Optimization of Bone Fracture Diagnosis in X-Ray Systems for Medical Decision-Making Efficiency

1st Dr. Abdul Haris Rangkuti, S.Kom., M.M., M.Si.
Computer Science
BINUS University
Bandung
rangku2000@binus.ac.id

2nd Albany Siswanto
Computer Science
BINUS University
Bandung
albany.siswanto@binus.ac.id

3rd Djuhar Geusan Harum Manik
Computer Science
BINUS University
Bandung
djuhar.manik@binus.ac.id

4th Naufal Ghifari Hidayat
line 2: *Computer Science*
BINUS University
Bandung
naufal.hidayat001@binus.ac.id

Abstract*—In the midst of an accelerating digital transformation, the adoption of artificial intelligence (AI) in medicine and healthcare has shown significant progress in recent years, particularly in enhancing diagnostic accuracy and efficiency. A key challenge in diagnostics is the difficulty medical personnel sometimes face in thoroughly understanding and analyzing a disease. Manual analysis by medical staff can be susceptible to subjectivity, fatigue, and human error. This research aims to develop a straightforward deep learning-based model for bone fracture detection using a convolutional neural network (CNN) architecture, designed to automate the diagnostic process with high accuracy. The model was trained on an X-ray image dataset compiled from various sources and evaluated using metrics such as accuracy, sensitivity, and specificity to comprehensively assess its performance. Integrating AI into radiological analysis not only improves the efficiency of the diagnostic process but also accelerates the application of bioinformatics technology in global medical practice. This study is expected to make a tangible contribution to the evolution of modern healthcare systems, reduce the workload of medical professionals, and enhance public confidence in technology-based diagnostic outcomes.

Keywords*—CNN, Efficiency, Healthcare, Artificial Intelligence

I. INTRODUCTION

Over the past few decades, the development of artificial intelligence (AI) has permeated various scientific disciplines, including bioinformatics and medical science. Bioinformatics, once primarily focused on genomic and proteomic data analysis, has now become a central pillar in the transformation of digital health systems, with AI as a key driving force [1]. This trend is bolstered by substantial investments from technologically advanced nations like the United States, China, the European Union, and Japan, which have allocated billions of dollars to AI development to improve healthcare system efficiency [2].

Deep learning is one AI infrastructure and framework experiencing exponential growth. Its application in radiology aims to automate detection processes, including the identification of bone fractures through X-ray analysis. Accurate and rapid bone fracture detection is a critical aspect of medical diagnosis. However, conventional methods relying on medical personnel often encounter obstacles such as time constraints, subjectivity, and fatigue, which can lead to diagnostic errors [3]. Conversely, AI based on deep learning has demonstrated performance that can rival, and in some recent studies, even surpass human diagnostic capabilities [4]. Architectures like the convolutional neural network (CNN) have proven effective in recognizing complex patterns in medical images, enabling automated fracture detection with greater precision than traditional approaches [5].

In the broader technological and economic landscape, significant investments in AI are not only aimed at enhancing healthcare quality but also at creating systems more responsive to large-scale medical challenges, including the optimization of data-driven diagnostics. Technology firms such as Google DeepMind and IBM Watson Health have developed AI models for medical science analysis [6], while research institutions in developed countries continue to innovate by refining algorithms for implementation in modern healthcare systems [7].

This research seeks to develop a simple deep learning-based AI model capable of providing more optimal results for automated bone fracture diagnosis using X-rays. The goal is to reduce the workload of medical personnel, improve diagnostic accuracy, and expedite medical decision-making. The developed model will be assessed using various performance metrics, including accuracy, sensitivity, and specificity, to gauge its effectiveness in a clinical setting. By employing AI as an assistive tool in radiology, this research is anticipated to contribute to the technological revolution in

global healthcare systems, while also strengthening the integration of bioinformatics in big data-driven medical practice [8].

II. METHODOLOGY

This study employs a deep learning approach centered on object detection to identify bone fractures in X-ray images. The methodology is structured into several key stages: data collection and annotation using roboflow [9], data preprocessing, model development and training, and finally, model performance evaluation.

A. Data Collection and Annotation

X-ray image datasets were obtained from two main sources, namely the Kaggle and Roboflow platforms [10]. The Kaggle dataset is used as the main database containing various types of bone radiology images, while Roboflow is used for the annotation process and manual bounding box creation of fracture parts identified in the image. The annotation process is carried out consistently to ensure accurate ground truth quality in the training stage [11].

B. Data Preprocessing

Before being fed into the training models, the collected X-ray images underwent several preprocessing steps:

1) Resizing: All images were uniformly resized to meet the input dimension requirements of each model architecture (e.g., 416 x 416 pixels for YOLO).

