**Department of**

**Computer Science & Technology**

**Mini Project Report**

**on**

**Innovation in Medicine: ML model to detect fracture in X-ray images**

**Artificial Intelligence & Machine Learning**

**(22CT3503)**

By

**D GNANA PRASOONA -ENG22CT0028 YAMUNA V -ENG22CT0052**

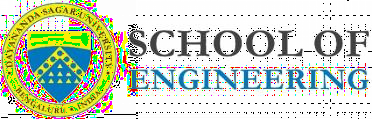
**PUSHYAMI B -ENG22CT0045 ZEENATHUNNISA -ENG22CT0023 VISHISHTA H E -ENG22CT0022**

**Under the supervision of**

**Dr. Danthuluri Sudha Associate Professor,**

**Prof Ramandeep Kaur Assistant Professor,**

**Department of Computer Science and Technology**

****

**Department of Computer Science & Technology**

**CERTIFICATE**

This is to certify that the work titled **“Innovation in Medicine: ML model to detect fracture in bone”** is carried out by **D Gnana Prasoona (ENG22CT0028), Yamuna V (ENG22CT0052), Pushyami B (ENG22CT0045), Zeenathunnisa (ENG22CT0023),**

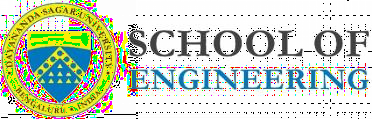
**Vishishta H E (ENG22CT0022)** Bonafide students of Bachelor of Technology in Computer Science and Technology at the School of Engineering, Dayananda Sagar University, Bangalore in partial fulfillment for the award of degree in Bachelor of Technology in Computer Science and Technology, during the year **2024-2025**.

## Dr. Danthuluri Sudha Dr. M Shahina Parveen

Associate Professor, Dept. of CST, Chairperson CST,

School of Engineering, School of Engineering,

Dayananda Sagar University. Dayananda Sagar University.



**Department of Computer Science & Technology DECLARATION**

We, **D Gnana Prasoona (ENG22CT0028), Yamuna V (ENG22CT0052), Pushyami(ENG22CTOO45), Zeenathunnisa (ENG22CT0023), Vishishta H E (ENG22CT0022**), are students of the fifth semester B.Tech in Computer Science and Technology, at School of Engineering, Dayananda Sagar University, hereby declare that the mini project titled “**Innovations in Medicine: ML model to detect fracture in**” has been carried out by us and submitted in partial fulfillment for the award of degree in Bachelor of Technology in Computer Science and Technology during the academic year 2023-2024.

Student Signature

Name1: D Gnana Prasoona USN: ENG22CT0028

Name2: Yamuna V USN: ENG22CT0052

Name3: Pushyami B USN: ENG22CT0045

Name4: Zeenathunnisa USN: ENG22CT0023

Name5: Vishishta H E USN: ENG22CT0022

Place: Bangalore Date:

# ABSTRACT

Many times, it is difficult and time consuming to find out the location of fracture in the patient who is suffering from pain. Today medical imaging technique played the significant role in research and diagnosis field. X-ray imaging technique is used to diagnose and also used to represent anatomical structures such as bones, in human beings. X-ray imaging technique which is further used to detect bone fractures and then the obtained image is processed by different image processing methods such as Computer Aided Diagnosis, Edge Detection, segmentation which are beneficial for technicians.

To help doctors and hospitals better care for their patients, a lot of innovative technical resources have been developed, because the typical scanner for X-ray produces a fuzzy picture of the bone component in issue, surgeons risk making an inaccurate diagnosis of bone fracture when they utilize it. Various stages such as pre-processing, edge detection, feature extraction and machine learning classifications, constitute the backbone of this system.

several machine learning algorithms, such as Naïve Bayes, Decision Tree, Nearest Neighbors, Random Forest, and SVM, have been used specifically for handling bone fracture detection on a dataset with 270 x-ray images. Accuracy measures for the various algorithms employed in the study range from 0.64 to 0.92,

**Keywords:** Medical Imaging, X-ray imaging, Bone fractures, Diagnosis, Image Processing, Computer -Aided Diagnosis (CAD), Edge Detection, Segmentation, Anatomical structures, Medical Technicians**.**

# LIST OF FIGURES

**Figure 1:** Image processing Steps 2

**Figure 2:** System Architecture 8

**Figure 3:** Implementation ………………………………………………...

* 1. : Loading images and labels 10
  2. : Visualizing data 11
  3. : Resizing the training images 11
  4. : Resizing the test and validation images 12
  5. : Fix label format 13
  6. : Resizing train, test and valid labels 14
  7. : Normalize images and labels 15
  8. : Cleaning empty labels 16
  9. : Validate yolo labels 17
  10. : Check bounding boxes 18
  11. : Visualize bounding boxes 19
  12. : Fix invalid classes 20 - 21
  13. : Remove images with invalid labels 22 - 23
  14. : Training the Model 24

**Figure 4:** Results …………………………………………………………...

* 1. **:** Pair Plot 25
  2. : Class Distribution (Bar Chart) and

Bounding Box Characteristics (Scatter Plot) 26

* 1. : Confusion Matrix 27
  2. : Results Graphs 28
  3. : Batch Images 28 - 34

**LIST OF TABLES**

**Table 1:** Dataset distribution for training and testing 34

**Table 2:** Hyperparameters used for training and the model 34

**Table 3:** Comparison table 04

# LIST OF ABBREVATIONS

**AI** - Artificial Intelligence

**CNN**- Convolutional Neural Network

**CT –** Computed Tomography

**DICOM –** Digital Imaging and Communications in Medicine

**ML –** Machine Learning

**YOLO –** You Only Look Once

**TABLE OF CONTENTS**

|  |  |
| --- | --- |
| **Page No.** | |
| **Certificate** | **i** |
| **Declaration** | **ii** |
| **Abstract** | **iii** |
| **List of Figures** | **iv** |
| **List of Tables** | **v** |
| **List of Abbreviations** | **v** |
| 1. **Introduction**    1. **Basic Information.**    2. **Steps involved in the model working.** | **01 - 02** |
| **2. Literature Survey** | **03 - 04** |
| **3. Project Requirement Specification** | **05 - 06** |
| 1. **Problem Definition**    1. **Problem Statement**    2. **Relevance of the Problem** | **07 - 08** |
| **5. System Architecture** | **09** |
| **6. Implementation** | **10 - 35** |
| **7. Conclusion** | **36** |
| **8. References** | **37 - 38** |

**CHAPTER 1**

# INTRODUCTION

Human body consists of 206 bones with different size, shape and complexity. In humans, lower-leg bone breaks are a commonly occurred. The bone fractured risk are high in aged people due to the weak bones. The small and tiny bone fractures in the bones are difficult to analyze by the doctor. The manual process to find the fracture in the bone is time consuming and also we may have errors. So, it is necessary to develop a computer based system to reduce the time and the errors. We have many machine learning technology which are used in medical imaging as well as in the power electronics fields.

Wilhelm Roentgen discovered X-ray in 1895. X-ray is an electromagnetic radiation. The wavelength of X-ray ranges from 0.01 to 10 Nm and its frequency ranges from 30 Petahertz and 30 Exahertz. And the energy range of X-ray is from 100 Ev and 100 Kev. Fractures can be detected by X-ray.

Image processing is a powerful tool used to detect the image details with high accuracy. The bone images contain noise and edges. So, a suitable preprocessing algorithm is used to remove the noise and edges. Finally, the system is trained with the features and classification is performed by the ML algorithms.

Dimililer and Kamil have used ANN (Artificial Neural Network) to classify fracture bone but he could not classify the bone into the healthy and the fracture. Yang et al have used a contour feature of the X-ray image to detect the fracture bone. The accuracy of the system is 85% which need to be improved. To analyze and locate abnormalities in medical images of the human skeleton Sophisticated algorithm are needed.

Traditional machine learning approaches including pre-processing, feature extraction and classification have been widely applied in previous studies on fracture detection and classification. The first stage is Noise Reduction and image processing. The second stage of the project involves the process extracting unique properties from the images.

And the last step is we classify and test and train the model. The development of a system with computer-assisted automated fracture identification employing image processing techniques this efforts seeks to determine whether there is a broken bone. In this model the X-ray image is given, then by applying filter the noise is removed, then edge detection occurs, after feature extraction is used then classification method is used to classify images to detect fracture and non-fracturing images.

Image aquisition

Preprocessing

Edge detection

Feature Extraction

Classification

## Figure 1: Image processing Steps

**Description:** The above image shows how dataset is loaded and processed through steps to classify the types fractures.

CT/X-ray images are taken from a Kaggle that has fractured images along with normal bone images.

# LITERATURE SURVEY

Deep learning approaches for bone fracture diagnosis that have attracted a lot of study since several models and approaches are investigated to raise the diagnostic accuracy. Using X-ray images it was used in one study to identify the bone fractures that have chosen for its ability to lower computing complexity while preserving the high accuracy, hence obtaining an outstanding 98% of the accuracy [1].

CNN architecture on a large number of annotate X-ray images including various types of the fracture. Using many datasets from different medical institutions, the system’s assessment of accuracy, sensitivity, specificity and F1 score unequivocally indicates CNNS tremendous strength in medical image processing [2]. CNN’s natural ability is to extract complex information from the medical images was used to identify the minute elements in fracture patterns [4].

Transfer of learning and employing pre-trained models such as ResNet and VGG expedited training and improved performance. The data augmentation technique were used to improve the model generalization [2]. At last, a proposal integrating ResNet50 in a machine learning approach aimed to lower reliance on human diagnosis and improve the diagnosis accuracy [5].

RCNN algorithm regularly showed the better performance in the study over several trial environments [3]. This paper that describes the need of choosing the suitable optimization method and network topologies to raise the model performance.

Support Vector Machine (SVM), is one of the best machine learning algorithms, it was created in the 1990s and it is largely employed to pattern recognition [6]. SVM is a type of supervised machine learning in which an SVM training algorithm develops a model that predicts the category of the new example that has given as a set of training samples [7].

One of the simplest machine learning algorithms, K- Nearest Neighbour (KNN) so it is based on the supervised learning approach. The KNN algorithm categorizes new data points based on the similarity and records all available information [8].

|  |  |  |  |
| --- | --- | --- | --- |
| **ALGORITHM** | **ACCURACY** | **ISSUES YET TO BE ADDRESSED** | **UNADDRESSED PROBLEM** |
| **DEEP LEARNING** | Up to 98% (using x-ray images) | Computational complexity reduction while maintaining the accuracy. | Dependence on large annotated datasets. |
| **CNN ARCHITECTURE** | High accuracy, sensitivity, specificity, and F1 score. | Requires extensive  datasets and  computational resources. | Limited generalization across diverse medical scenarios. |
| **Transfer Learning (ResNet, VGG)** | Improved performance and expedited training. | Dependence on pre - trained models from non – medical datasets. | Optimization for medical specific challenges. |
| **ResNet50 INTEGRATION** | Aimed to enhance diagnostic accuracy. | Reducing human reliance and improving automation. | Comprehensive validation across different factures types. |
| **RCNN ALGORITM** | Better performance in trails. | Selectable of suitable optimization methods and network topologies. | Scalability real world medical environments. |
| **SUPPORT VECTOR MACHINE(SVM)** | Effective for pattern recognition. | Requires optimal feature selection. | Limited to linear separability without kernel tricks. |
| **K-NEAREST NEIGHBOUR(KNN)** | Simple and effective for classification. | Computational costs increases with large datasets. | Limited by similarity- based classification. |

## Table 3: Comparison table

**Description:** The above table shows the differences between CNN, ResNet, RCNN, SVM and KNN algorithm . It defines how they differ in our training model. This also shows the issues of Existing model and issues to be resolved on Proposed model.

# PROJECT REQUIREMENT SPECIFICATION

## Data Requirements:

The first step is to gather a lot of X-ray images. These should include images with fractures (like simple, complex or tiny hairline fractures) and images without fractures. To make the system useful for everyone, the data should cover different body parts (like hands, legs and wrists) and include people of all ages and backgrounds.

Radiologists will help by marking exactly where the fractures are and what kind of fractures they are. We’ll clean up the images to make them look consistent by adjusting brightness and size. To make the system smarter, we’ll create more examples by flipping or rotating the images. A separate set of images will be kept aside to check and improve the system as it’s developed.

## Model Requirements:

We’ll start with an existing AI model (like ResNet or EfficientNet) and train it to detect fractures. The system will:

1. Tell whether there’s a fracture or not.
2. Identify what type of fracture it is.
3. Highlight the exact area on the X-ray where the fracture is.

The system should also show how it made its decision so doctors can trust it. For example, it can point out which part of the X-ray it focused on.

## Software and Tools:

We’ll use the Visual Studio (VS code) tool to build the system and the software which has been used is Ultralytics in machine learning. VS code is the powerful tool and also it is lightweight code editor that is ideal for machine learning. These both tools complement each other to streamline the development and implementation of machine learning projects.

## Deployment Requirements:

The system should be simple to use. The doctor or technician should be able to upload an X- ray and get the results quickly. It should work well with hospital systems and give results fast.

## Regulatory and Ethical Considerations:

Because this involves medical data, we’ll ensure patient privacy by following rules like HIPAA or GDPR. This means removing personal details from the X-rays.

The system will also be tested on a wide variety of data to make sure it works well for everyone, no matter their age, gender, or background. Before it’s used in real life, doctors will test it thoroughly to confirm it’s reliable.

