IIETA International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Ingénierie des Systèmes d'Information
Vol. 30, No. 7, July, 2025, pp. 1775-1782
Journal homepage: http://iieta.org/journals/isi

Bone Fracture Detection Using Hybrid EfficientNet-B0 and ResNet50 with SVM: A Comparative Performance Analysis

Mahmood Hameed Qahtan[1*], Areej Mahmoud Asaad[2], Ahmed Kh. Younis[2]

[1] Department of Artificial Intelligence Techniques Engineering, Technical Engineering College for Computer and Artificial Intelligence, Mosul, Northern Technical University, Mosul 41001, Iraq
[2] Department of Computer Techniques Engineering, Technical Engineering College for Computer and Artificial Intelligence, Mosul, Northern Technical University, Mosul 41001, Iraq

Corresponding Author Email: mahmood.hameed@ntu.edu.iq

https://doi.org/10.18280/isi.300710

Received: 1 June 2025
Revised: 5 July 2025
Accepted: 20 July 2025
Available online: 31 July 2025

Keywords:
bone fracture detection, deep neural networks, comparative analysis, ResNet50, EfficientNet-B0, SVM, medical classification, medical imaging

ABSTRACT

The accurate identification of osseous fractures is crucial for precise medical diagnoses and treatment planning. This study introduces a new hybrid classification approach, integrating EfficientNet-B0 and ResNet50 deep learning models with an SVM classifier, surpassing traditional versions. Leveraging pre-trained feature extractors for EfficientNet-B0 and ResNet50, the proposed method achieves a test accuracy of 98.01% and a recall of 0.99 for fractured cases with EfficientNet-B0+SVM, while reducing runtime to 20.44 minutes. ResNet50 + SVM also improved accuracy from 80.05% to 96.41% with a runtime of 38.47 minutes, compared to 83.96 minutes standalone. This hybrid approach demonstrates significant enhancements in accuracy and efficiency, positioning it as a promising tool for clinical bone fracture detection.

1. INTRODUCTION

Fractures are either entire or partial breaks in the bone. The primary cause of fracture is the significant influence or force applied to a bone that it is structurally capable of supporting [1]. For proper treatment planning and patient care, fractures a frequent orthopedic condition need to be diagnosed quickly and accurately. Traditional fracture diagnosis relies heavily on radiologists' ability to visually analyze X-ray images in order to detect and classify fractures. However, when working with intricate fracture patterns or minute irregularities, this approach can be laborious, subjective, and prone to human error [2-4]. In order to help radiologists identify bone fractures, artificial intelligence (AI) and deep learning (DL) are currently gaining a lot of interest [5-7]. Given that AI enables the discovery of significant patterns in massive data sets, it offers chances for automated classification, detection, and localization, which may result in quicker and more accurate diagnosis. In order to manage the case, determine the extent of the fracture, and enhance patient outcomes, the treating physician may find this useful in making prompt and effective judgments. Additionally, it can assist in removing diagnostic variability across several observers [8, 9]. Medical imaging has undergone a revolution because to the development of artificial intelligence (AI), especially deep learning, which provides automated solutions for problems like fracture detection. Convolutional Neural Networks (CNNs), such as ResNet50 and EfficientNetB0, have demonstrated remarkable success in image classification tasks by learning hierarchical features directly from raw data [10-12]. Notwithstanding these developments, the combination of deep learning architectures and machine learning frameworks may improve diagnostic precision even further, increasing the dependability and interpretability of automated systems for bone fracture identification. In general, hybrid deep learning algorithms minimize computing overhead by addressing class imbalance with methods like Synthetic Minority Oversampling Technique (SMOTE), preventing overfitting with simpler classifiers, and extracting features using pre-trained models [13].

This study evaluates four bone fracture classification approaches: traditional training of ResNet50 and EfficientNetB0 and hybrid techniques using their features with SVM classification. We incorporate SMOTE for data balancing, PCA for dimensionality reduction, and threshold optimization using Precision-Recall curves to improve classification performance. These techniques address key challenges such as class imbalance and feature redundancy, resulting in significant improvements over direct training.

There are five sections in this study. The research is introduced in Section 1. Section 2 examines relevant research on current methods for fracture identification and medical imaging. The bone fracture detecting system's methodology is described in Section 3. The findings are presented in Section 4, and Section 5 provides a summary of the paper's conclusions.

2. RELATED WORK

Numerous related studies that have investigated deep learning and hybrid techniques for fracture classification attest to the widespread use of deep learning in medical imaging applications, including the identification of bone fractures.

In 2019, Castro-Gutiérrez et al. [14] suggested a technique for identifying acetabular fractures in X-ray pictures, with an emphasis on the noisy pictures that are frequently seen in medical settings. SVM is used for classification, Local Binary Pattern (LBP) is used for feature extraction, and Contrast-Limited Adaptive Histogram Equalization (CLAHE) is used for pre-processing to improve image contrast. A trauma specialist verified the method's 80% accuracy against a gold standard using a small dataset of 15 anteroposterior X-ray pictures (10 for training and 5 for validation). While the small sample size limits generalizability, the use of CLAHE significantly improved feature extraction in low-resolution images.

In 2019, Hržić et al. [15] introduced a novel local-entropy-based approach for segmenting and detecting fractures in X-ray images of pediatric radius and ulna bones. The method employs a multi-stage pipeline, starting with image alignment using Principal Component Analysis (PCA), followed by the calculation of local Shannon entropy within a sliding 2D window to denoise and remove tissue, thereby enhancing the visibility of bone contours. Bone contour extraction is refined using a graph theory-based method, and fracture detection is accomplished by using polynomial regression to compare extracted contours to an ideal healthy bone model. On a dataset of 860 X-ray images, the accuracy and precision were 91.16% and 86.22%, respectively (218 with fractures, 642 without). The approach excels at identifying minor, hard-to-detect fractures, outperforming traditional methods, such as the Canny filter (61.86% accuracy). False negatives result from its inability to detect some fracture types that do not substantially change bone shapes.