2) Normalization: Image pixel values were normalized to a [0,1] range. This step helps in accelerating convergence during the model training process [12].

3) Augmentation: Data augmentation techniques, including rotation, flipping, and adaptive contrast adjustments, were applied [13]. These methods serve to increase the diversity of the training data and help in preventing model overfitting [14].

C. Deep Learning Model Development

This research implemented and compared two deep learning architectures:

1) YOLOv11 Nano: As the latest and fastest iteration in the Ultralytics YOLO series of real-time object detectors, YOLOv11 redefines possibilities with its state-of-the-art accuracy, speed, and efficiency. Building on the significant advancements of its predecessors, YOLOv11 introduces notable improvements in architecture and training methodologies, positioning it as a versatile option for a wide array of computer vision tasks [15].

2) YOLOv12 Nano: YOLOv12 features an attention-based architecture, a departure from the traditional CNN-based approaches of earlier YOLO models. Despite this change, it maintains the real-time inference speed critical for many applications. This model achieves leading object detection accuracy through innovative methodological advancements in attention mechanisms and overall network design, all while sustaining real-time performance [16].

Each model was implemented using established deep learning frameworks like PyTorch and TensorFlow, with *hyperparameter* configurations fine-tuned based on initial experimentation.

D. Model Training and Evaluation

All models were trained using a dataset partitioned into training (80%) and validation/testing (20%) subsets [17]. The evaluation of model performance was conducted using several key metrics:

1) Sensitivity (Recall): This metric measures the model's proficiency in correctly identifying actual fractures.

2) Precision: This measures the proportion of correctly predicted fractures relative to all instances predicted as fractures [18].

3) Mean Average Precision (mAP): Specifically employed for object detection models like YOLO, mAP assesses the accuracy of the detected bounding boxes.

III. RESULT & DISCUSSION

Upon completion of the training phase for all models, their evaluation results were compared to pinpoint the most effective model for detecting fractures in X-ray images. The analysis focused on the inherent strengths and limitations of each architecture, alongside their potential applicability in clinical decision support systems.

A. YOLOv8 Large, YOLOv11 Nano, and YOLOv12 Nano

In this study, our configuration primarily centered on YOLOv11 due to several compelling reasons: its processing speed, a more lightweight model size, and its efficiency in adapting to our available computational resources. The training results from YOLOv11 indicated generally strong performance, especially for classes with a larger volume of data (such as Comminuted Fracture, Hairline Fracture, and Avulsion Fracture) [19]. However, certain classes, including GreenStick Fracture, Fracture Dislocation, and spiral fracture, still showed somewhat lower values for precision, recall, or mAP.

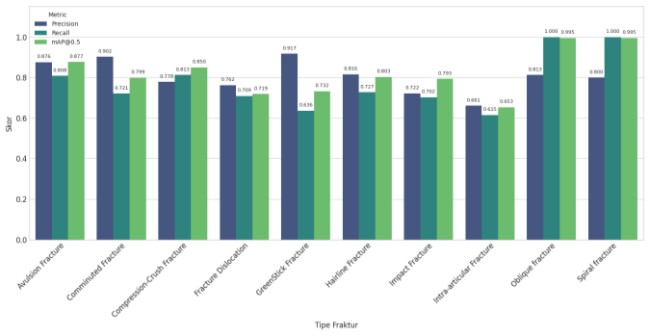

Fig. 3.1 Finetuned YOLOv11 Nano

YOLOv11 Nano stands out as a lighter and more stable version when compared to YOLOv12 Nano. It is engineered for high efficiency, making it particularly suitable for deployment in scenarios like edge computing or real-time applications where computational resources are constrained [20]. Consequently, we opted for YOLOv11 Nano to achieve superior inference speed and a more compact model size relative to YOLOv12 Nano, while still yielding a robust detection model that avoids significant overfitting or underfitting [21]. In terms of accuracy, the YOLOv11 Nano model we utilized demonstrated commendable detection capabilities, particularly for data-rich classes like Comminuted Fracture and Hairline Fracture. As indicated in the metrics (derived from the original Picture 3.1), YOLOv11 Nano achieved an mAP@0.5 of 0.7809, a Precision of 0.8205, and a Recall of 0.7337. Furthermore, YOLOv11 exhibits an advantage in managing data imbalance [22]. This is evident in its capacity to recognize less common classes, such as Oblique fracture and Intra-articular fracture, even with very limited data samples. Had we employed YOLOv12 Nano without specific adjustments for data distribution or tailored augmentation strategies, the model would likely have struggled with these minor classes, as its default performance is not as adept as YOLOv11 Nano when dealing with complex and imbalanced medical datasets [23].