# PROBLEM DEFINITION:

**PROBLEM STATEMENT**

Detecting fractures in X-ray images is an important but often it is a difficult task in healthcare. The goal of this project is to create smart system using machine learning that can automatically look at X-ray images and it can figure out if there is a fracture (a broken bone). The system will not only detect if there is a fracture but also point out exactly where it is in the X-ray image. This system is meant to help doctors and medical professionals by acting as an extra set of “eyes”. While doctors are very skilled, sometimes the fractures can be a tricky task for them because of the complexity of X-r=`ay images. This can lead to mistakes or delays in the diagnosis, which could affect the treatment of the patient.

Solutions:

YOLOv8 is a popular algorithm that is used in machine learning, especially for the object detection and image recognition tasks.

YOLOv8 it improves on the previous versions by using more efficient techniques to detect the objects with the higher precision, even in the complex environments.

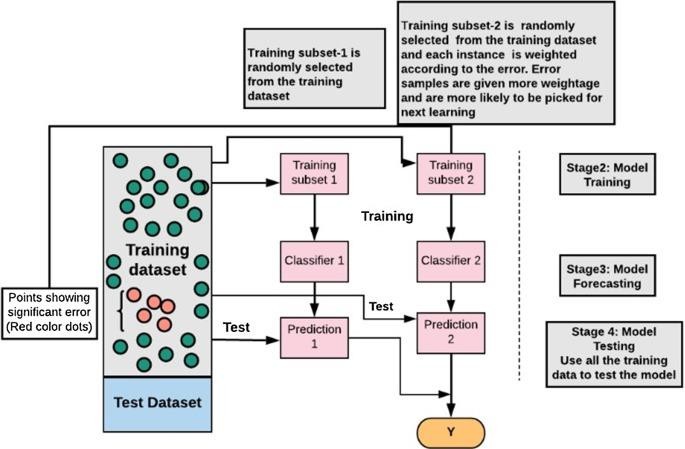
It is very easy to use, it also requires less computational power compared to the other advanced models and it can handle the multiple tasks like object classification and segmentation also it’s in a single framework.

# RELEVANCE OF THE PROBLEM

Detecting fractures through X-ray images is a critical issue in healthcare, with a direct impact on patient outcomes and recovery. Fracture detection is a very big critical aspect of the medical, especially in the emergency and during the orthopedic care (It is treatment for a bone, joint and muscle disorders). Misdiagnosed or undiagnosed fractures can lead to severe complications such as improper healing, chronic pain and long-term disability. The development of a machine learning- based fracture detection system addresses several pressing challenges in healthcare:

1. Reducing Diagnostic Errors: Radiologists and medical professionals often they face a highworkloads, leading to the potential oversight in interpreting X-ray images. A machine learning system can serve as a second opinion, reducing errors and ensuring to accurate diagnosis.
2. Improving the accessibility: In many of the regions, especially in rural and undeserved areas, there is a shortage of skilled radiologists. An automated fracture detection system can bridge this way of gap, enabling the faster more reliable diagnoses.
3. Enhancing efficiency: The manual analysis of X-rays is time-consuming. Automating this process can speed up the diagnosis, allowing patients to receive timely treatment and improving the overall workflow in the medical facilities.
4. Supporting complex diagnosis: Differentiating between the various types of fractures requires expertise. A robust system that identifies the type and location of the fracture with the high sensitivity that can assist doctors in planning precise treatments.
5. Improving Patient Outcomes: Early and accurate fracture detection ensures the proper intervention, reducing the risk of complications, improving recovery times and enhancing the quality of life for patients.

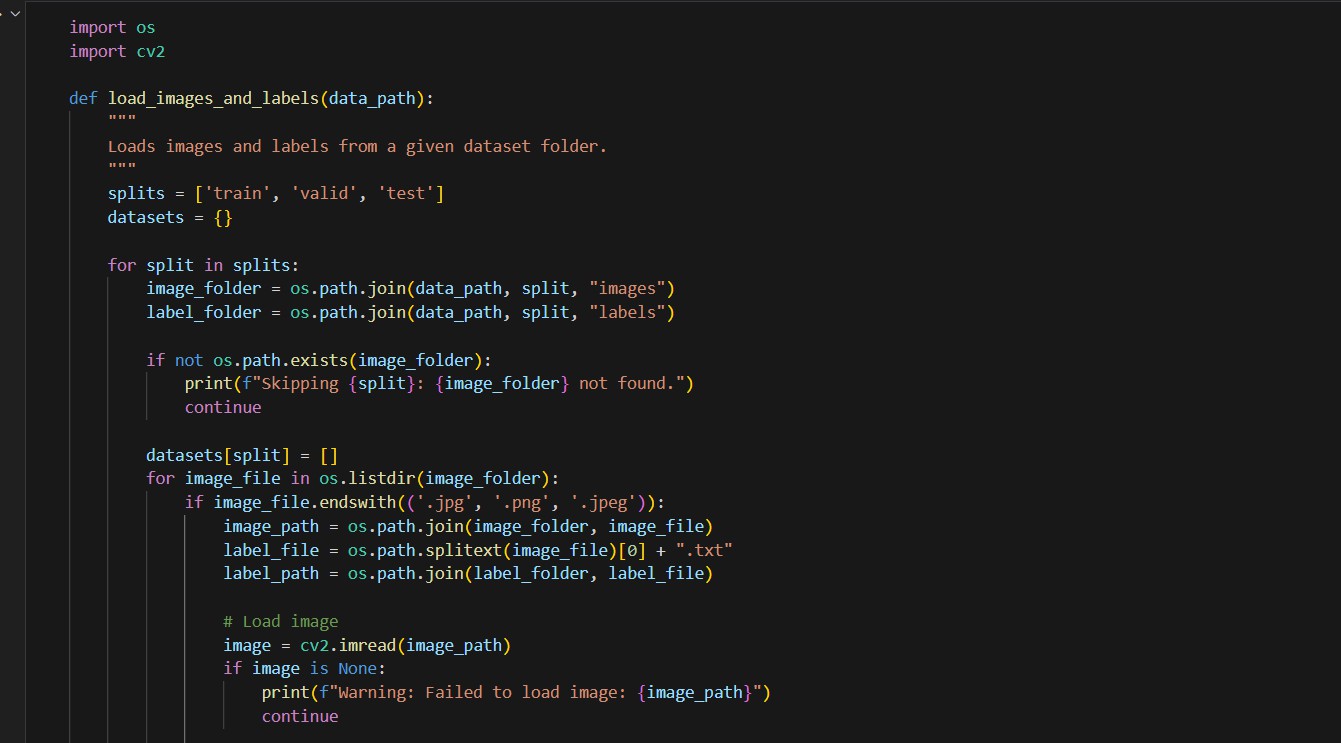
# SYSTEM ARCHITECTURE

****

## Figure 2: System Architecture

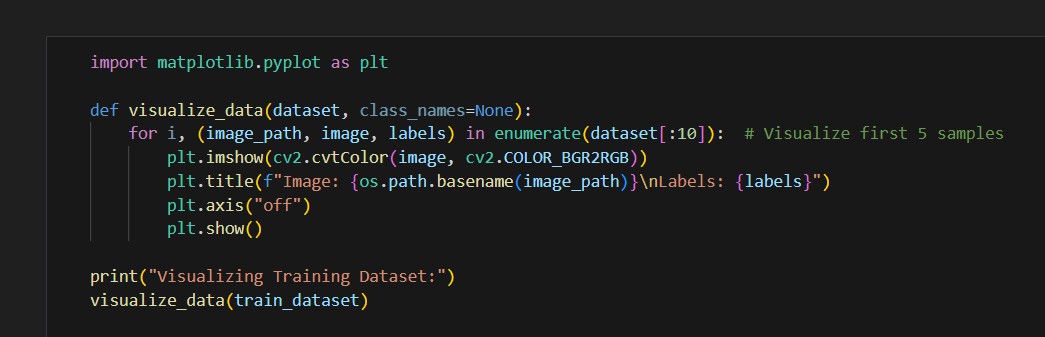
**Description:** The image illustrates a boosting process where a training dataset is split into subsets, with misclassified samples given higher weights in subsequent iterations. Multiple classifiers are trained sequentially, and their predictions are combined to improve overall model accuracy. The process involves stages for training, forecasting, and final testing.

# IMPLEMENTATION

****

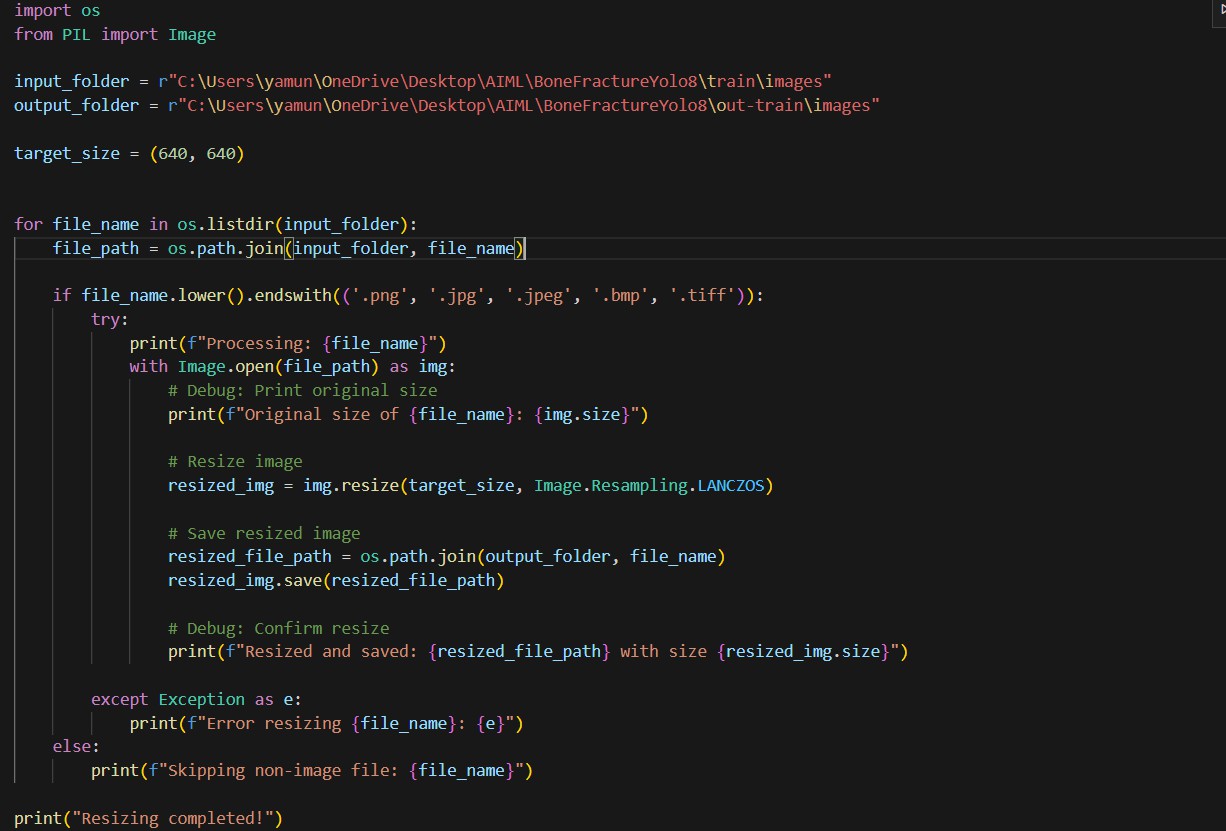
## Figure 3.1: Loading images and labels

