J., Hwang, S. & Lee, J. Enhancing X-ray-Based wrist fracture diagnosis using HyperColumn-Convolutional block attention module. *Diagnostics* **13**, 2927. https://doi.org/10.3390/diagnostics13182927 (2023).
11. Akhmedov, F., Nasimov, R. & Abdusalomov, A. Dehazing algorithm integration with YOLO-v10 for ship fire detection. *Fire* **7**, 332. https://doi.org/10.3390/fire7090332 (2024).
12. Cohen, M. et al. Artificial intelligence vs. radiologist: accuracy of wrist fracture detection on radiographs. *Eur. Radiol.* **33**, 3974–3983. https://doi.org/10.1007/s00330-022-09349-3 (2023).
13. Kutbi, M. Artificial Intelligence-Based applications for bone fracture detection using medical images: A systematic review. *Diagnostics* **14**, 1879. https://doi.org/10.3390/diagnostics14171879 (2024).
14. Srinivasu, P. N. et al. An interpretable approach with explainable AI for heart stroke prediction. *Diagnostics* **14**, 128. https://doi.org/10.3390/diagnostics14020128 (2024).
15. Dlshad Ahmed & Kosrat and Roojwan Hawezi. Detection of Bone Fracture Based on Machine Learning Techniques. Measurement Sensors 27, 100723. https://doi.org/10.1016/j.measen.2023.100723 (2023).
16. Karimunnisa, S., Savarapu, P. R., Madupu, R. K., Basha, C. Z. & Neelakanteswara, P. Detection of Bone Fractures Automatically with Enhanced Performance with Better Combination of Filtering and Neural Networks, 2020 Second International Conference on Inventive Research in Computing Applications (ICIRCA), Coimbatore, India, 2020, pp. 189–193. https://doi.org/10.1109/ICIRCA48905.2020.9183085.
17. Yang, A. Y., Cheng, L., Shimaponda-Nawa, M. & Zhu, H. Y. Long-Bone Fracture Detection using Artificial Neural Networks based on Line Features of X-ray Images, 2019 IEEE Symposium Series on Computational Intelligence (SSCI), Xiamen, China, pp. 2595–2602. https://doi.org/10.1109/SSCI44817.2019.9002664 (2019).
18. Ma, Y. and Yixin Luo. Bone Fracture Detection through the Two-stage System of Crack-Sensitive Convolutional Neural Network. Informatics in Medicine Unlocked, vol. 22, p. 100452. https://doi.org/10.1016/j.imu.2020.100452 (2020).
19. Waseem Abbas, S. M., Adnan, M. A., Javid, F., Majeed, T. & Ahsan Haroon zeb, Syed Saqlian Hassan Lower Leg Bone Fracture Detection and Classification Using Faster RCNN for X-Rays Images, 2020 IEEE 23rd International Multitopic Conference (INMIC), Bahawalpur, Pakistan, pp. 1–6. https://doi.org/10.1109/INMIC50486.2020.9318052 (2020).
20. Yadav, D. P. & Rathor, S. Bone Fracture Detection and Classification using Deep Learning Approach, International Conference on Power Electronics & IoT Applications in Renewable Energy and its Control (PARC), Mathura, India, 2020, pp. 282–285. https://doi.org/10.1109/PARC49193.2020.236611 (2020).
21. El-Saadawy, H., Tantawi, M., Shedeed, H. A. & Tolba, M. F. A Hybrid Two-Stage GNG–Modified VGG Method for Bone X-Rays Classification and Abnormality Detection, in IEEE Access, vol. 9, pp. 76649–76661. https://doi.org/10.1109/ACCESS.2021.3081915 (2021).