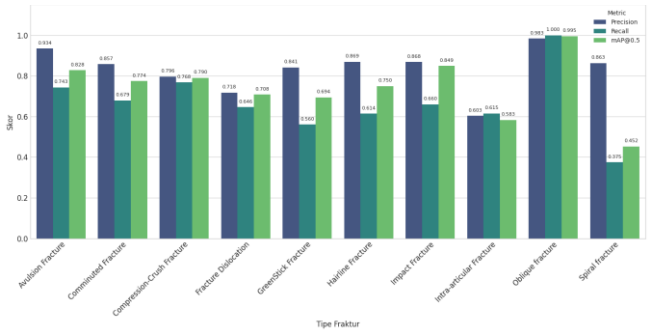

Fig. 3.2 Finetuned YOLOv12 Nano

When we experimented with YOLOv12, an objective look at the metrics revealed a slight dip in performance for YOLOv12 Nano. YOLOv12, being the newest addition to the YOLO family, incorporates several enhancements in its architecture and training algorithms [24]. These include optimizations in the backbone and head structures, alongside features like dynamic label assignment and cross-layer feature aggregation. While these are advancements, they also contribute to a model that is considerably more resource-intensive than YOLOv11 Nano. With our specific training dataset, achieving performance superior to YOLOv11 Nano would necessitate careful attention to data distribution, specialized augmentation techniques, and access to more powerful computational resources.

B. Bounding Box YOLOv11 Nano

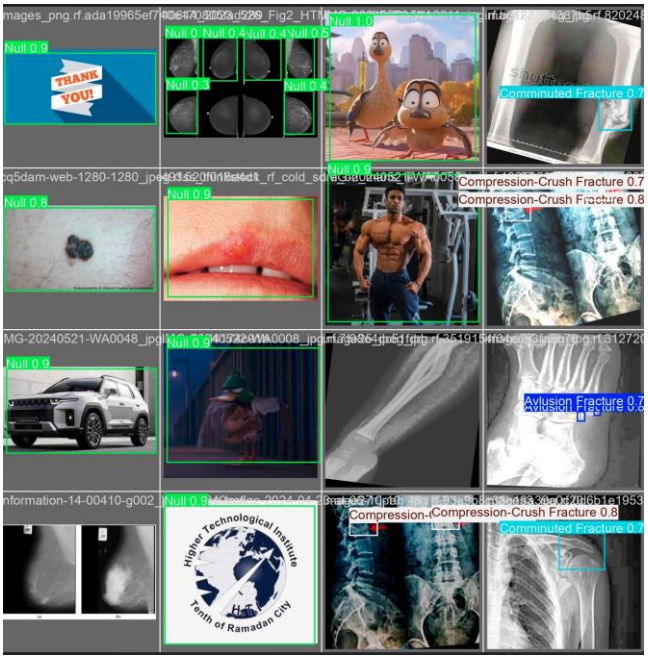

Fig. 3.3 Bounding Box

The visual output from the YOLOv11n object detection model, after training, shows how well it can detect objects in various input images. Each colored bounding box indicates a successful recognition of an object by the model. The recognized objects include different types of bone fractures, such as "Comminuted Fracture," "Compression-Crush Fracture," and "Avulsion Fracture." It also identifies other, unrelated items labeled as "Null." The model can detect several types of bone fractures with fairly high confidence scores, usually between 0.7 and 0.8. This indicates that the model has learned to identify visual features that indicate fractures in X-ray images. However, many detections happened on images that were not X-rays. These included objects like cars, cartoons, logos, and people. The model labeled all these non-target objects as "Null" with high confidence scores, often between 0.9 and 1.0. This suggests that while processing a varied set of input data, the model was also effectively identifying and categorizing these non-target objects, separating them from the fracture classes it was trained to recognize.

Overall, the model demonstrated skill in telling apart images of fractured bones from images containing other objects. To improve its accuracy and general ability, more work on curating the dataset is needed. Specifically, increasing the sample size for each type of fracture would help reduce misclassifications of non-bone images and might enhance the model's focus on relevant medical imagery.

A. Metrics Evaluation

Fig. An analysis of the training performance graphs for the bone fracture object detection model shows several key trends in the loss metrics. The Box Loss metric dropped notably from about 1.5 to 1.2. This indicates an

improvement in the model's precision in predicting bounding box locations. Likewise, the Classification Loss (Cls Loss) steadily declined from 1.3 to 0.7. This means the model improved its ability to correctly classify bone fractures. The DFL Loss, however, only decreased slightly from 1.5 to 1.25. This suggests that this aspect could benefit from more optimization [25].