In 2020, Abbas et al. [16] proposed a transfer learning-based Faster R-CNN model for detecting and classifying lower leg bone fractures in X-ray images. The model uses a Region Proposal Network (RPN) to create bounding boxes around fracture locations using the VGG-16 architecture as a foundation network. These boxes are then classified into fracture or non-fracture categories. The top layer was retrained using the Inception V2 network on a dataset of 50 X-ray images (30 for training, 20 for validation) from Bahawal Victoria Hospital, achieving an overall accuracy of 94% and a mean average precision (mAP) of 60% for fracture localization. The model showed effectiveness in locating fractures using bounding boxes after 40,000 steps of training till the loss hit 0.0005. However, generalizability is limited by the small sample size, and the map indicates that detection precision could be increased.

In 2020, Yadav and Rathor [17] presented a study that suggested using X-ray pictures to detect and classify bone fractures using a deep learning technique. To extract information and categorize bones as either healthy or fractured, the technique uses a Convolutional Neural Network (CNN) with four convolutional layers, max pooling, and dense layers. To address overfitting resulting from the initial dataset's low size of 100 photographs, data augmentation techniques were employed to expand the dataset to 4000 photos. The model's classification accuracy, using 5-fold cross-validation, was 92.44%; it was over 95% and 93% accurate on 10% and 20% test splits.

In 2023, Ahmed and Hawezi [18] developed a method that focuses on the lower leg bones and uses machine learning to detect bone fractures in X-ray images. They employed pre-processing techniques including Gaussian filtering and adaptive histogram equalization to enhance image quality. The next step was to use the Gray-Level Co-occurrence Matrix (GLCM) for feature extraction and canny edge detection. When the extracted features—such as contrast, correlation, and homogeneity—were fed into numerous classifiers, such as Naive Bayes, Decision Tree, Nearest Neighbors, and Random Forest, SVM achieved the highest accuracy of 92% on a dataset of 270 X-ray images. Although the study's applicability is limited by its focus on lower leg bones and short dataset size, it emphasizes the benefits of integrating machine learning with reliable pre-processing for automated fracture detection.

In 2024, Thota et al. [19] compared Convolutional Neural Networks (CNNs) with MobileNet, a lightweight architecture designed for devices with little resources, to create a deep learning-based system for bone fracture detection utilizing X-ray images. Their methods included pre-processing, data augmentation, and training with ReLU and sigmoid activation functions on a dataset of 5,000 labeled X-ray pictures from Kaggle. They achieved an outstanding 98% accuracy with MobileNet, compared to 86% with CNNs. Through depth-wise separable convolutions, the study demonstrates MobileNet's effectiveness, which qualifies it for real-time applications. Its wider usefulness is constrained by its focus on single-bone types and difficulties with mobile deployment.

In 2024, Spoorthi et al. [20] introduced a hybrid approach for hand bone fracture classification, integrating the EfficientNet-B3 deep learning model with a Support Vector Machine (SVM) classifier, achieving a high accuracy of 93.5%. To improve feature extraction, the methodology uses a dataset of 8,812 hand X-ray pictures that have been pre-processed using edge detection, denoising, Gaussian and median blurring, and grayscale conversion. EfficientNet-B3, fine-tuned on this dataset, extracts detailed features, which SVM then classifies to establish robust decision boundaries. The model outperforms traditional methods, such as SIFT and other CNN-based approaches, as demonstrated by low false positives and negatives in the confusion matrices and high-performance indicators including precision, recall, and F1-score.

In 2025, Aldhyani et al. [21] used a Kaggle dataset of 10,580 X-ray images to propose a deep learning framework for automated bone fracture identification. The framework uses VGG16, ResNet152V2, and DenseNet201, is trained with a 70-20-10 split using the Adam optimizer, and is supplemented with attention mechanisms and skip connections. Because of its deep connection, DenseNet201 fared better than the others, obtaining 97.35% accuracy, 97.41% F1-score, 97.78% sensitivity, and 97.06% specificity. The accuracy scores for VGG16 and ResNet152V2 were 96.55% and 92.15%, respectively. The method is restricted by the diversity of datasets, but it enhances feature extraction for clinical diagnosis. Future research proposes combining multi-modal imaging and transformer attention.

In 2025, Torne et al. [22] evaluated VGG-16, VGG-16 with Random Forest, ResNet-50 with SVM, and EfficientNetB0 with XGBoost on a dataset of 1,129 images, spanning 10 fracture categories, in order to compare deep learning models for bone fracture classification using X-ray images. Because of the deep architecture of the VGG-16 and Random Forest's capacity to reduce overfitting, the VGG-16 and its ensemble with Random Forest obtained the greatest accuracy of 95%, along with higher precision, recall, and F1 scores. EfficientNetB0 with XGBoost fared poorly at 41%, most likely because it was ineffective with grayscale X-ray pictures, but ResNet-50 with SVM produced a strong 93% accuracy. With data taken from pre-trained CNNs and categorized using conventional ML algorithms, the study used ensemble and transfer learning approaches. Grad-CAM images were used to confirm the classification of clinically significant regions.