**Description:** The above code shows how to load our train, test and valid dataset into our project environment.



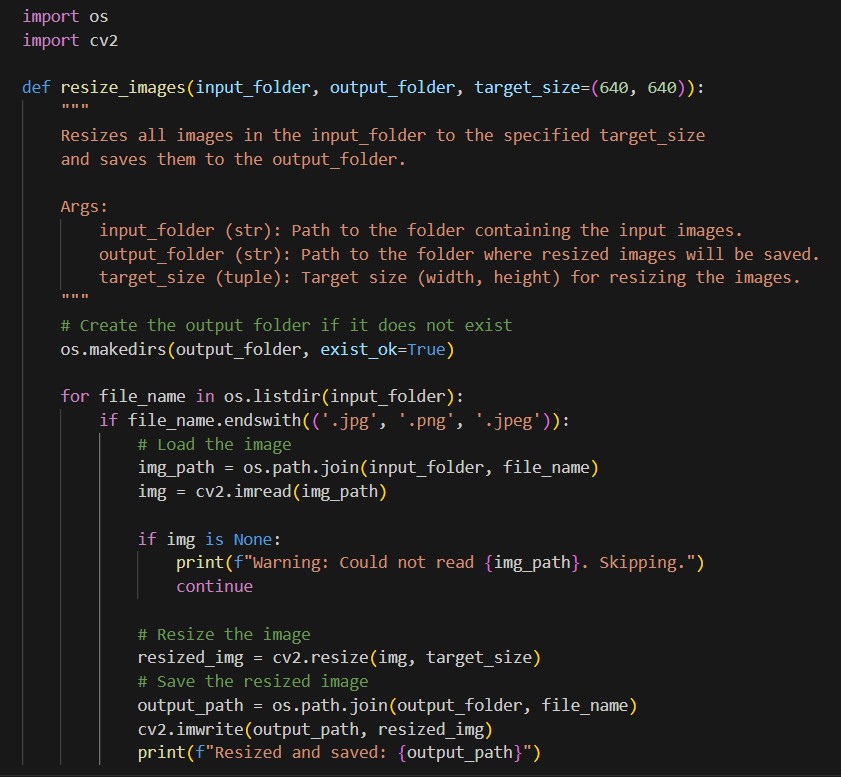
## Figure 3.2: Visualizing data

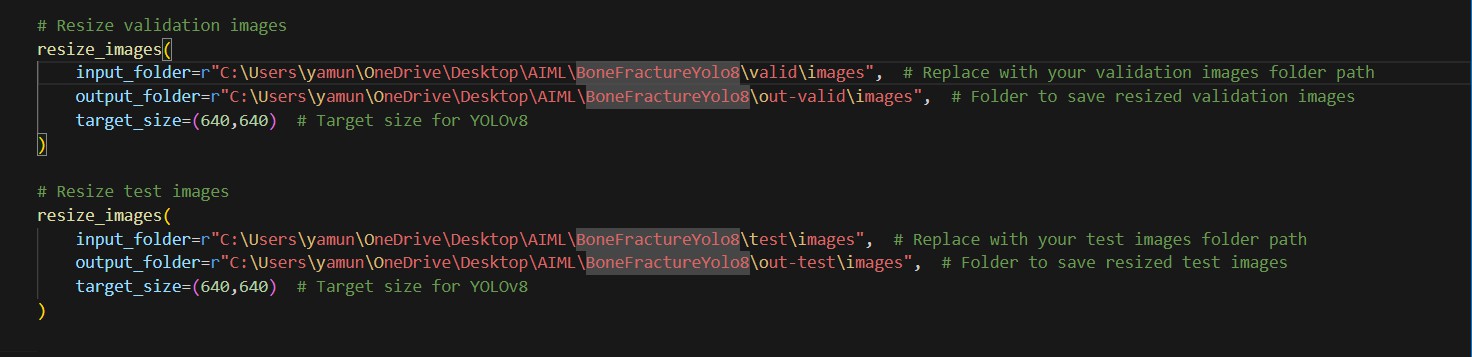
**Description:** The code snippet shows to visualize the images from the dataset and confirms whether dataset is loaded correctly or not.



## Figure 3.3: Resizing the training images

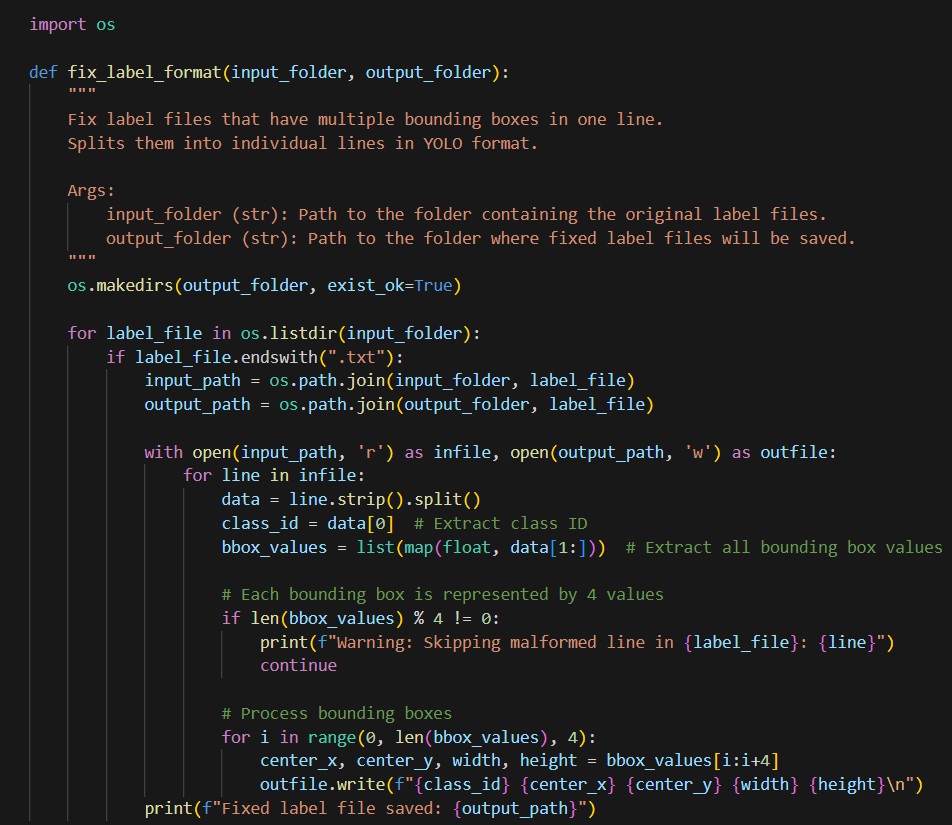
**Description:** This is the step where preprocessing of dataset starts. The above snippet shows first step of preprocessing i.e. resizing of training dataset and stores the preprocessed images in out-train-images folder.

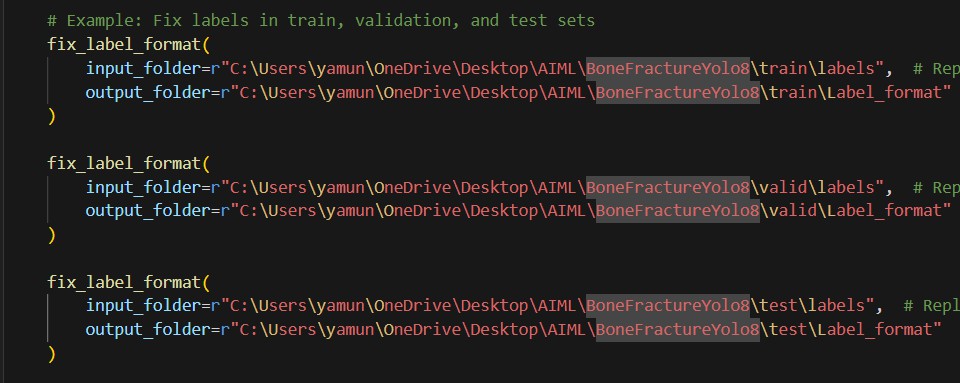




## Figure 3.4: Resizing the test and validation images

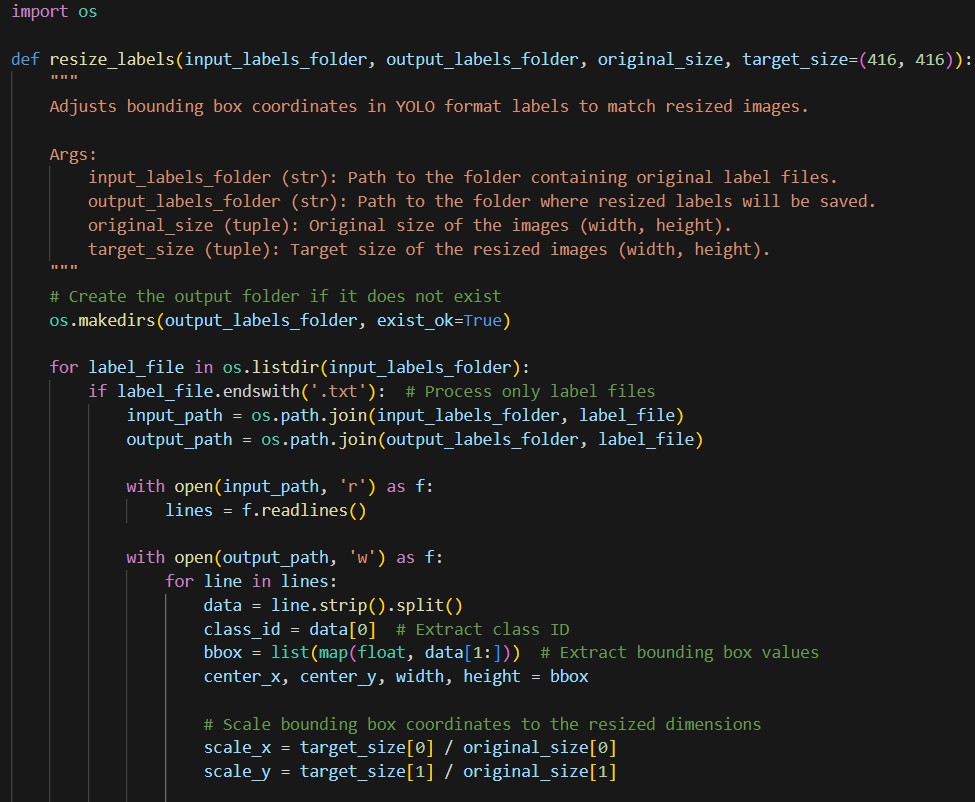
**Description:** This snippet shows resizing of valid and testing datasets and stores them into respective out-valid and out-test folders.