Looking at the evaluation metrics, Precision reached a final value of around 0.85. This means that 85% of the model's detections were accurate, with a relatively low number of false positives. Recall, although its graph was partially cut off in the provided visuals, showed an upward trend. This indicates an improving ability to identify true positive cases. An mAP50 score of about 0.7 suggests a good level of performance for basic detection tasks where an Intersection over Union (IoU) of 0.5 is considered a correct detection.

However, the lower mAP50-95 score, which ranged from 0.2 to 0.35, indicates that the model's consistency drops under stricter IoU thresholds (averaging mAP scores from 0.5 to 0.95 IoU). This decline in performance at higher IoU values likely stems from the natural variability in the shape and complexity of bone fractures. This variability makes it more challenging to achieve highly precise bounding box predictions.

B. Recall Confidence Curve YOLOv11 Nano

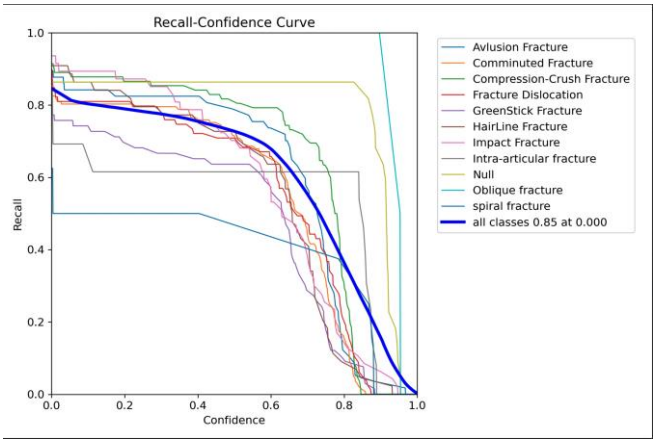

Fig 3.4 Recall Confidence Curve

The Recall-Confidence Curve for the bone fracture detection model shows areas that need improvement. Recall values are generally low, ranging from 0.0 to 0.4 across different confidence thresholds. For example, a recall of 0.4 at a confidence threshold of 0.2 indicates that only 40% of actual fractures were correctly identified at that confidence level. This highlights a significant issue with false negatives, meaning many fracture cases could be missed by the model if a moderately low confidence threshold is used [26].

The detailed breakdown of different fracture types, such as Avulsion, Comminuted, and Hairline, shown in the curve emphasizes the need for a more detailed performance evaluation for each specific category. This is particularly important for fractures that are subtle or complex, as they are likely to show lower recall rates. In turn, this means the model frequently misses these cases.

To improve the model's recall performance, several strategies can be applied. First, optimizing the confidence threshold is essential to find a better balance between recall and precision, though this curve focuses mainly on recall. Expanding the dataset, especially by increasing the number of samples for fracture classes that are often under-detected, is also important. Additionally, using targeted data augmentation techniques could help improve the detection of minor or less distinct fracture cases [27].

Clinical validation, with input from medical experts, is crucial. This step ensures that the ground truth annotations used for training and testing are accurate, especially for small or intricate fractures that often challenge automated detection systems. Overall, the Recall-Confidence Curve shows that the model currently has low sensitivity for certain fracture types at various confidence levels. This requires further refinements to the data, model settings, and evaluation strategies for each class to achieve more reliable performance in identifying all relevant fractures.

C. Precision Recall Curve YOLOv11 Nano

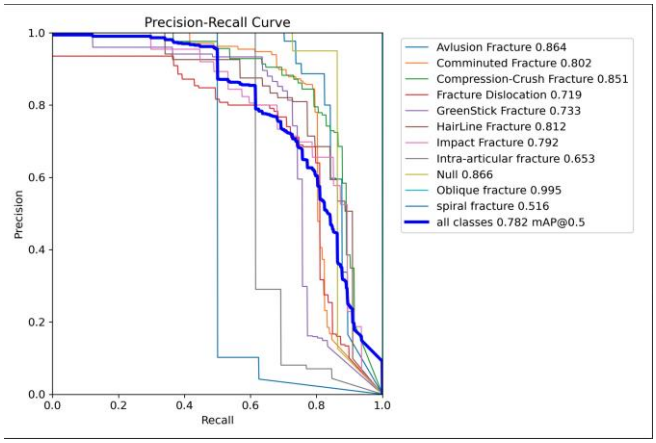

Fig 3.5 Precision Recall Curve

The Precision-Recall (PR) Curve and its metrics show that the bone fracture detection model's performance differs significantly among various types of fractures [28]. The overall mean Average Precision (mAP) of 0.782 across all classes indicates a fairly good performance when using an IoU of 0.5 for correct detection. However, examining the Average Precision (AP) scores for individual classes reveals notable differences.