3. METHODOLOGY

3.1 Dataset

Two classes comprise the dataset used in the present study: non-fractured and fractured. The 3367 images in the FracAtlas data set, which is accessible on Kaggle, were used to create the non-fractured X-ray images [18]. According to bone fracture detection, the 4147 images in the Roboflow data set were used to obtain the fracture X-ray images [19]. Thus, 7514 X-ray pictures make up the entire dataset. Stratified sampling was used to maintain class distribution across splits, dividing the dataset into 80% training (6011 images), 10% validation (751 photos), and 10% testing (752 images).

3.2 Training of ResNet50 and EfficientNetB0

The model architecture for ResNet50 and EfficientNetB0, including their block structures, is presented in Figure 1. Traditional training comprises eight main steps after data selection, as shown in Figure 2 and illustrated below:

- Data Pre-processing: it is enhancing the model's reliability. There are some necessary operations that must be applied to images that have been loaded such as converted to RGB, resized all images to 224×224, and normalized to [0, 1].
- Data Augmentation: it is a method to increase the number of data set files through applying various operations [23]. To rectify class imbalance, ImageDataGenerator was used to apply rotation, shifting (both vertically and horizontally), and zooming to the Fractured class.
- Model Selection: This study utilized pre-trained ResNet50 or EfficientNetB0 models, which were originally trained on the ImageNet dataset (a large-scale image database widely used in computer vision research, particularly for training and evaluating deep learning models). It was created as part of the ImageNet Large Scale Visual Recognition Challenge (ILSVRC). During the experiment, the layers of these models were "frozen," meaning their weights were not updated during training, allowing only the newly added layers to be trained for the specific task (likely bone fracture detection). This approach leverages the pre-trained models' learned features while adapting them to a new task, resulting in reduced training time.
- Model Layers: Added a GlobalAveragePooling2D layer to reduces the spatial dimensions of the feature maps by averaging each feature map, enhancing the model's effectiveness and fewer possibilities to overfit "The architecture for both EfficientNetB0 and ResNet50 included the addition of a single GlobalAveragePooling2D layer to reduce feature map dimensions, followed by Dense layers for classification". a Dense layer it is a fully connected layer with 1024 neurons and Rectified Linear Unit activation is added to learn complex patterns from the pooled features. And a Dense layer as final fully connected layer with one neuron, and Sigmoid activation is added for binary classification. These layers adapt the pre-trained model to the binary classification task by transforming its features into a single probability score as illustrated in the Figures 1(a) and 1(b).

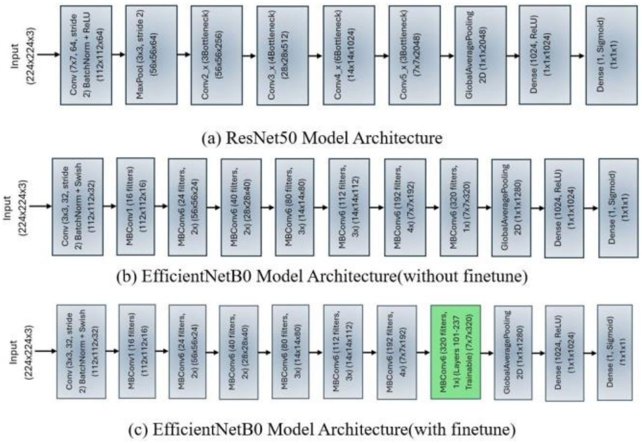


Figure 1. Model architecture

Through Figure 1(a), the "Bottleneck" in the architecture diagram refers to a specialized residual block in ResNet50, consisting of 1x1, 3x3, and 1x1 convolutional layers, designed to reduce computational complexity by narrowing channel dimensions before expanding them with skip connections enhancing gradient flow [24, 25]. Similarly in Figure 1(b), the "BatchNorm + Swish" denotes batch normalization stabilizing training by normalizing layer inputs across each mini-batch, followed by the Swish activation function enhancing non-linearity and performance [26].

- This study's feature extraction method extracted hierarchical visual characteristics from X-ray pictures using pre-trained ResNet50 and EfficientNetB0 models with frozen layers that were trained on the ImageNet dataset. These features included low-level elements such as edges, textures, and gradients (e.g., bone boundaries), mid-level patterns like shapes and contours (e.g., bone outlines), and high-level semantic information (e.g., potential fracture lines). The extracted feature maps were transformed into fixed-length vectors using a GlobalAveragePooling2D layer, enabling the subsequent Dense layers to perform binary classification.
- During the training phase, a binary cross-entropy loss function optimized with the Adam optimizer was used to train the EfficientNetB0 and ResNet50 models. In order to train the custom classification layers (GlobalAveragePooling2D, Dense 1024, and Dense 1), the models were first trained using a learning rate of 0.001, a batch size of 16, and 10 epochs, while maintaining the pre-trained base layers frozen. For EfficientNetB0, an additional fine-tuning phase was implemented, unfreezing layers 101 to 237, reducing the learning rate to 0.0001, and training for 5 extra epochs to enhance performance on the fracture detection task. The training process utilized a stratified dataset split (80% training, 10% validation, 10% test) to ensure balanced class representation, with data augmentation applied to the fractured class to address potential imbalances, as detailed in Table 1.
- EfficientNetB0 Fine-Tuning: Fine-tuning of EfficientNetB0 involved unfreezing the model (originally frozen), then freezing the first 100 layers while allowing

layers 101 to 237 to train as shown in Figure 1(c). The model was recompiled with a lower learning rate of 0.0001 and trained for 5 additional epochs to enhance performance on the fracture detection task. While, we maintained the pre-trained weights of ResNet50 frozen to ensure a robust baseline for comparison with EfficientNetB0, leveraging its strong feature extraction capabilities learned from ImageNet. This approach was adopted based on the acceptable initial accuracy of 80.05% and computational considerations, while reserving fine-tuning as a potential enhancement for future investigations to further improve performance if necessary.