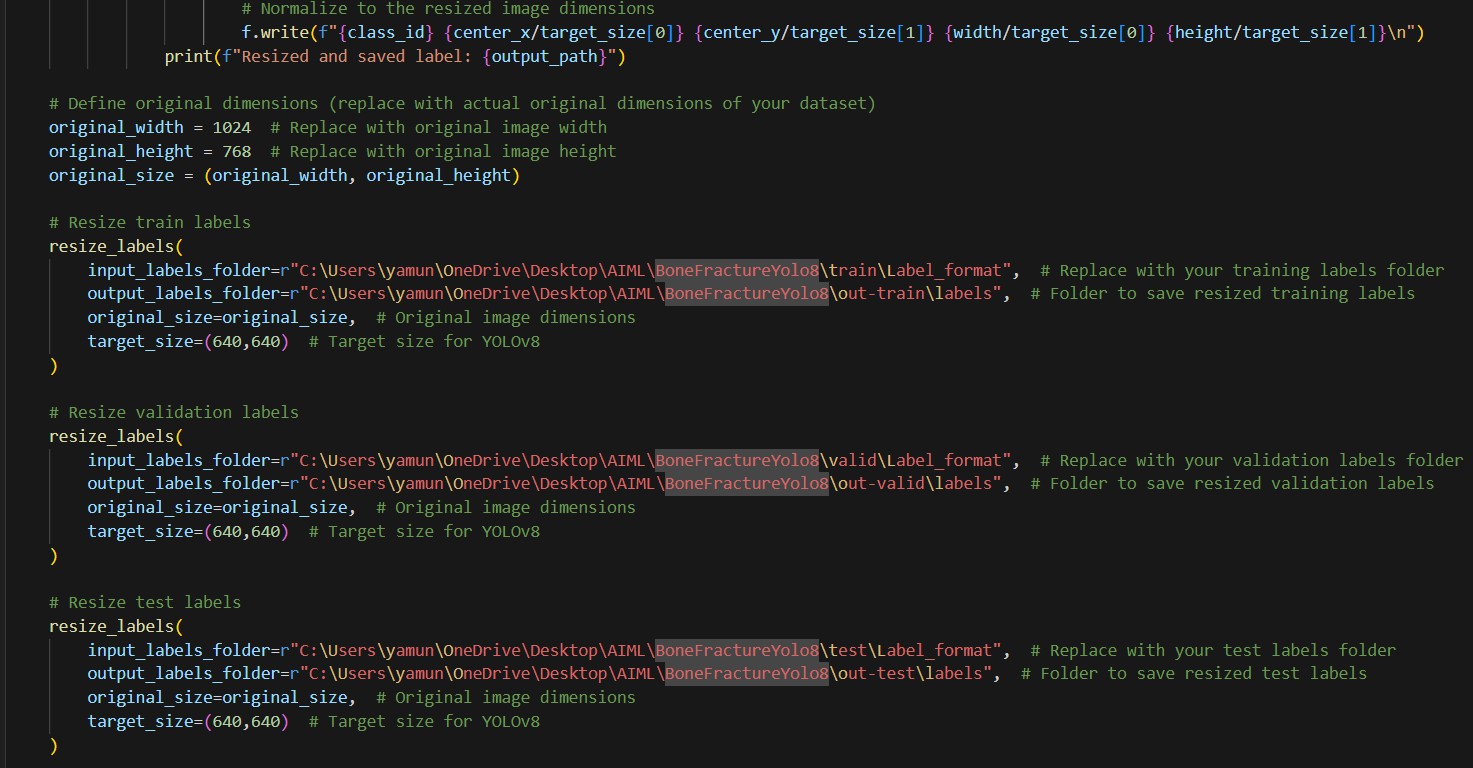




## Figure 3.5: Fix label format

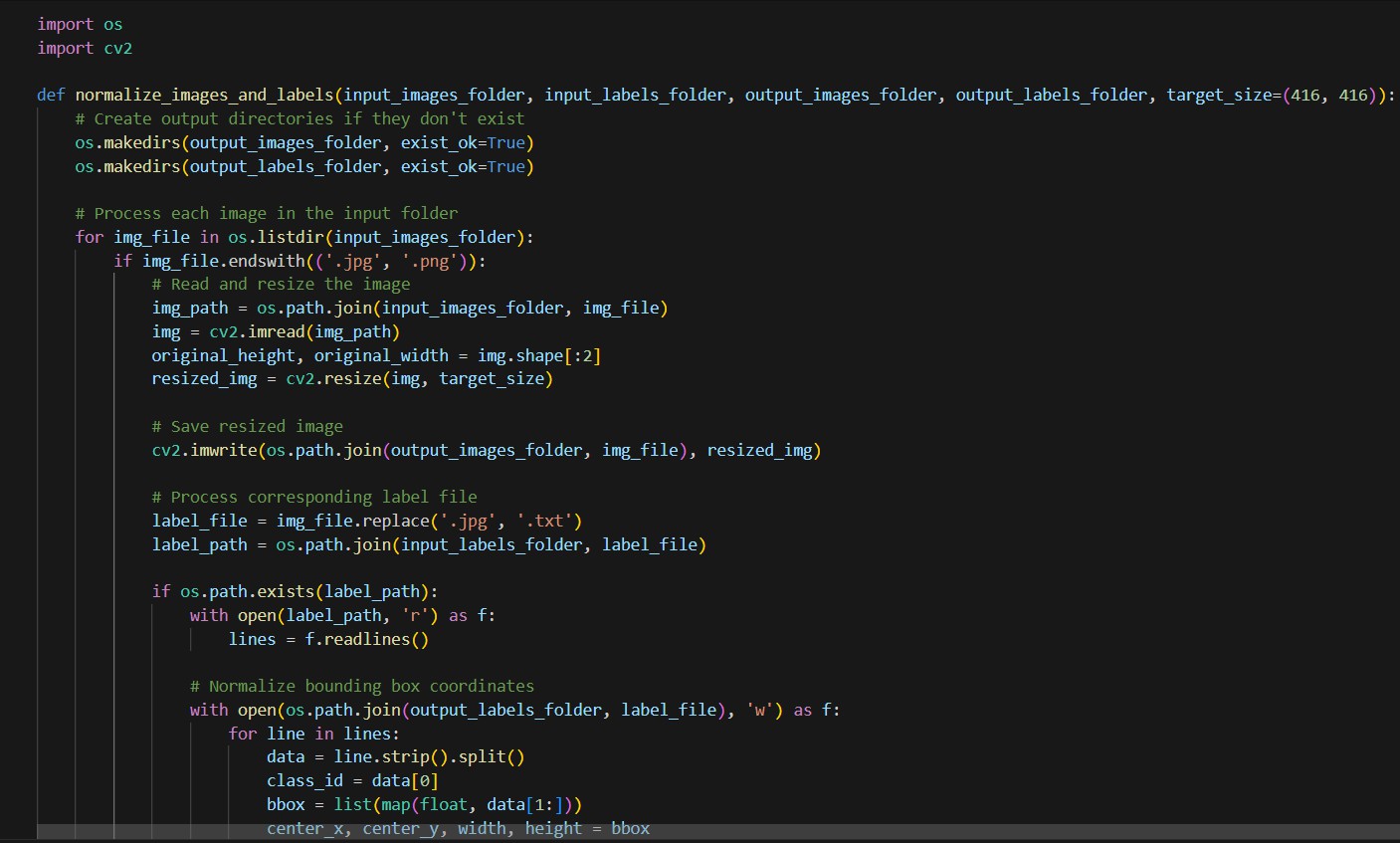
**Description:** Here we have to fix the format of labels where Yolov8 model accepts the format as {class\_id center\_x center\_y width height}.

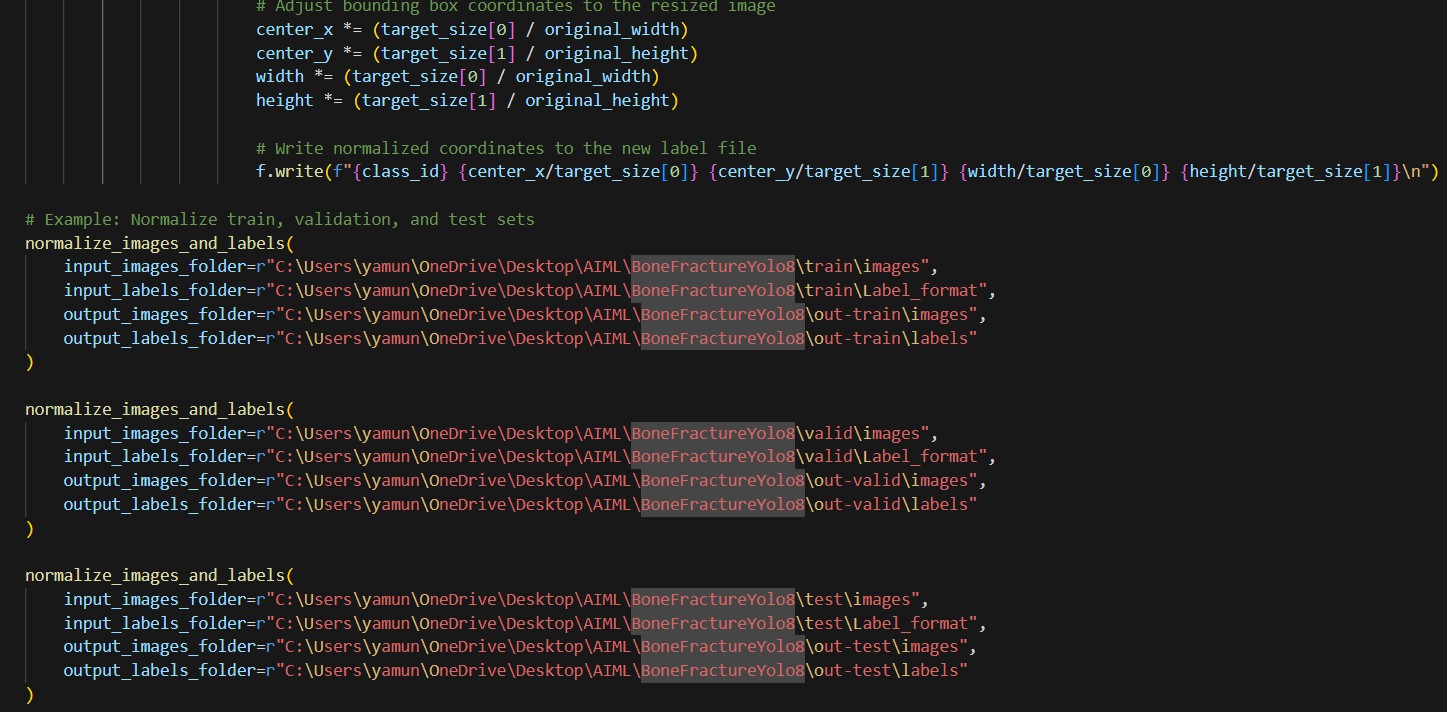




## Figure 3.6: Resizing train, test and valid labels

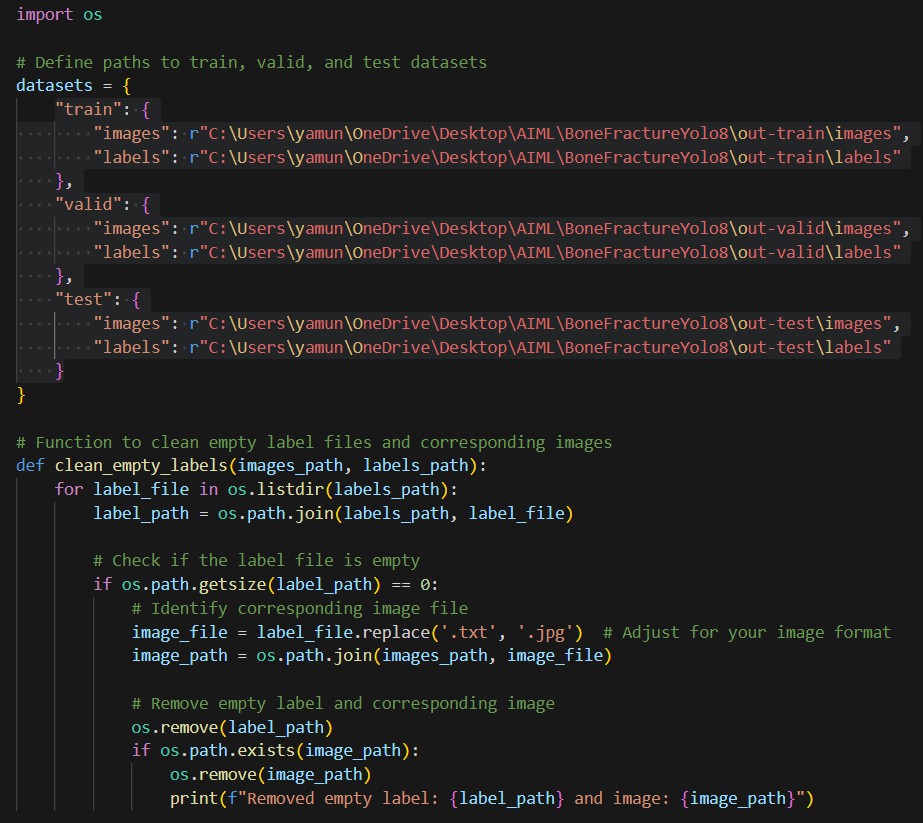
**Description:** The preprocessing should be done for labels for train, valid and test datasets. The above code shows resizing of labels and storing them in the respective out-train-valid-test-Labels folders.

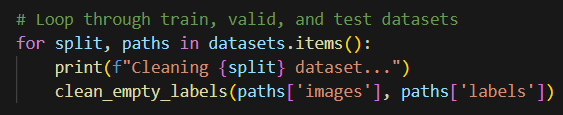




## Figure 3.7: Normalize images and labels

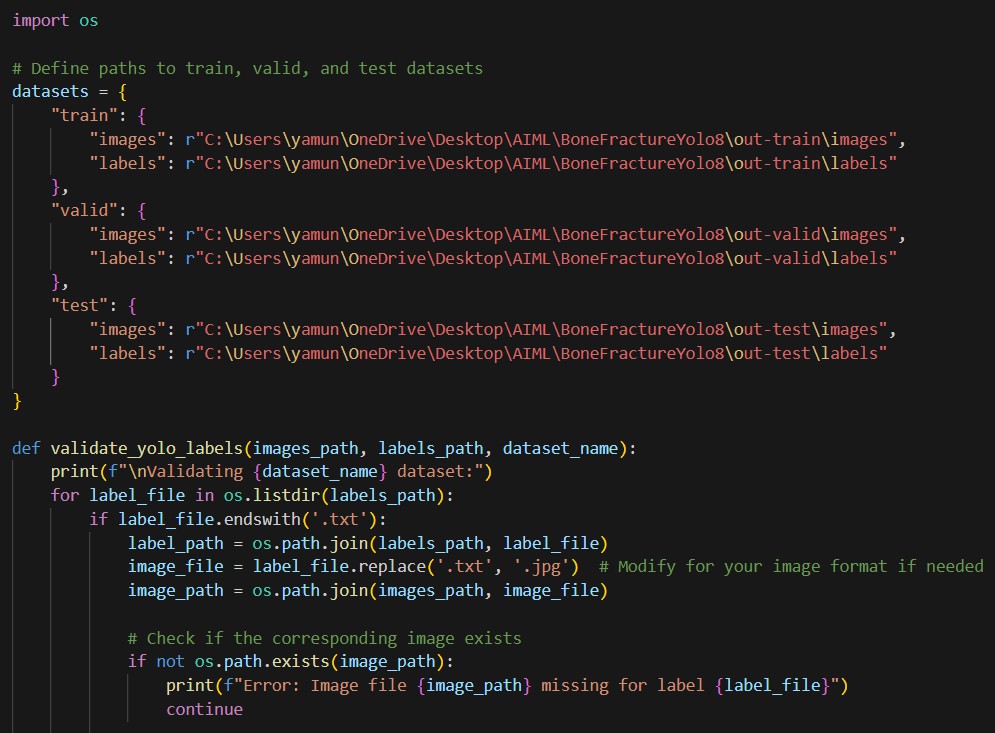
**Description:** The next of preprocessing involves Normalization of images and labels. The above snippet shows the normalizing of both images and labels and stores them into the respective output folders.

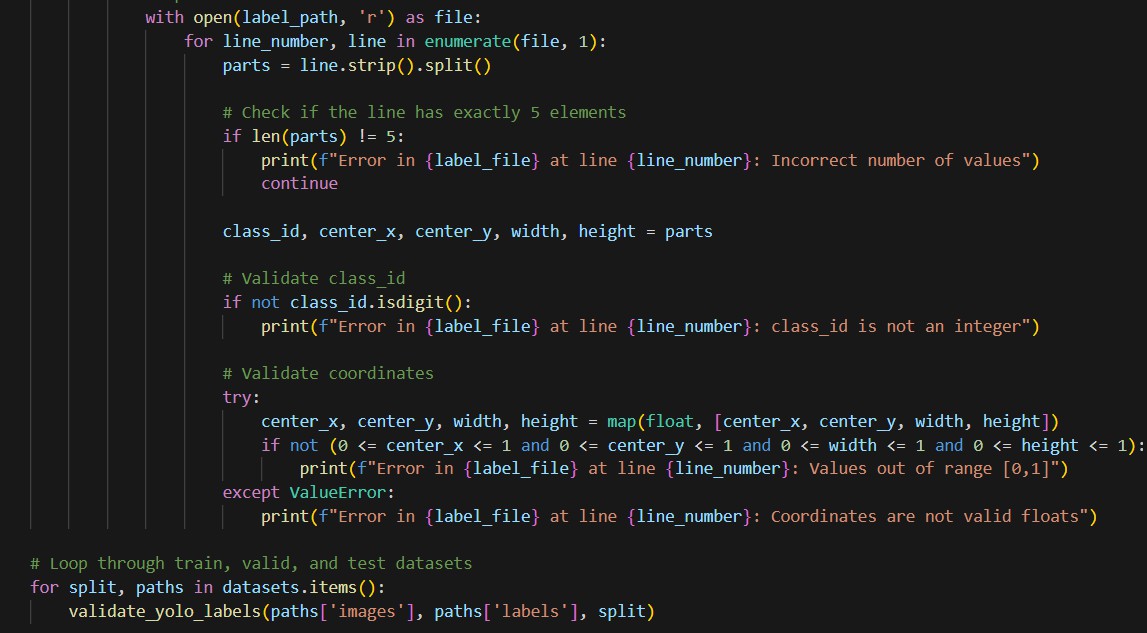




## Figure 3.8: Cleaning empty labels

**Description:** The model requires images only with labels and it can’t process without labels. This is the step to remove empty labels from the dataset.