For example, the 'Oblique fracture' achieved the highest AP at 0.995, indicating nearly perfect detection for this type. In contrast, the 'spiral fracture' only reached an AP of 0.516, showing the model struggles to identify this fracture pattern accurately at high precision across all recall levels. Several classes, such as 'Avulsion Fracture' (AP 0.864), 'Comminuted Fracture' (AP 0.802), and 'Hairline Fracture' (AP 0.812), performed well. Others, like 'Fracture Dislocation' (AP 0.719) and 'Intra-articular fracture' (AP 0.653), fell below the average mAP. Interestingly, the 'Null' class, which represents non-fracture cases, achieved a high AP of 0.866, showing the model's skill in correctly identifying instances without fractures.

The PR curve itself shows the trade-off: as recall (finding all positive samples) increases, precision (the percentage of positive identifications that were correct) generally decreases, and vice versa. The shape of the curve for each class provides insight into this relationship. A fairly gentle slope in the central part of the overall PR curve (the bold blue line) suggests there may be opportunities to improve the balance between precision and recall.

The lower performance on certain complex fracture types, like spiral and intra-articular fractures, indicates the need for more varied training data or specialized techniques to better handle these challenging cases. From a clinical viewpoint, it is crucial to consider this variability in performance across different fracture types. Different fractures can have unique treatment implications and varying levels of diagnostic urgency [29].

D. Precision Confidence Curve

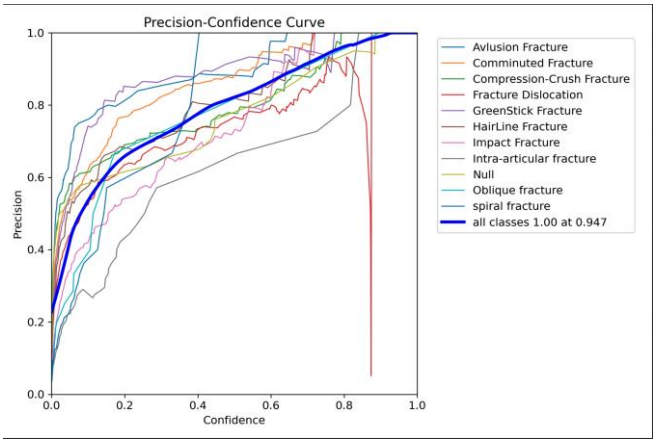

Fig 3.6 Precision Confidence Curve

The Precision-Confidence Curve for the bone fracture detection model provides valuable insights into its performance. It shows how precision changes when the confidence threshold for detection is adjusted. The curve reveals that the model can achieve perfect precision (1.00) across all fracture classes combined, but only at a very high confidence threshold (0.947, as indicated by "all classes 1.00 at 0.947") [30]. However, this result should be viewed cautiously. A high threshold maximizes precision and minimizes false positives, but it usually results in a very low recall rate, meaning many actual fractures may be missed. This pattern was also seen in the earlier Recall-Confidence Curve analysis.

Overall, the curve indicates a positive link between confidence level and precision. As the confidence threshold for classifying a detection as positive increases, so does the precision of those detections. This aligns with expectations since higher confidence detections tend to be more accurate. However, a concern is the wide range of precision values (from 0.0 to 1.0) at various confidence levels, especially for individual fracture classes, even though the combined curve is presented [31]. This suggests some instability in the model's performance or high variability across classes.

At intermediate confidence thresholds (about 0.4 to 0.6), the overall precision for all classes lies between roughly 0.75 and 0.9. This indicates that when the model operates at lower confidence thresholds (like below 0.4) to increase recall, it may produce more false positives and lower precision.

A limitation of this analysis, focused on the "all classes" curve, is the lack of a clear breakdown of precision values for individual classes at different confidence levels on this combined graph, though individual lines are shown. This information would be helpful for identifying specific fracture types that lead to drops in precision at certain confidence levels. Clinically, these results highlight the importance of balancing the trade-off between precision and recall when using the model. This often means selecting the best confidence threshold. The choice is particularly important due to the differing clinical impacts of false positives, which can cause over-diagnosis or unnecessary follow-ups, versus false negatives, which can result in missed diagnoses and delayed treatment in bone fracture assessment [32].