- Evaluation of the models was conducted on the test set, comprising approximately 10% of the total dataset (751 images, with a balanced distribution of fractured and non-fractured X-ray images), using a probability threshold of 0.5 to classify predictions as either fractured or non-fractured. The performance was assessed using accuracy as the primary metric, alongside comprehensive classification reports that included precision, recall, and F1-score for both classes, providing a detailed insight into the models' ability to detect bone fractures. Predictions were generated by applying the trained models to the test images, with results compared against ground truth labels to compute these metrics, ensuring a robust evaluation of the EfficientNetB0 and ResNet50 architectures under the defined experimental conditions.

Table 1. Training parameters

Parameter	EfficientNetB0 (Initial Training)	EfficientNetB0 (Fine-Tuning)	ResNet50
Loss Function	Binary Cross-Entropy	Binary Cross-Entropy	Binary Cross-Entropy
Optimizer	Adam	Adam	Adam
Learning Rate	0.001	0.0001	0.001
Batch Size	16	16	16
Number of Epochs	10	5	10
Frozen Layers	All (0 to 237)	First 100 (0 to 100)	All (0 to 175)
Trainable Layers	Custom Layers (3)	Layers 101 to 237	Custom Layers (3)
Data Augmentation	Yes (Fractured class only)	No	Yes (Fractured class only)
Dataset Split	80% Train, 10% Validation, 10% Test	80% Train, 10% Validation, 10% Test	80% Train, 10% Validation, 10% Test
Input Shape	(224, 224, 3)	(224, 224, 3)	(224, 224, 3)
Activation (Final Layer)	Sigmoid	Sigmoid	Sigmoid

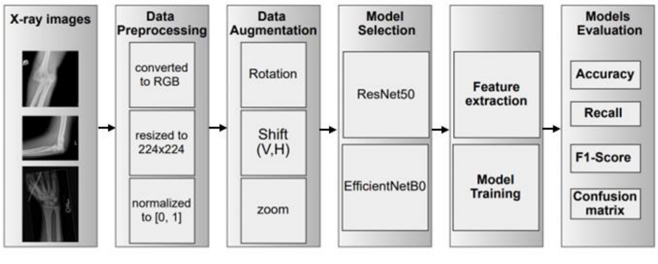


Figure 2. Workflow of bone fracture classification using traditional training

3.3 Hybrid approaches (ResNet50 and EfficientNetB0 with SVM)

Hybrid approaches comprise essentially nine main steps after data selection: as shown in Figure 3, and described below:

- Data Pre-processing: Same as Traditional Training.
- Data Augmentation: Same as Traditional Training
- Feature Extraction: Features were extracted using either pre-trained ResNet-50 or EfficientNet-B0 (with frozen layers).
- Feature Normalization: The StandardScaler, a pre-processing tool from the scikit-learn module in Python that is used to standardize features in a dataset, was used to apply feature normalization. The method, also known as Z-score normalization or standardization, modifies the data so that the standard deviation is one and the mean is zero for each feature. At this point, the extracted features are scaled to have a mean of zero and a standard deviation of one in order to guarantee consistent feature distributions and improve the performance of the subsequent machine learning classifier.
- Dimensionality Reduction: The data's dimensionality was decreased using Principal Component Analysis (PCA), which preserved 95% of the variance while reducing the feature dimensions. This technique reduces noise and computing complexity while maintaining the most crucial information for classification by distilling the original characteristics into a smaller set of uncorrelated primary components, with the 95% threshold selected to balance information retention and computational efficiency.
- Data Balancing: In order to improve model performance, SMOTE was employed to address the class imbalance in the training set, initially comprising 6011 images (Fractured: 3318, Non-Fractured: 2693), by generating synthetic samples for the minority non-fractured class to ensure equal representation, resulting in a balanced set of 6636 images (Fractured: 3318, Non-Fractured: 3318). The sampling strategy was set based on the majority fractured class size (3318) to adjust the oversampling ratio, aligning the non-fractured class to match, thus achieving equal class distribution and confirming robust training.
- SVM Training: SVM was trained using GridSearchCV to optimize hyperparameters, selecting C=10, gamma=0.01, and an RBF kernel, ensuring robust performance. StratifiedKFold cross-validation was employed to maintain class distribution across folds, enhancing the model's generalizability for fracture detection.
- Threshold Optimization: The optimal threshold (ResNet50: 0.79, EfficientNetB0: 0.45) was determined using the Precision-Recall curve to maximize the F1-Score.
- Evaluation: Using the ideal threshold, models were assessed on the test set.

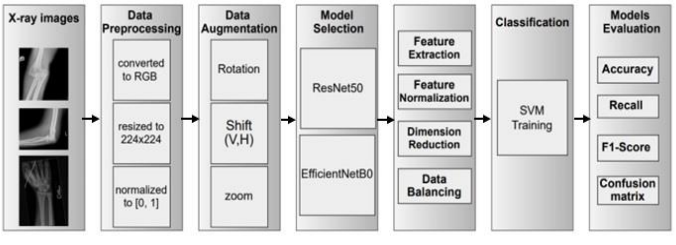


Figure 3. Workflow fracture of classification using ResNet50 and EfficientNetB0 with SVM

4. RESULTS

Evaluating the performance of model and dependability is essential to determining the bone fraction classification system's efficacy. To do this, we used commonly used metrics, including a confusion matrix, accuracy, recall, and precision. The models were evaluated and implemented with scikit-learn and TensorFlow. A CPU-only system, lacking a GPU, was used for training in order to mimic scenarios with limited resources, ensuring reproducibility across diverse hardware without reliance on specialized equipment. Python 3.8 with TensorFlow 2.5.0 was used for all tests on an HP ProBook 450 G4 laptop with the following specs: 32 GB RAM, Windows 10 Pro 64-bit, Intel Core i7-7500U CPU @ 2.70 GHz (4 CPUs, ~2.9 GHz).