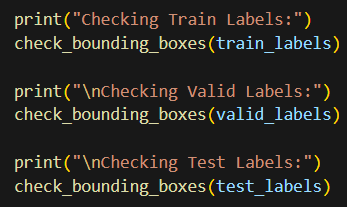




**Figure 3.9**: Validate yolo labels

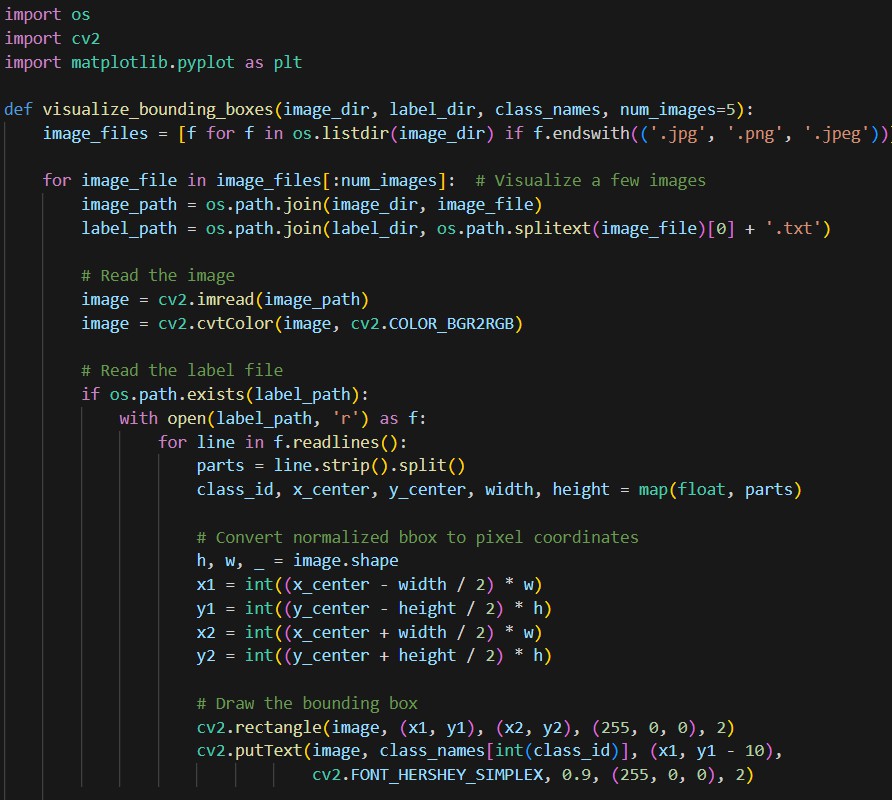
**Description:** The purpose of a label validation script is to ensure that the label files are correctly formatted and consistent with the corresponding images. This is crucial to avoid training errors or incorrect model performance.

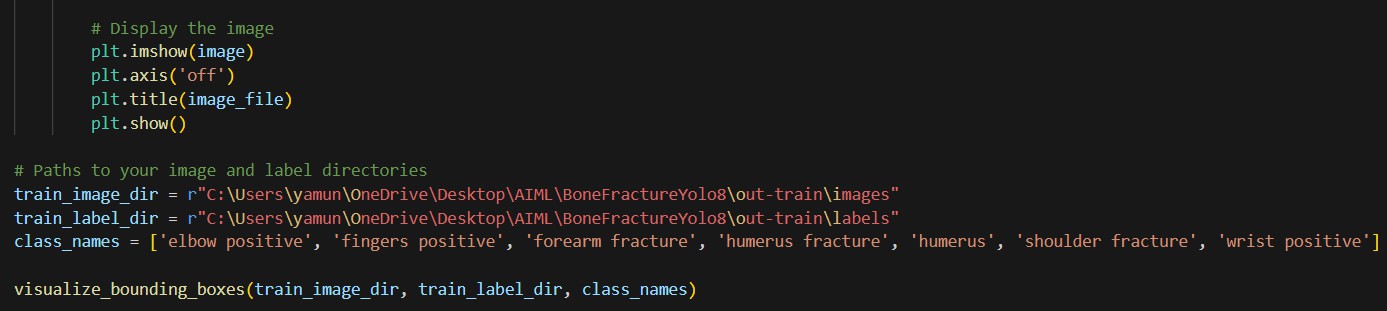




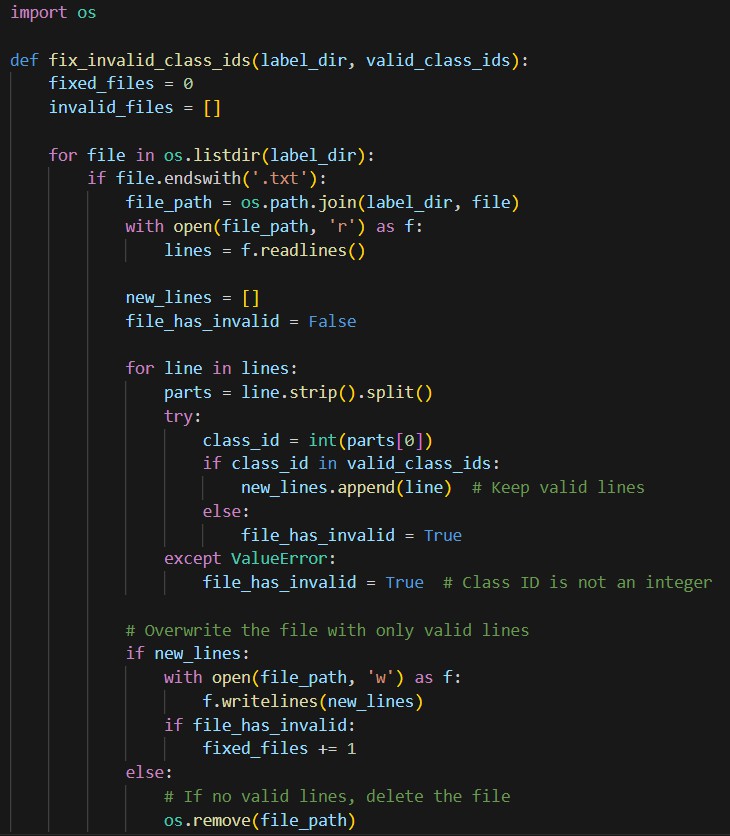
## Figure 3.10: Check bounding boxes

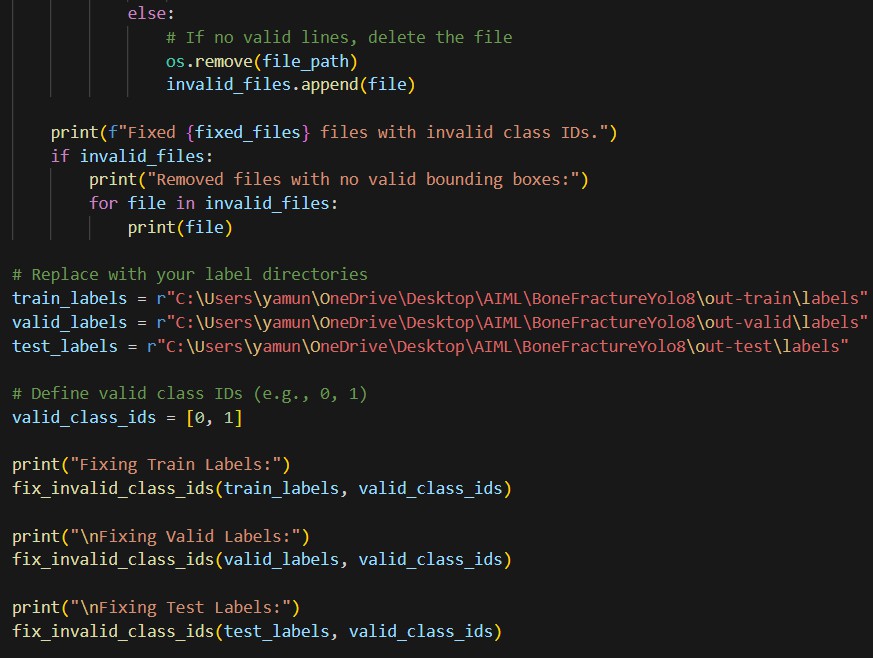
**Description:** A bounding box is a rectangular outline used in computer vision and image processing tasks to define the position and size of an object within an image. Bounding boxes are essential for object detection tasks, as they help models locate and classify objects.





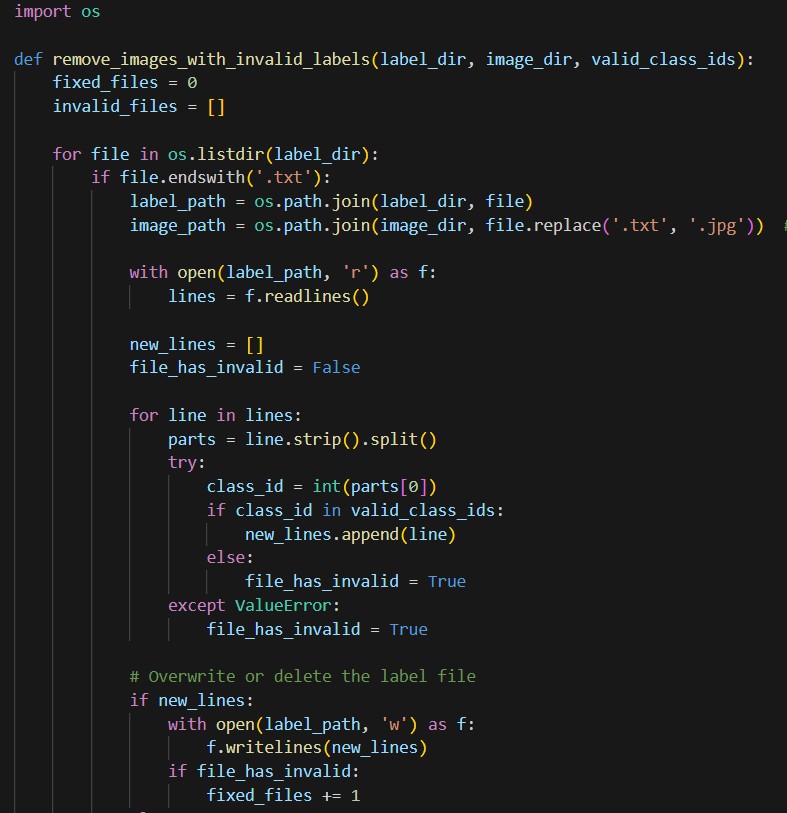
## Figure 3.11: Visualize bounding boxes

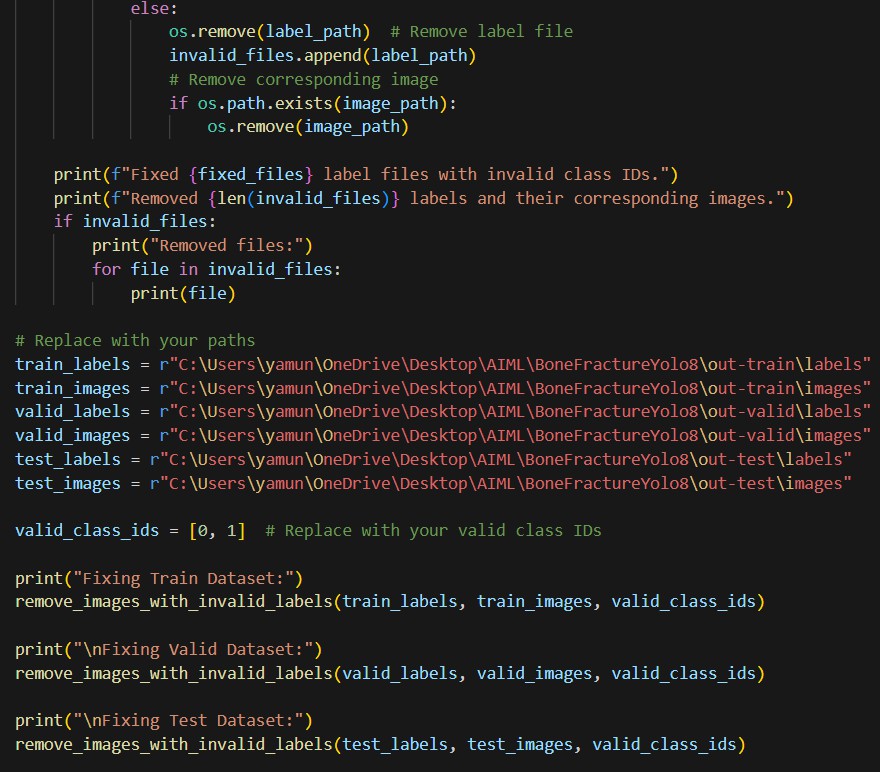




**Figure 3.12: Fix invalid classes**

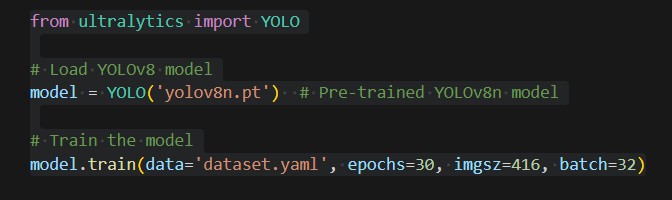
**Description:** Fixing invalid classes in a dataset involves ensuring that the class\_id values in the labels are correct and fall within the range of predefined classes. This is critical for successful training and evaluation in machine learning tasks like object detection.