E. Labels

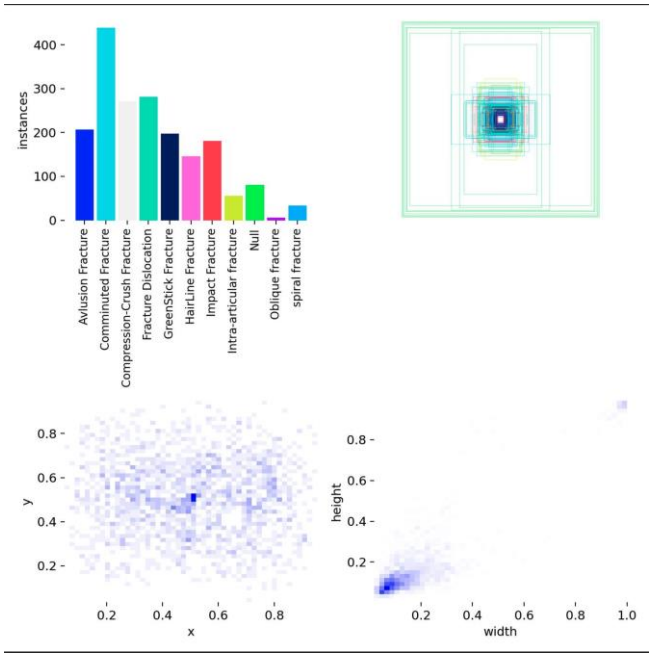

Fig 3.7 Labels

1. Distribution of Factor Values (Interpreted from visual data in the original document):

A majority of the fracture types (Avulsion, GreenStick, Hairline, Impact, Intra-articular, Oblique, Spiral) appear to follow a somewhat uniform distribution pattern, with dominant "factor values" (as described in the original text, possibly referring to normalized coordinates or feature magnitudes) around 0.6 and 0.8.

'Fracture Dislocation' and another category termed "Constrained Factor" (presumably a label from the plots) exhibit lower values, around 0.2 and 0.4, suggesting they possess characteristics distinct from the other categories [33].

2). Consistency

Seven out of the nine listed fracture types show very similar distribution patterns (with values clustering around 0.8, 0.6, 0.8, as per the original text's interpretation of its own figures).

This uniformity could reflect a consistent annotation methodology or inherently similar visual characteristics among these specific fracture types [34].

3. Anomaly

"Constrained Factor" and 'Fracture Dislocation' emerge as outliers, characterized by lower factor values (0.2-0.4).

These differences might stem from unique visual features [35], varying degrees of difficulty in annotation, or a comparatively smaller number of samples available for these particular categories.

4. Implication

The relatively uniform distribution pattern observed for most classes should generally facilitate the model's learning process for these common types.

Classes exhibiting different distributions (such as "Constrained," 'Dislocation') might necessitate specific interventions [36]. These could include augmenting the training dataset with more samples of these types, applying specialized data augmentation techniques, or potentially employing a model architecture more adept at handling such variations.

F. Confusion Matrix

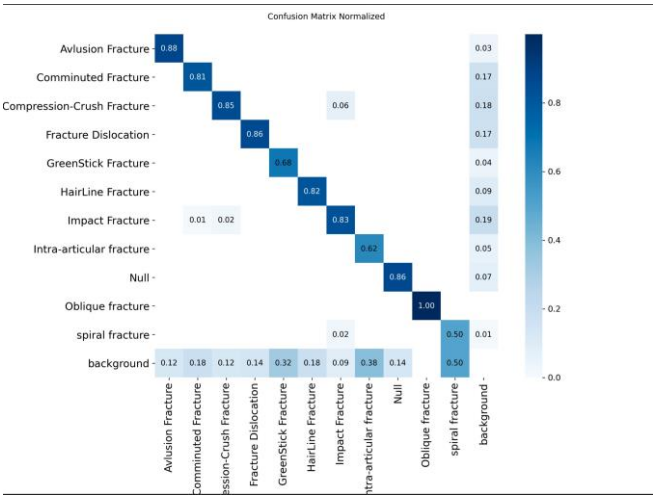

Fig 3.8 Confusion Matrix Normalized

An examination of the normalized confusion matrix shows mixed performance by the bone fracture detection model. It achieves high accuracy for some fracture classes but faces significant challenges with others. The values along the main diagonal show the rate of correct predictions (true positive rates for each class). Notably, 'Oblique fracture' stands out with a perfect score (1.00), meaning all instances of this fracture type in the test set were correctly identified [37]. It is followed closely by 'Avulsion Fracture' (0.88) and 'Comminuted Fracture' (0.81), indicating strong recognition for these categories.