4.1 Accuracy

Since accuracy calculates the proportion of accurately predicted data to all anticipated data, it is sometimes regarded as the most explicit performance metric [27]. Below is the formulation of the Eq. (1):

$$\text{Accuracy} = \text{TP} + \text{TN}/\text{TP} + \text{FP} + \text{TN} + \text{FN} \tag{1}$$

- Recall: The following Eq. (2) illustrates this ratio, which is often referred to as sensitivity or true positive rate: accurately predicted positive observations divided by total positive observations [28].

$$\text{Recall} = \text{TP}/\text{TP} + \text{FN} \tag{2}$$

- Precision: The total number of predictions the model makes is used to determine its precision. The link between the total predictions (TP + FP) and the prediction percentage (TP) is then obtained by dividing the number of right predictions by the total predictions [29, 30]. Below is Eq. (3) that illustrates this information.

$$\text{Precision} = \text{TP}/\text{TP} + \text{FP} \tag{3}$$

where: The term "True Positive" (TP) describes X-ray pictures that are accurately classified as broken [18]. Images that are appropriately classified as non-fractured are referred to as True Negatives (TN). When non-fractured X-ray images are mistakenly classified as fractured, this is known as a false positive (FP). When broken X-ray scans are mistakenly classified as non-fractured, this is known as a False Negative (FN).

4.2 Classification performance

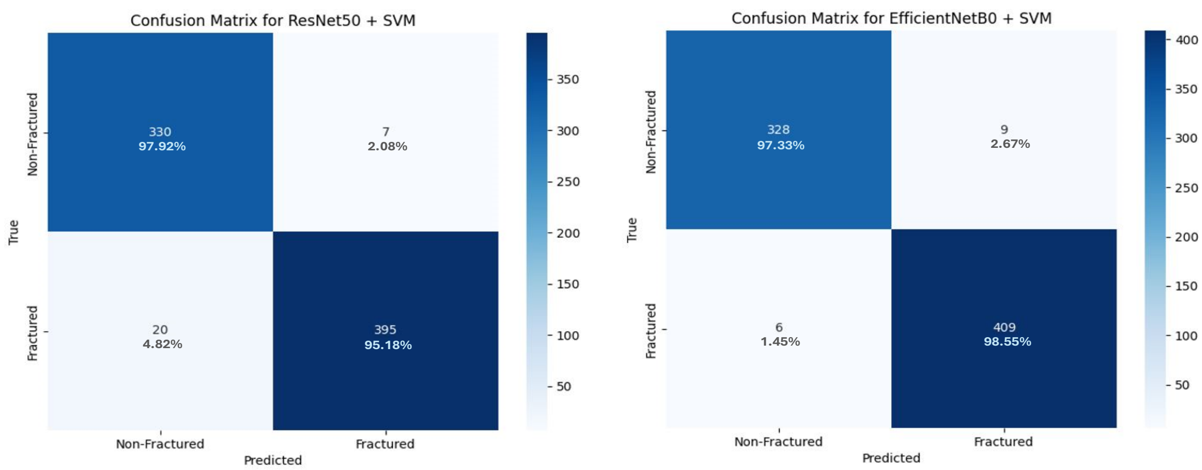
Table 2 summarizes the performance of the four approaches that implemented based on datset consists of 752 images (415 Fractured, 337 Non-Fractured).

Table 2. Comparison of classification performance

Model	Test Accuracy	Recall (Fractured)	F1-Score (Fractured)	Total Time (Minutes)
ResNet50	80.05%	0.72	0.80	83.96
ResNet50 with SVM	96.41%	0.95	0.97	38.47
EfficientNetB0	95.35%	1.00	0.96	63.92
EfficientNetB0 with SVM	98.01%	0.99	0.98	20.44

As shown in the Table 2 above, ResNet50 obtained a test accuracy of 80.05%, with a low Recall of 0.72 for Fractured cases, indicating that 28% of fractures were missed. The F1-Score for Fractured cases was 0.80, reflecting the poor balance between Precision and Recall. While ResNet50 with SVM improved the test accuracy to 96.41%, with a Recall of 0.95 for Fractured cases, missing only 5% of fractures and F1-Score increased to 0.97, showing a better balance. Also, EfficientNetB0 (traditional training): Obtained a test accuracy of 95.35%, with a perfect Recall of 1.00 for Fractured cases, detecting all fractures. The F1-Score was 0.96, slightly affected by a lower Precision. Finally, EfficientNetB0 with SVM: Recorded the highest test accuracy of 98.01%, with a Recall of 0.99 for Fractured cases and an F1-Score of 0.98, demonstrating excellent performance.