22. Mittal, K., Gill, K. S., Chauhan, R. & Kapruwan, A. Innovative Fracture Diagnosis: MobileNet CNN Approach for Precise Bone Fracture Detection and Classification, 2024 International Conference on Intelligent Systems for Cybersecurity (ISCS), Gurugram, India, 2024, 1–5. https://doi.org/10.1109/ISCS61804.2024.10581396.
23. Zou, J., Mohd, R. & Arshad Detection of Whole Body Bone Fractures Based on Improved YOLOv7. Biomedical Signal Processing and Control, vol. 91, p. 105995. https://doi.org/10.1016/j.bspc.2024.105995 (2024).
24. Medaramatla, S. C., Samhitha, C. V., Pande, S. D. & Vinta, S. R. Detection of Hand Bone Fractures in X-Ray Images Using Hybrid YOLO NAS. IEEE Access 12, 57661–57673. https://doi.org/10.1109/ACCESS.2024.3379760 (2024).
25. Tanzi, L., Vezzetti, E., Moreno, R. & Moos, S. X-Ray bone fracture classification using deep learning: A baseline for designing a reliable approach. *Appl. Sci.* **10**, 1507. https://doi.org/10.3390/app10041507 (2020).
26. Parvin, S. & Rahman, A. A real-time human bone fracture detection and classification from multi-modal images using deep learning technique. *Appl. Intell.* **54**, 9269–9285. https://doi.org/10.1007/s10489-024-05588-7 (2024).
27. Hardalaç, F. et al. Fracture detection in wrist X-ray images using deep Learning-Based object detection models. *Sensors* **22**, 1285. https://doi.org/10.3390/s22031285 (2022).
28. Tahi, A. et al. Enhancing Diagnosis: Ensemble Deep-learning Model for Fracture Detection Using X-ray Images. Clinical Radiology, vol. 79, no. 11, pp. e1394-e1402. https://doi.org/10.1016/j.crad.2024.08.006 (2024).
29. Natarajan, V. V. S. K., N. P, A. M. & Moorthi Hosahalli, N. M. C. A and Efficient CNN-Based Bone Fracture Detection in X-Ray Radiographs with MobileNetV2, 2024 2nd International Conference on Recent Advances in Information Technology for Sustainable Development (ICRAIS), Manipal, India, pp. 72–77. https://doi.org/10.1109/ICRAIS62903.2024.10811726 (2024).
30. Yıldız Potter, İ. et al. Proximal femur fracture detection on plain radiography via feature pyramid networks. *Sci. Rep.* **14**, 12046. https://doi.org/10.1038/s41598-024-63001-2 (2024).
31. Alam, A. et al. Novel transfer learning-based bone fracture detection using radiographic images. *BMC Med. Imaging.* **25**, 5. https://doi.org/10.1186/s12880-024-01546-4 (2025).
32. Ma, Y. and Yixin Luo. Bone fracture detection through the Two-stage system of Crack-Sensitive convolutional neural network. Informatics in medicine unlocked, **22**, p. 100452. https://doi.org/10.1016/j.imu.2020.100452 (2020).
33. Roboflow Universe, bone fracture detection Dataset. https://universe.roboflow.com/veda/bone-fracture-detection-daoon (2023). [Last Accessed on 18 Oct 2024].
34. Song, Y. et al. Unsharp masking image enhancement the parallel algorithm based on cross-platform. *Sci. Rep.* **12**, 20175. https://doi.org/10.1038/s41598-022-21745-9 (2022).