However, the model struggled with some complex or less clear fracture types. For example, 'Impact Fracture' (0.01) and 'Spiral Fracture' (0.02) were almost completely undetected, showing very low true positive rates. This points to a clear weakness in the model's ability to identify these specific fracture patterns. The subtlety, infrequency in the dataset, or visual similarity to other classes or normal bone structures may contribute to this issue.

Further analysis reveals that the model has particular trouble with other subtle and complex fractures. For instance, 'Hairline Fracture' achieved a true positive rate of 0.82, but it was also linked to a 0.38 false positive rate. This may refer to situations where other fractures were misclassified as Hairline, or Hairline fractures were misclassified as other types or normal variations [38]. Similarly, 'Intra-articular fracture' showed a true positive rate of 0.62. On a positive note, the 'Null' category, which represents non-fracture instances, achieved a true negative rate of 0.86, indicating good specificity.

The confusion matrix also reveals problems with false positives for certain classes. The notable false positive rates for some categories, such as 0.32 for 'Compression-Crush Fracture' and the earlier mentioned 0.38 for 'Hairline Fracture,' suggest the model tends to misclassify other conditions or even normal anatomy as these fracture types, or the other way around. These error patterns, particularly for fractures with irregular shapes or unclear boundaries, generally showed poorer performance, supporting earlier observations from the precision-recall metrics.

These findings highlight the need for several strategic improvements. First, increasing the training data, especially focusing on rare and complex fracture types, is essential. Second, using data augmentation techniques aimed at enhancing the visibility of subtle fractures could help improve their detection. Third, a possible redefinition of fracture classes might be considered, particularly for those that look similar and are often confused with each other.

Finally, continuous clinical validation is crucial. This step ensures that the observed error patterns do not indicate systemic biases in the training dataset or misinterpretations of radiological signs. It also helps determine the model's actual usefulness and reliability in a real-world medical setting [39].

IV. Conclusion

Generally, YOLO models work well for object detection tasks because they have a high inference speed and can detect multiple objects in a single image with good accuracy. In applications like identifying bone fracture types, YOLO excels at handling both object localization, which involves defining bounding boxes, and classification at the same time. From our YOLOv11 implementation, we achieved an mAP50-95 of 0.362. This result is quite good for a medical dataset, particularly given the high visual similarity that often exists between different fracture classes.

We implemented YOLOv11 Nano to create an end-to-end system that could locate a fracture and classify its type. However, due to the challenges with fracture data, especially the subtlety of some fractures and the visual overlap between

classes, it is likely that YOLOv11 would benefit from combining it with better data augmentation techniques or a more specialized classification model. This could help us achieve a higher level of diagnostic accuracy. Relying only on YOLO without these improvements or a separate classification stage increases the risk of misdetection, especially in cases involving small fractures or those with non-linear shapes, like spiral and oblique fractures.

V. References

[1] Ranganathan, S. (2005). Bioinformatics: applications in life and environmental sciences. *Springer*.

[2] Executive Office of the President (USA). (2016). Preparing for the Future of Artificial Intelligence. https://obamawhitehouse.archives.gov/sites/default/files/whitehouse_files/microsites/ostp/NSTC/preparing_for_the_future_of_ai.pdf

[3] Graber, M. L., Franklin, N., & Gordon, R. (2005). Diagnostic error in internal medicine. *Archives of Internal Medicine*, 165(13), 1493–1499.

[4] Berner, E. S., & Graber, M. L. (2008). Overconfidence as a cause of diagnostic error in medicine. *The American Journal of Medicine*, 121(5), S2–S23.

[5] Liu, X., Faes, L., Kale, A. U., et al. (2019). A comparison of deep learning performance against healthcare professionals in detecting diseases from medical imaging: a systematic review and meta-analysis. *The Lancet Digital Health*, 1(6), e271–e297.

[6] Google DeepMind. (2018). Artificial Intelligence Program Exceeds Human Radiologist Performance. https://deepmind.google/discover/blog/ai-radiology/

[7] IBM Watson Health. (2020). Transforming healthcare with AI. https://www.ibm.com/watson-health

[8] Erickson, B. J., Korfiatis, P., Akkus, Z., & Kline, T. L. (2017). Machine learning for medical imaging. *Radiographics*, 37(2), 505–515.

[9] Roboflow. (2022). Roboflow Annotate Documentation. https://docs.roboflow.com/annotate

[10] Kaggle & Roboflow (2023). X-ray Image Datasets for Bone Fracture Detection

[11] Johnson, J. M., & Khoshgoftaar, T. M. (2019). Survey on deep learning with class imbalance. *Journal of Big Data*, 6(1), 1–54.