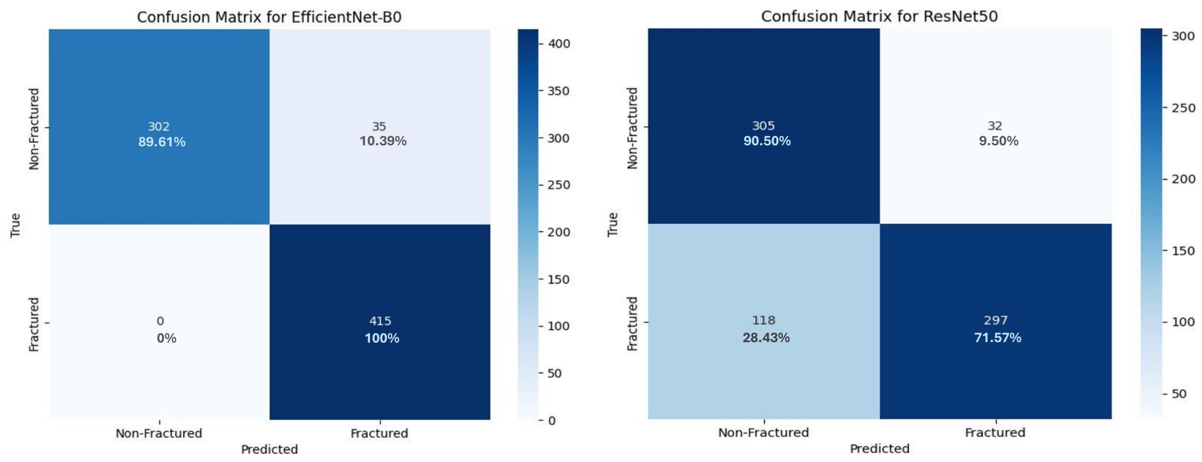
Figure 4 and Figure 5 present the confusion matrices for the hybrid and the traditional approaches respectively.



(a) ResNet50 with SVM

(b) EfficientNetB0 with SVM

Figure 4. Confusion matrix for Hybrid approaches



(a) EfficientNetB0 (b) ResNet50

Figure 5. Confusion matrix for traditional approaches

The confusion matrix Figure 4(a) is correctly identified 395 out of 415 Fractured cases (TP) and 330 out of 337 Non-Fractured cases (TN). It misclassified 7 Non-Fractured cases as Fractured (FP) and 20 Fractured cases as Non-Fractured (FN). However, the confusion matrix for EfficientNetB0 with SVM (Figure 4(b)) correctly identified 409 out of 415 Fractured cases and 328 out of 337 Non-Fractured cases, with only 9 False Negatives and 6 False Positives, demonstrating superior performance.

Figure 5(a) shows the confusion matrix for EfficientNetB0 that correctly identified 415 out of 415 Fractured cases (TP) and 302 out of 337 Non-Fractured cases (TN), with only 35 Non-Fractured cases as Fractured (FP). However, the confusion matrix for ResNet50 (Figure 5(b)) correctly identified 297 out of 415 Fractured cases and 305 out of 337 Non-Fractured cases. It misclassified 35 False Negatives and 118 False Positives.

4.3 Loss function and Precision-Recall

Figure 6 illustrates the loss functions of training and validation over 10 epochs for the traditional ResNet50 and EfficientNetB0. ResNet50 exhibited higher loss values, indicating suboptimal learning, while EfficientNetB0's loss decreased more effectively due to fine-tuning.

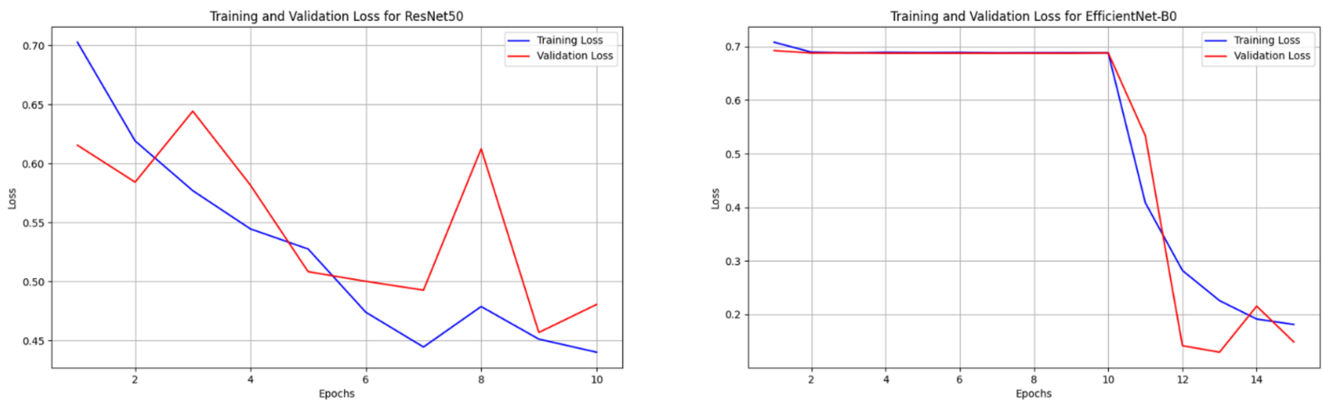


Figure 6. Loss function of training and validation loss for ResNet50 and EfficientNetB0