## Figure 3.13: Remove images with invalid labels

**Description:** Removing invalid labels from a dataset involves identifying and discarding labels that do not conform to the required format or fall outside acceptable bounds. Invalid labels can disrupt the training process and lead to poor model performance, so it is essential to address them systematically.

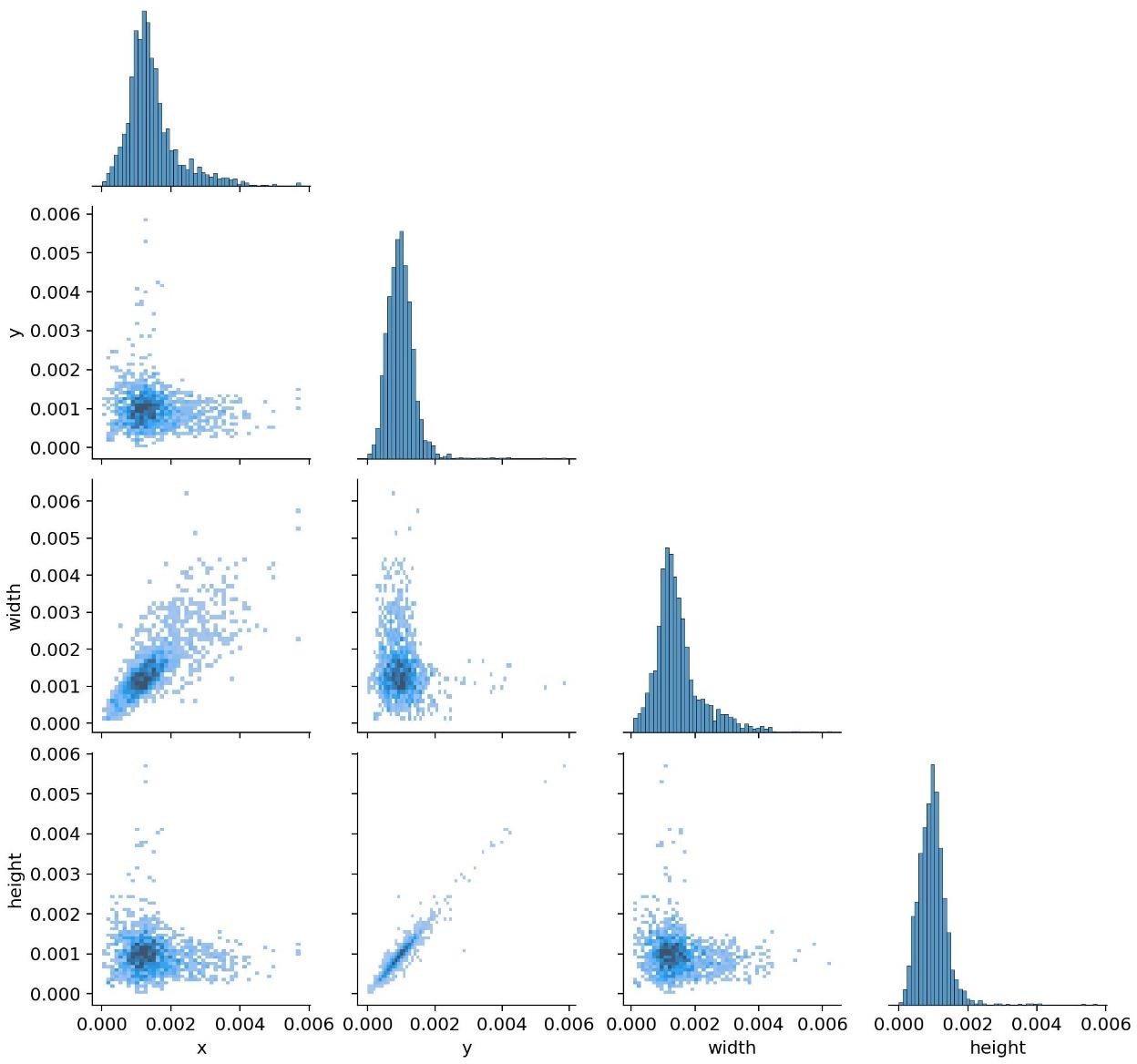


## Figure 3.14: Training the Model

**Description:** Training a YOLOv8 model involves preparing data, setting up the YOLOv8 framework, configuring the model, and executing the training process to detect objects in images. YOLOv8, developed by Ultralytics, is a highly efficient and versatile deep learning model for object detection, segmentation, and classification tasks.

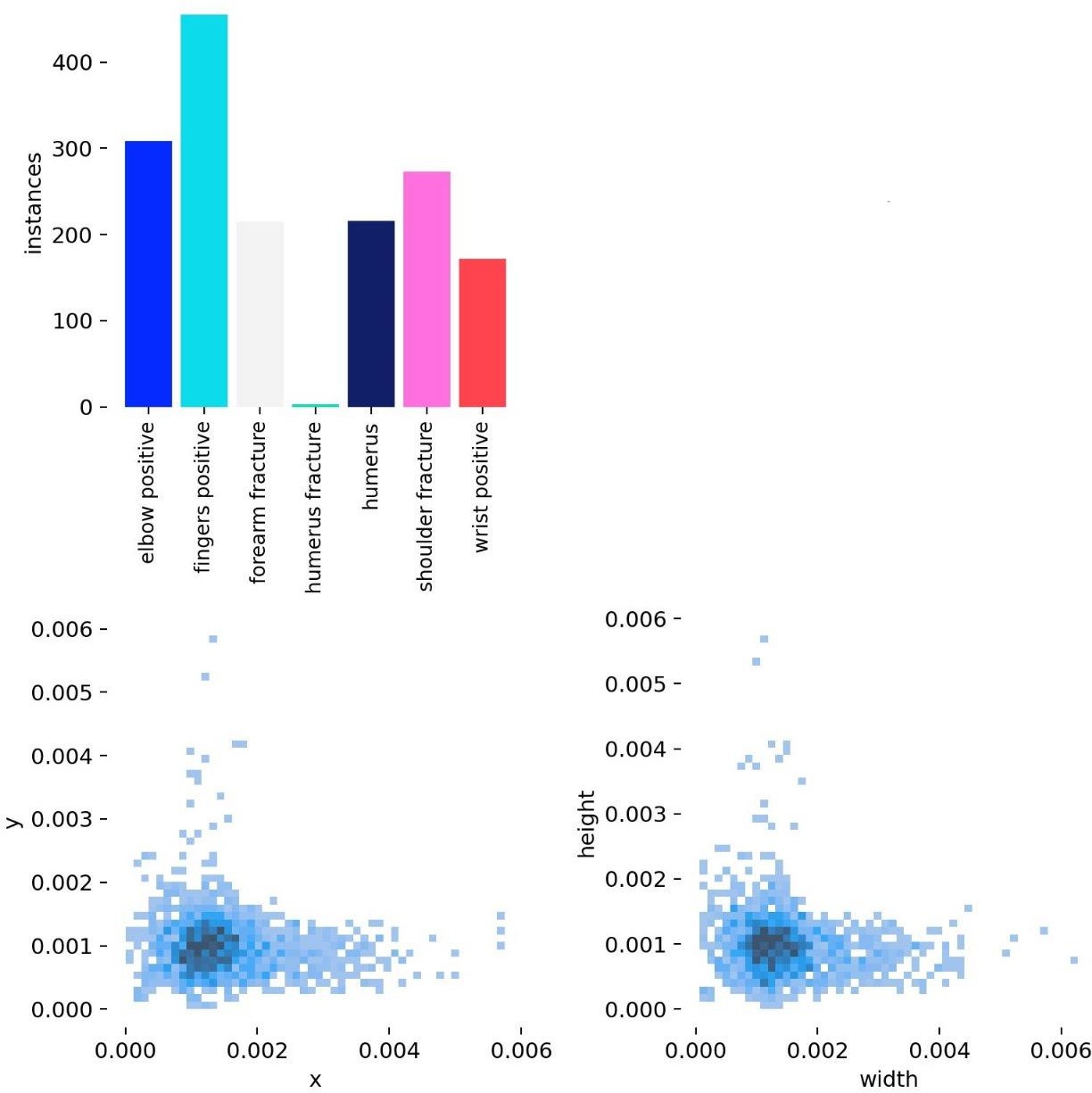
**Innovation in Medicine**: **ML model to detect fracture in X-ray images**

# RESULTS

****

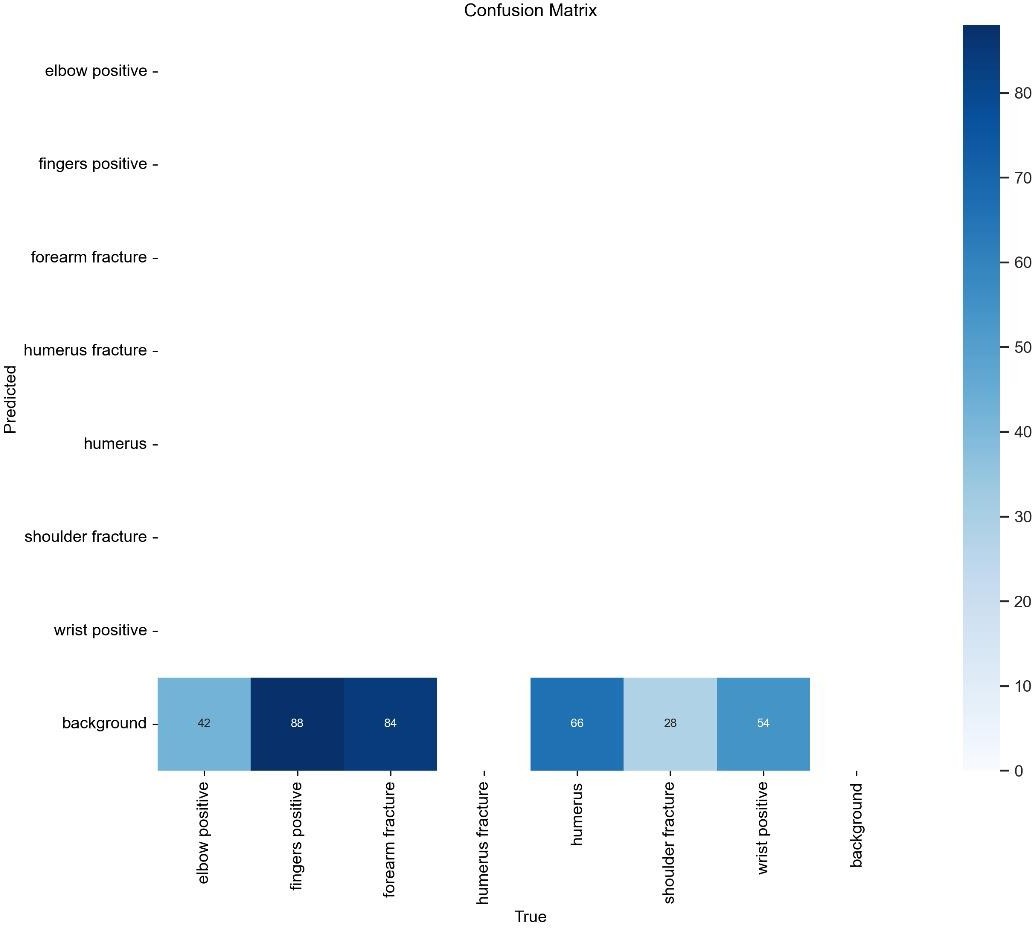
## Figure 4.1: Pair Plot

**Description:** This pair plot visualizes relationships between four variables that are x, y, width, and height. The diagonal shows histograms of individual variables, while the scatter plots display pairwise relationships. Most values are concentrated near zero, with some correlations evident, especially between width and height.