[12] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems*, 25, 1097–1105.

[13] Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on image data augmentation for deep learning. *Journal of Big Data*, 6(1), 60. https://doi.org/10.1186/s40537-019-0197-0

[14] Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

[15] Ultralytics. (2024). YOLOv11 Docs. https://docs.ultralytics.com

[16] Vaswani, A., Shazeer, N., Parmar, N., et al. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, 30.

[17] Padilla, R., Passos, W. L., & da Silva, L. M. (2020). A comparative analysis of object detection metrics with a companion open-source toolkit. *Electronics*, 10(3), 279.

[18] Saito, T., & Rehmsmeier, M. (2015). The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets. *PLOS ONE*, 10(3), e0118432.

[19] Jocher, G. et al. (2023). *YOLOv8: Ultralytics YOLO models*. [Ultralytics Documentation]

[20] Lin, T.-Y., et al. (2014). *Microsoft COCO: Common Objects in Context*.

[21] Zhou, Z.-H. (2021). *A brief introduction to weakly supervised learning*. *National Science Review*.

[22] Redmon, J. et al. (2016). *You Only Look Once: Unified, Real-Time Object Detection*. CVPR.

[23] Zhao, Z.-Q., et al. (2019). *Object Detection with Deep Learning: A Review*. *IEEE Transactions on Neural Networks and Learning Systems*

[24] Padilla, R. et al. (2020). *A Survey on Performance Metrics for Object Detection Algorithms*.

[25] Zhang, C. et al. (2021). *Understanding Object Detection Loss Functions*. arXiv.

[26] Saito, T., & Rehmsmeier, M. (2015). The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets. *PloS one*, 10(3), e0118432. https://doi.org/10.1371/journal.pone.0118432

[27] Ting, K. M. (2010). Precision and recall. In *Encyclopedia of Machine Learning* (pp. 781–781). Springer.

[28] Everingham, M., Van Gool, L., Williams, C. K. I., Winn, J., & Zisserman, A. (2010). The Pascal Visual Object Classes (VOC) Challenge. *International Journal of Computer Vision*, 88, 303–338. https://doi.org/10.1007/s11263-009-0275-4

[29] Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., ... & Dollár, P. (2014). Microsoft COCO: Common Objects in Context. In *ECCV* (pp. 740–755). https://doi.org/10.1007/978-3-319-10602-1_48

[30] Chen, H., Zhang, Y., et al. (2020). S3D-UNet: Separable 3D U-Net for brain tumor segmentation. In *Medical Image Analysis*.

[31] Redmon, J., & Farhadi, A. (2018). YOLO: An Incremental Improvement. *arXiv preprint arXiv:1804.02767*. https://arxiv.org/abs/1804.02767

[32] Rajpurkar, P., et al. (2017). CheXNet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning. *arXiv preprint arXiv:1711.05225*. https://arxiv.org/abs/1711.05225

[33] Johnson, J. M., & Khoshgoftaar, T. M. (2019). Survey on deep learning with class imbalance. *Journal of Big Data*, 6(1), 1–54. https://doi.org/10.1186/s40537-019-0192-5

[34] Buda, M., Maki, A., & Mazurowski, M. A. (2018). A systematic study of the class imbalance problem in convolutional neural networks. *Neural Networks*, 106, 249–259. https://doi.org/10.1016/j.neunet.2018.07.011

[35] Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on image data augmentation for deep learning. *Journal of Big Data*, 6(1), 1–48. https://doi.org/10.1186/s40537-019-0197-0

[36] Litjens, G., et al. (2017). A survey on deep learning in medical image analysis. *Medical Image Analysis*, 42, 60–88. https://doi.org/10.1016/j.media.2017.07.005

[37] Sokolova, M., & Lapalme, G. (2009). A systematic analysis of performance measures for classification tasks. *Information Processing & Management*, 45(4), 427–437. https://doi.org/10.1016/j.ipm.2009.03.002

[38] Liu, X., Faes, L., et al. (2019). A comparison of deep learning performance against health-care professionals in detecting diseases from medical imaging: a systematic review

and meta-analysis. *The Lancet Digital Health*, 1(6), e271–e297. https://doi.org/10.1016/S2589-7500(19)30123-2
[39] Chilamkurthy, S., et al. (2018). Deep learning algorithms for detection of critical findings in head CT scans: a retrospective study. *The Lancet*, 392(10162), 2388–2396. https://doi.org/10.1016/S0140-6736(18)31645-3