35. Li, L. et al. X-ray image enhancement based on adaptive gradient domain guided image filtering. *Appl. Sci.* **12**, 10453. https://doi.org/10.3390/app122010453 (2022).
36. Naga Srinivasu, P., Srinivasa Rao, T., Srinivas, G. & Prasad Reddy, P. V. G. D. A Computationally Efficient Skull Scraping Approach for Brain MR Image, Recent Advances in Computer Science and Communications; Volume 13, Issue 5, Year. https://doi.org/10.2174/2213275912666190809111928 (2020).
37. Stephen, M. et al. Nov., Trey Greer, Bart ter Haar Romeny, John B. Zimmerman, Karel Zuiderveld, Adaptive Histogram Equalization and Its Variations. Computer Vision, Graphics, and Image Processing, vol. 39, no. 3, 1987, pp. 355–368. https://doi.org/10.1016/S0734-189X(87)80186-X (2024).
38. Wang, C. Y., Bochkovskiy, A., Liao, H. Y. M. & Recognition, P. Scaled-YOLOv4: Scaling Cross Stage Partial Network, 2021 IEEE/CVF Conference on Computer Vision and (CVPR), Nashville, TN, USA, pp. 13024–13033. https://doi.org/10.1109/CVPR46437.2021.01283 (2021).
39. Chen, Y., Liu, H. & Fang, F. A novel pansharpening method based on cross stage partial network and transformer. *Sci. Rep.* **14**, 12631. https://doi.org/10.1038/s41598-024-63336-w (2024).
40. Liu, S., Qi, L., Qin, H., Shi, J. & Jia, J. Path Aggregation Network for Instance Segmentation, 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, pp. 8759–8768. https://doi.org/10.1109/CVPR.2018.00913 (2018).
41. Li, D., Ahmed, F., Wu, N., Sethi, A. I. & YOLO-JD: A deep learning network for jute diseases and pests detection from images. *Plants* **11**, 937. https://doi.org/10.3390/plants11070937 (2022).
42. Rainio, O., Teuho, J. & Klén, R. Evaluation metrics and statistical tests for machine learning. *Sci. Rep.* **14**, 6086. https://doi.org/10.1038/s41598-024-56706-x (2024).
43. Markoulidakis, I. et al. Multiclass confusion matrix reduction method and its application on net promoter score classification problem. *Technologies* **9**, 81. https://doi.org/10.3390/technologies9040081 (2021).
44. Cai, D., Zhang, Z. & Zhang, Z. Corner-Point and Foreground-Area IoU loss: better localization of small objects in bounding box regression. *Sensors* **23**, 4961. https://doi.org/10.3390/s23104961 (2023).
45. Tahir, A. et al. Enhancing Diagnosis: Ensemble Deep-learning Model for Fracture Detection Using X-ray Images. Clinical Radiology, vol. 79, no. 11, pp. e1394-e1402. https://doi.org/10.1016/j.crad.2024.08.006 (2024).
46. Justin, D. et al. Eugene Ozhinsky, Sharmila Majumdar, and Valentina pedoia, automatic hip fracture identification and functional subclassification with deep learning. *Radiology: Artif. Intell.* **2**, 2 (2020).
47. Reddy, K. N. K. & Cutsuridis, V. 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, pp. 1–5. https://doi.org/10.1109/ICASSPW59220.2023.10193015 (2023).
48. Mohammad, F., Al-Ahmadi, S. & Al-Muhtadi, J. Block-Deep: A Hybrid Secure Data Storage and Diagnosis Model for Bone Fracture Identification of Athlete From X-Ray and MRI Images. *IEEE Access*, vol. 11, pp. 142360–142370. https://doi.org/10.1109/ACCESS.2023.3330914 (2023).
49. Guarnido-Lopez, P., Ramirez-Agudelo, J. F., Denimal, E. & Benaouda, M. Programming and setting up the object detection algorithm YOLO to determine feeding activities of beef cattle: A comparison between YOLOv8m and YOLOv10m. *Anim. (Basel)* **14** (19), 2821. https://doi.org/10.3390/ani14192821 (2024).
50. Ajayi, O. G., Ibrahim, P. O. & Adegboyega, O. S. Effect of hyperparameter tuning on the performance of YOLOv8 for multi crop classification on UAV images. *Appl. Sci.* **14**, 5708. https://doi.org/10.3390/app14135708 (2024).
51. Zhou, G., Liu, M., Wang, H., Zheng, Y. Power equipment image enhancement processing based on YOLO-v8 target detection model under MSRCR algorithm. *Int. J. Low-Carbon Technol.* **19**, 1717–1724. https://doi.org/10.1093/ijlct/ctae122 (2024).

Author contributions

The contributions of each author in the current study are as follows. Conceptualization and problem formulation, P.N.S., S.A. and S.C.N.; Data acquisition, S.A. and A.A.; Methodology, G.L.A., A.A., and S.C.N.; Project administration, S.A.; and A.A.; Investigation, A.A., S.C.N. and P.N.S.; Funding acquisition, P.N.S., A.A.; Resources, S.A., G.L.A. and S.C.N.; Software, P.N.S., and S.A.; Validation of results, S.C.N., S.A., and G.L.A.; Preparation of the draft, P.N.S., and S.A.; All authors have reviewed and approved the work's published version.

Funding

This work was supported by the Deanship of Scientific Research, Vice Presidency for Graduate Studies and Scientific Research, King Faisal University, Saudi Arabia [Grant No. KFU250948].

Declarations

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to P.N.S.