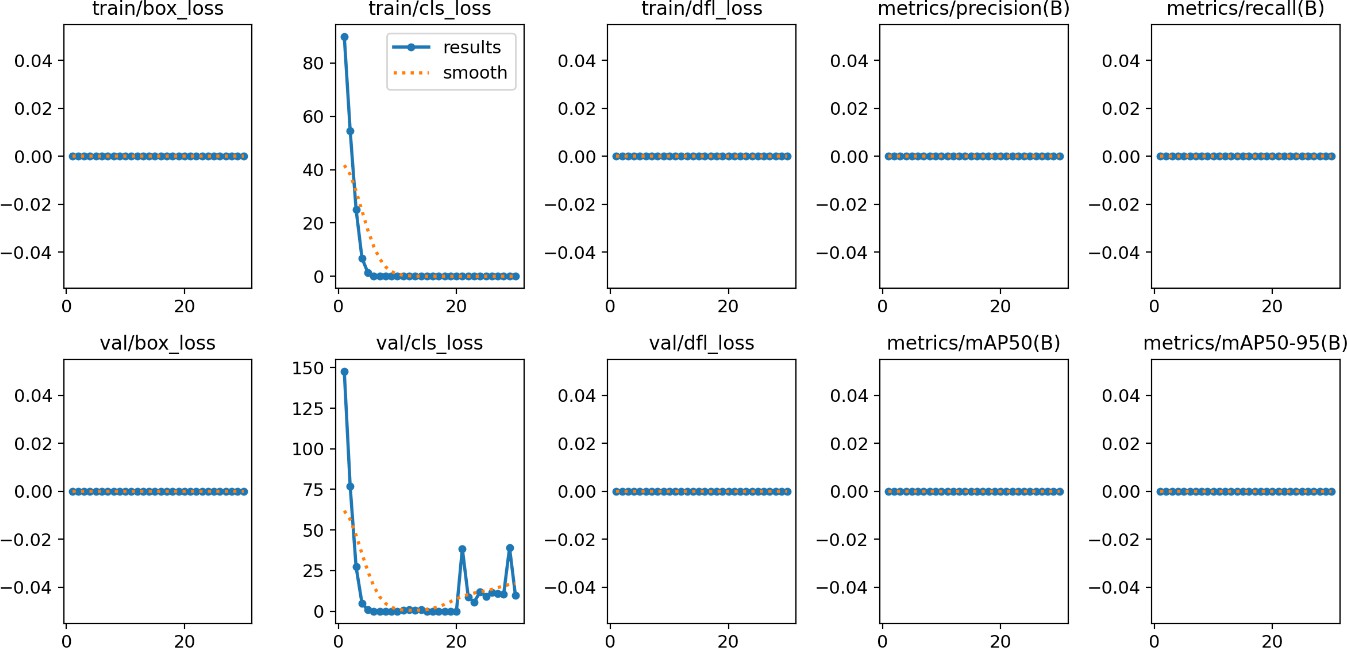
## Figure 4.2: Class Distribution (Bar Chart) and Bounding Box Characteristics (Scatter Plot)

**Description:** This image contains both bar chart and scatter plots. The bar chart at the top shows the count of instances for various classes like "elbow positive," "fingers positive," and others, with "fingers positive" having the highest count. The scatter plots below shows the relationships between x vs. y and width vs. height, showing a dense cluster of values near zero.



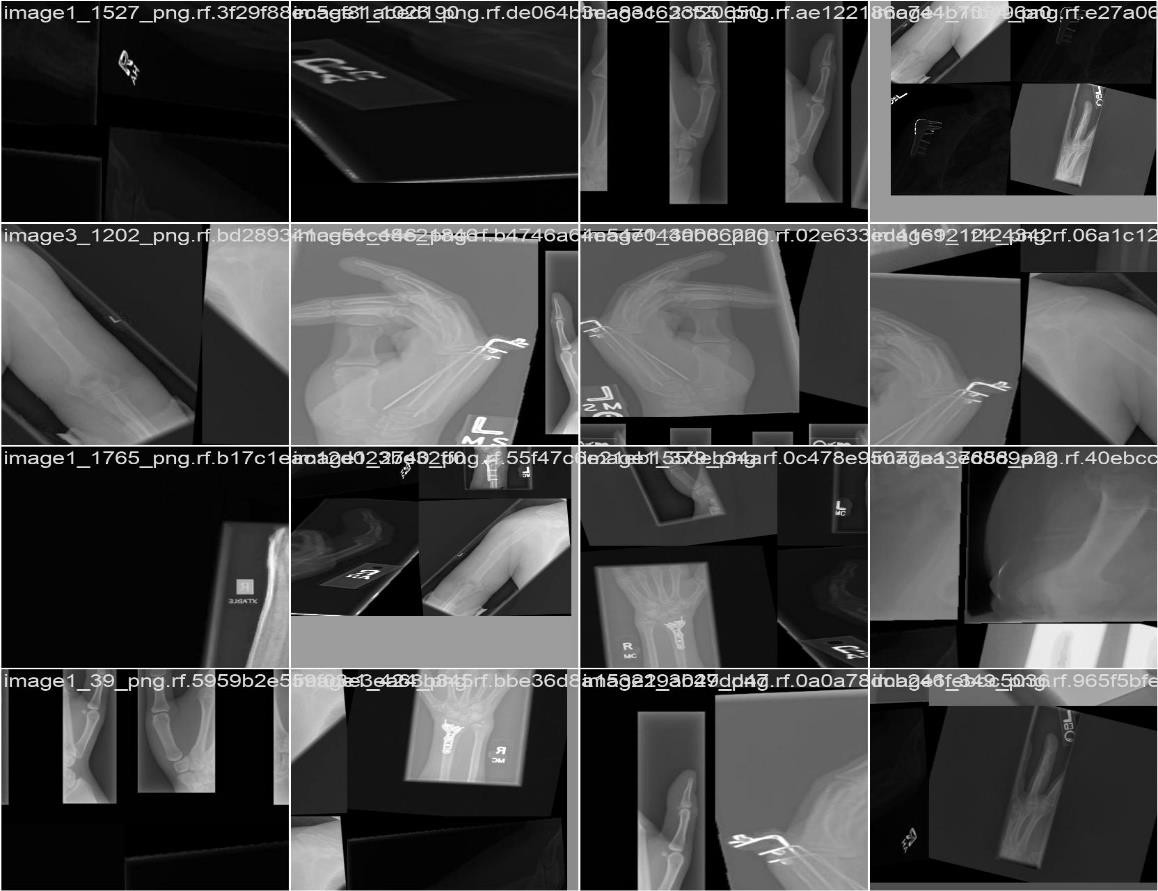
## Figure 4.3: Confusion Matrix

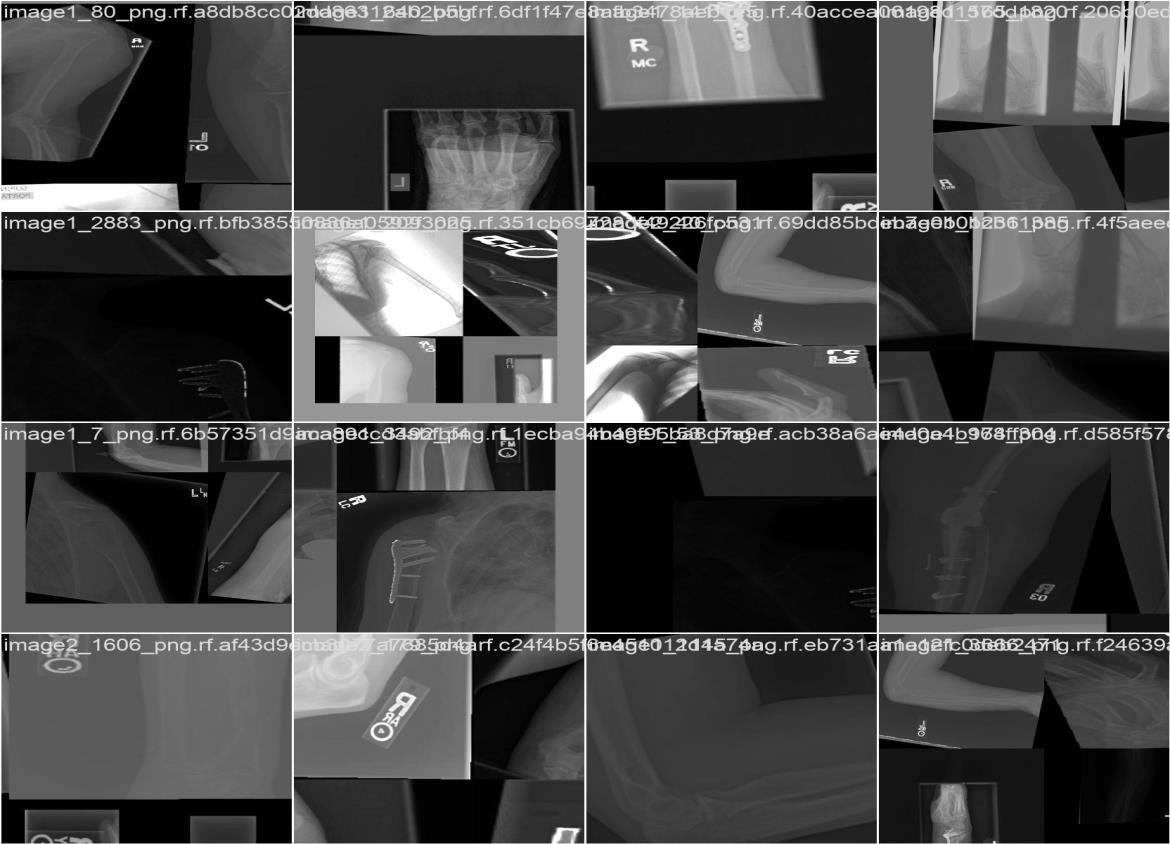
**Description:** This image shows a confusion matrix for a multi-class classification task, illustrating the performance of predictions versus true labels. Each cell indicates the number of instances where a specific class was predicted compared to its actual label, with higher values represented in darker blue shades. The matrix highlights both correct predictions (diagonal entries) and misclassifications (off-diagonal entries).