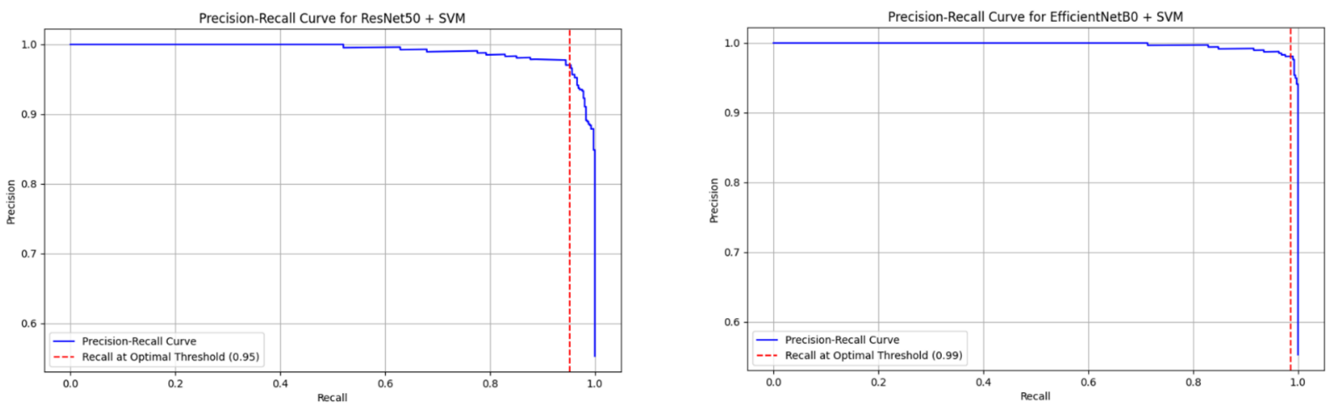


Figure 7. Precision-Recall curve for ResNet50 and EfficientNetB0 with SVM

Finally, Figure 7 presents Precision-Recall curves for hybrid approaches (ResNet50 and EfficientNetB0 with SVM), with vertical lines indicating recall at the optimal threshold 0.95 for ResNet50 with SVM and 0.99 for EfficientNetB0 with SVM, demonstrating how precision and recall are traded off.

5. CONCLUSION

This study demonstrates the superiority of hybrid approaches over traditional training for the classification of bone fractures. The hybrid approaches, combining ResNet50 with SVM and EfficientNetB0 with SVM, demonstrated superior performance compared to traditional training methods across multiple metrics. EfficientNetB0 with SVM achieved an impressive accuracy of 98.01% and a near-perfect recall of 0.99 for fractured cases, establishing it as a highly reliable option for clinical applications. The EfficientNet-B0 + SVM outperformed others due to its effective feature fusion mechanism, where EfficientNet-B0's optimized feature extraction complements SVM's precise classification.

Similarly, ResNet50 with SVM significantly enhanced performance over direct ResNet50 training, improving accuracy from 80.05% to 96.41% and recall from 0.72 to 0.95, effectively addressing challenges such as poor generalization and class imbalance inherent in traditional methods.

On the other hand, EfficientNetB0 with SVM was particularly efficient, requiring just 20.44 minutes compared to 63.92 minutes for direct training (a 68% reduction), while ResNet50 with SVM reduced training time from 83.96 minutes to 38.47 minutes (a 54% reduction). The hybrid approaches yielded steady and dependable results, which made them ideal for clinical deployment, particularly in settings with constrained computational resources, in contrast to older methods that gave inconsistent results across tests.

REFERENCES

[1] Noureen, A., Zia, M.A., Adnan, A., Hashim, M. (2023). Analysis and classification of bone fractures using machine learning techniques. E3S Web of Conferences, 409: 02015. https://doi.org/10.1051/e3sconf/202340902015

[2] Sharma, S. (2023). Artificial intelligence for fracture diagnosis in orthopedic X-rays: Current developments and future potential. SICOT-J, 9: 21. https://doi.org/10.1051/sicotj/2023018

[3] Younis, N.K., Qahtan, M.H., Ahmed, M.R. (2025). A review of artificial intelligence techniques for medical image enhancement. International Journal of Computational and Electronic Aspects in Engineering, 6(2): 98-107. https://doi.org/10.26706/ijceae.6.2.20250406

[4] Younis, A.K., Younis, B.M., Jarjees, M.S. (2022). Hardware implementation of Sobel edge detection system for blood cells images-based field programmable gate array. Indonesian Journal of Electrical Engineering and Computer Science, 26(1): 86-95. https://doi.org/10.11591/ijeecs.v26.i1.pp86-95

[5] Alhelal, D., Younis, A.K., Al-Mallah, R.H.A. (2021). Detection of brain stroke in the MRI image using FPGA. TELKOMNIKA Telecommunication, Computing Electronics and Control, 19(4): 1307-1315. https://doi.org/10.12928/TELKOMNIKA.v19i4.18988

[6] Meena, T., Roy, S. (2022). Bone fracture detection using deep supervised learning from radiological images: A paradigm shift. Diagnostics, 12(10): 2420. https://doi.org/10.3390/diagnostics12102420

[7] Riayth, A.M., Jarjees, M.S., Younis, A.K. (2023). A review of sickle cells classification system based on microscopic images. AIP Conference Proceedings, 2862(1): 020045. https://doi.org/10.1063/5.0181048

[8] Younis, Y.B., Al Azzo, F. (2025). A technique for retinal detachment detection manipulating YOLOv8 models. NTU Journal of Engineering and Technology, 4(1): 10-23. https://doi.org/10.56286/rxck4y17

[9] Marletta, D.A., Nanni, M., Giuca, G., Sanzarello, I., Zampogna, B., Leonetti, D. (2024). The role of artificial intelligence in predicting and managing pediatric fracture overgrowth: A comprehensive review. Applied Sciences, 14(24): 11652. https://doi.org/10.3390/app142411652

[10] Abdulloh, F.F., Fattah, F.A., Wulandari, D., Mustopa, A. (2025). Comparison of efficientnet and YOLOv8 algorithms in motor vehicle classification. SITEKNIK: Sistem Informasi, Teknik dan Teknologi Terapan, 2(3): 249-259. https://doi.org/10.5281/zenodo.16561038

[11] Ameen, R.H.M., Basheer, N.M., Younis, A.K. (2023). Breast cancer diagnosis based on support vector machine techniques. Indonesian Journal of Electrical Engineering and Computer Science, 32(1): 236-243. https://doi.org/10.11591/ijeecs.v32.i1.pp236-243

[12] Ahmed, H.A., Mohammed, E.A. (2022). Detection and classification of the osteoarthritis in knee joint using transfer learning with convolutional neural networks (CNNs). Iraqi Journal of Science, 63(11): 5058-5071. https://doi.org/10.24996/ijs.2022.63.11.40

[13] Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P. (2002). SMOTE: Synthetic minority over-sampling technique. Journal of Artificial Intelligence Research, 16: 321-357. https://doi.org/10.1613/jair.953

[14] Castro-Gutierrez, E., Estacio-Cerquin, L., Gallegos-Guillen, J., Obando, J.D. (2019). Detection of acetabulum fractures using X-ray imaging and processing methods focused on noisy images. In 2019 Amity International Conference on Artificial Intelligence (AICAI), Dubai, United Arab Emirates, pp. 296-302. https://doi.org/10.1109/AICAI.2019.8701297

[15] Hržić, F., Štajduhar, I., Tschauner, S., Sorantin, E., Lerga, J. (2019). Local-entropy based approach for X-ray image segmentation and fracture detection. Entropy, 21(4): 338. https://doi.org/10.3390/e21040338

[16] Abbas, W., Adnan, S.M., Javid, M.A., Majeed, F., Ahsan, T., Hassan, S.S. (2020). Lower leg bone fracture detection and classification using faster RCNN for X-rays images. In 2020 IEEE 23rd International Multitopic Conference (INMIC), Bahawalpur, Pakistan, pp. 1-6. https://doi.org/10.1109/INMIC50486.2020.9318052

[17] Yadav, D.P., Rathor, S. (2020). Bone fracture detection and classification using deep learning approach. In 2020 International conference on power electronics & IoT applications in renewable energy and its control (PARC), Mathura, India, pp. 282-285. https://doi.org/10.1109/PARC49193.2020.236611

[18] Ahmed, K.D., Hawezi, R. (2023). Detection of bone fracture based on machine learning techniques. Measurement: Sensors, 27: 100723.

https://doi.org/10.1016/j.measen.2023.100723

[19] Thota, S., Kandukuru, P., Sundaram, M., Ali, A., Basha, S.M., Bindu, N.H. (2024). Deep learning based bone fracture detection. In 2024 International Conference on Smart Systems for applications in Electrical Sciences (ICSSES), Tumakuru, India, pp. 1-7. https://doi.org/10.1109/ICSSES62373.2024.10561360

[20] Spoorthi, J.S., Keerthi, K., Divya Sankari, S., Pandiyaraji, V. (2024). Optimizing bone fracture diagnosis: A hybrid approach using EfficientNet-B3 and SVM for accurate classification. In 2024 International Conference on Emerging Research in Computational Science (ICERCS), Coimbatore, India, pp. 1-7. https://doi.org/10.1109/ICERCS63125.2024.10894779

[21] Aldhyani, T., Ahmed, Z.A., Alsharbi, B.M., Ahmad, S., Al-Adhaileh, M.H., Kamal, A.H., Nazeer, J. (2025). Diagnosis and detection of bone fracture in radiographic images using deep learning approaches. Frontiers in Medicine, 11: 1506686. https://doi.org/10.3389/fmed.2024.1506686

[22] Torne, S., Shetty, D.K., Makkithaya, K., Hegde, P., et al. (2025). VGG-16, VGG-16 with random forest, Resnet50 with SVM, EfficientNetB0 with XGBoost-Enhancing bone fracture classification in X-Ray using deep learning models. IEEE Access, 13: 25568-25577. https://doi.org/10.1109/ACCESS.2025.3534818

[23] Chaudhry, M., Shafi, I., Mahnoor, M., Vargas, D.L.R., Thompson, E.B., Ashraf, I. (2023). A systematic literature review on identifying patterns using unsupervised clustering algorithms: A data mining perspective. Symmetry, 15(9): 1679. https://doi.org/10.3390/sym15091679

[24] Rezende, E., Ruppert, G., Carvalho, T., Ramos, F., De Geus, P. (2017). Malicious software classification using transfer learning of resnet-50 deep neural network. In 2017 16th IEEE international conference on machine learning and applications (ICMLA), Cancun, Mexico, pp. 1011-1014. https://doi.org/10.1109/ICMLA.2017.00-19

[25] He, K., Zhang, X., Ren, S., Sun, J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770-778. https://doi.org/10.1109/CVPR.2016.90

[26] Tan, M., Le, Q. (2019). EfficientNet: Rethinking model scaling for convolutional neural networks. In Proceedings of the 36th International Conference on Machine Learning, pp. 6105-6114.

[27] Tong, W., Liu, S., Gao, X. Z. (2021). A density-peak-based clustering algorithm of automatically determining the number of clusters. Neurocomputing, 458: 655-666. https://doi.org/10.1016/j.neucom.2020.03.125

[28] Canever, H., Wang, X. (2023). Network traffic classification using Unsupervised Learning: A comparative analysis of clustering algorithms. https://hal.science/hal-04149117.

[29] Abdulwahab, E., Younis, A.K. (2024). A novel sound classification based on lightweight ResNet-34 for emergency detection. In National Conference on New Trends in Information and Communications Technology Applications, pp. 81-96. https://doi.org/10.1007/978-3-031-87076-7_6

[30] Strobl, M., Sander, J., Campello, R.J., Zaïane, O. (2020). Model-based clustering with hdbscan. In Joint European Conference on Machine Learning and Knowledge Discovery in Databases, pp. 364-379. https://doi.org/10.1007/978-3-030-67661-2_22