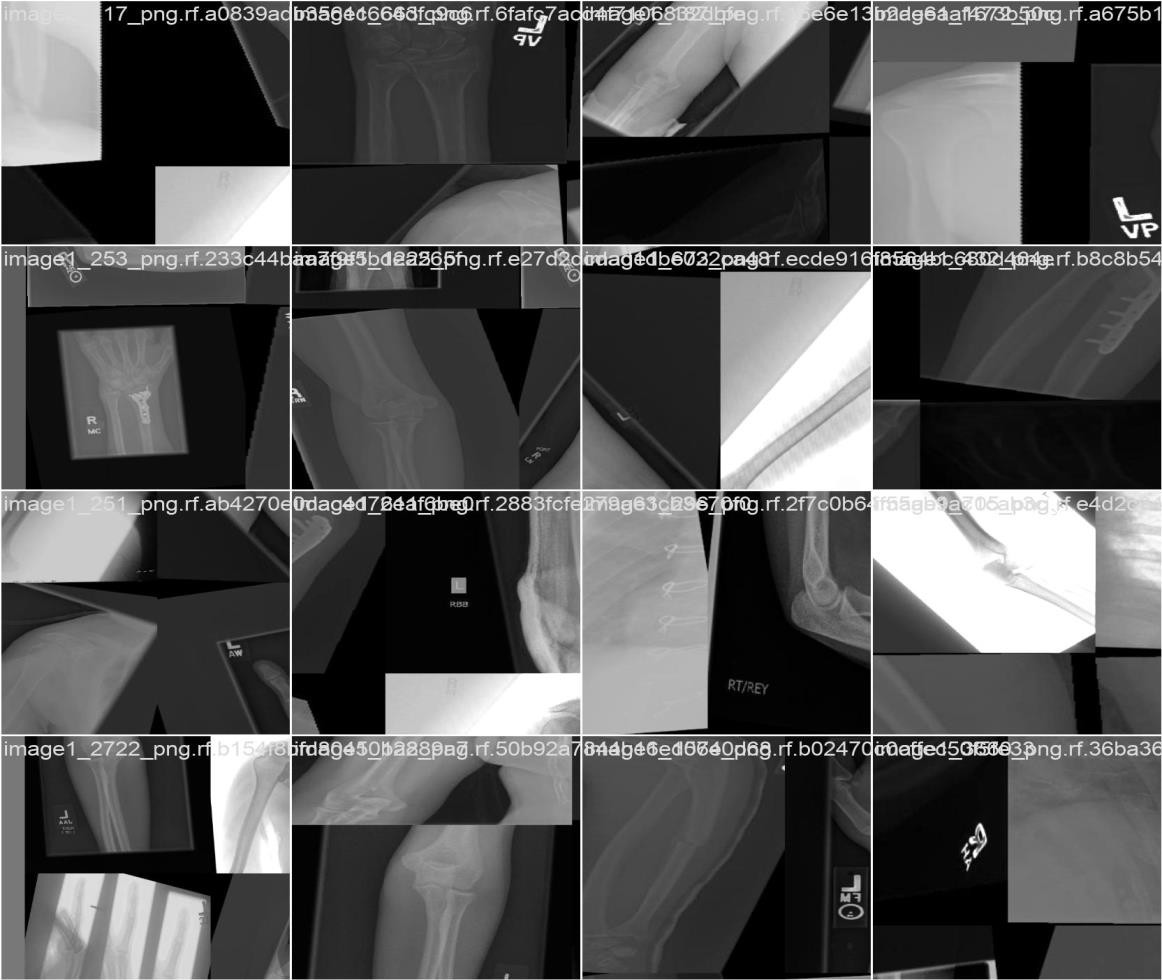


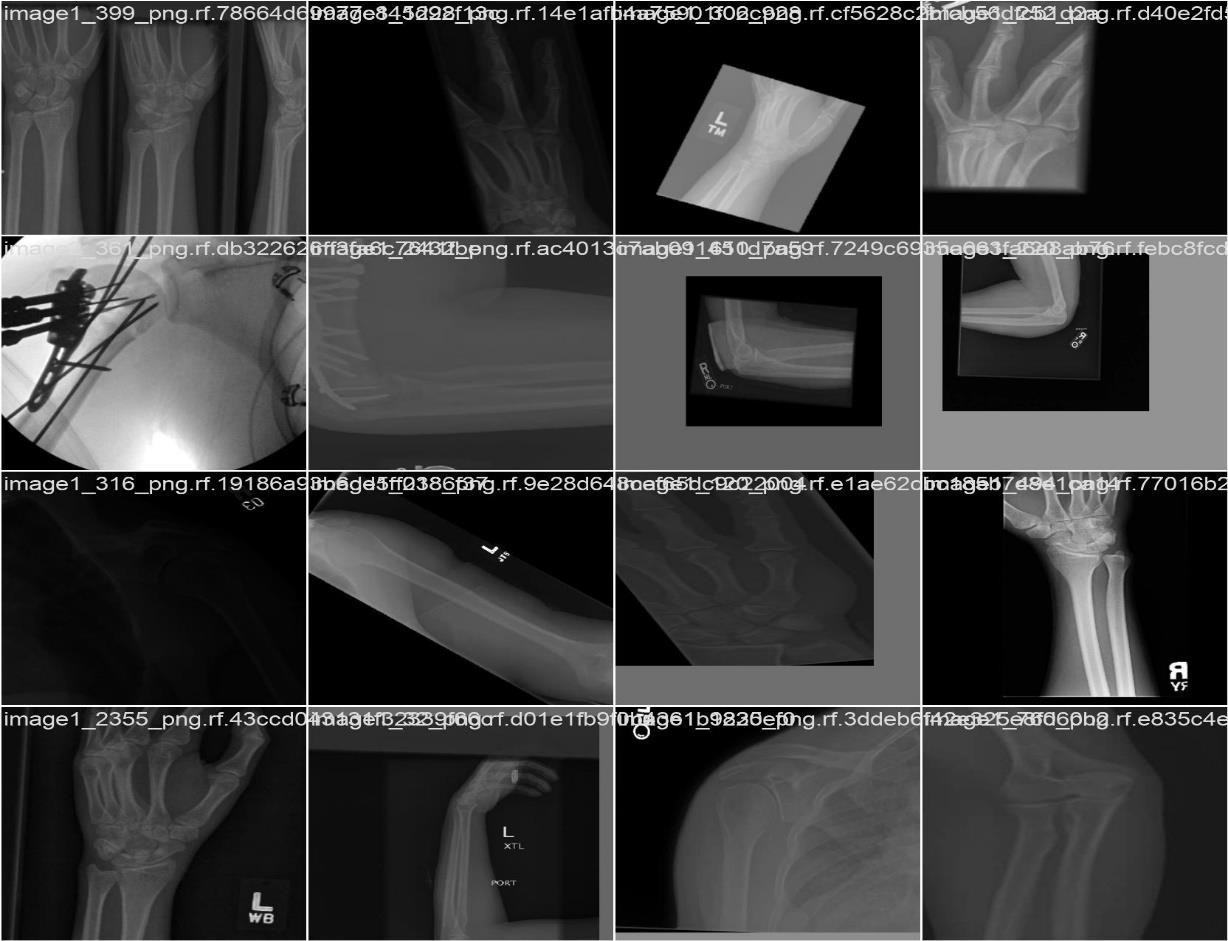
## Figure 4.4: Results Graphs

**Description:** The image shows a grid of training and validation loss and metric plots for a machine learning model. Most metrics, including train/box\_loss, train/dfl\_loss, metrics/precision(B), and metrics/recall(B), remain constant at zero, suggesting no improvement. The train/cls\_loss and val/cls\_loss show a decrease over epochs, indicating some optimization in classification loss.

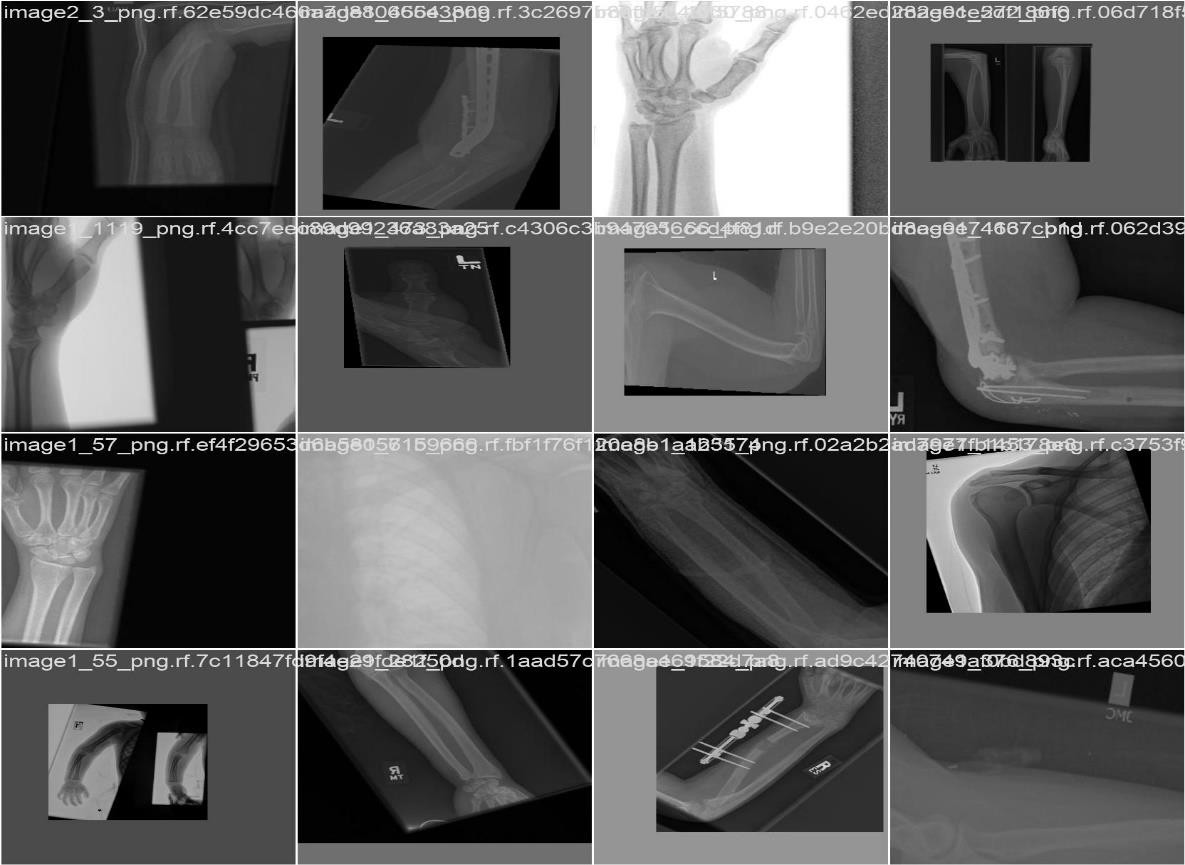








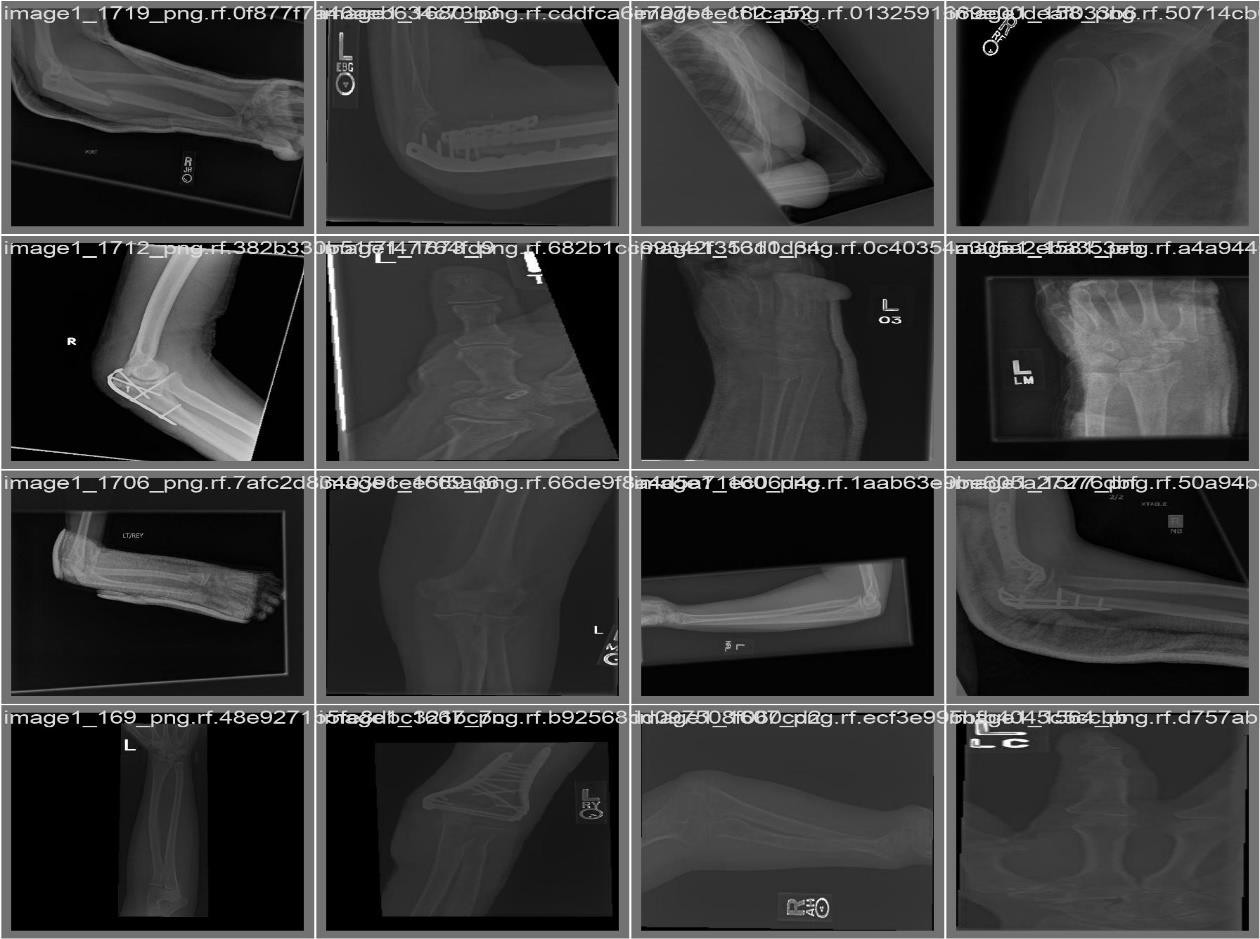










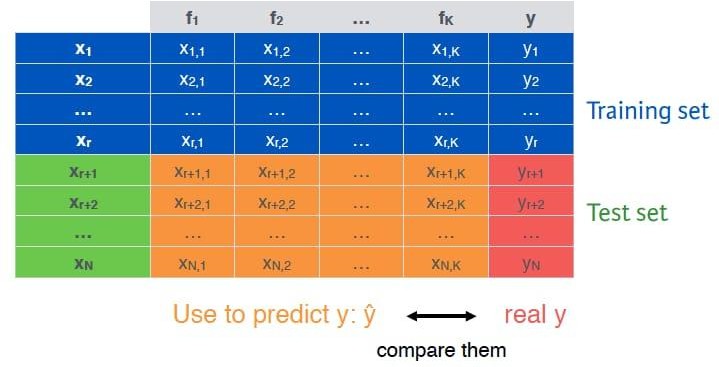




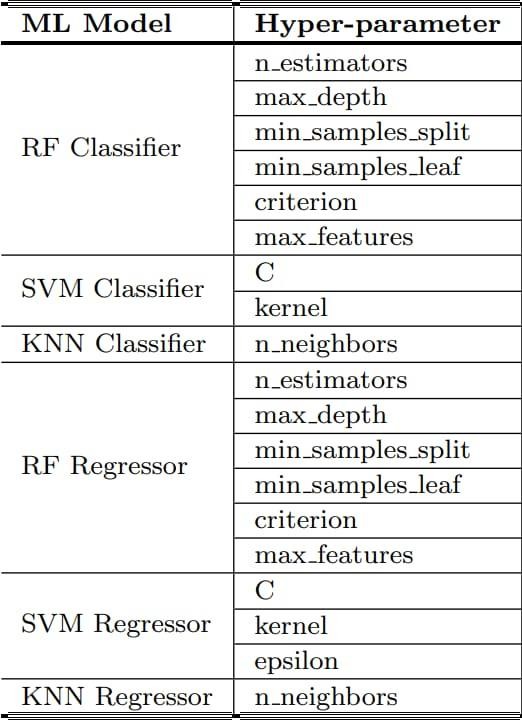


**Figure 4.5**: Batch Images

**Description:** The above images are classified as batches indicating all types of bones in each part, showcasing different angles and regions of bones, likely from limbs. Each tile is labeled with filenames or unique identifiers, indicating that these images might be part of a dataset for medical analysis or machine learning tasks. The X-rays vary in perspective, showing both close-up and zoomed-out views of bones.



**Table 1:** Dataset distribution for training and testing.



**Table 2:** Hyperparameters used for training and the model.

# CONCLUSION

The main goal of this project is to create a system that makes the work of the doctor faster and easier to figure out if a patient has a broken bone. Using machine learning, we’re fr

This system is designed to analyse X-ray scans and tell whether a bone is broken or not, or whether it has crack or not. It goes a step further by identifying the type and severity of the fracture. The X-rays used for training the system include all the sorts of fractures like simple, complex and even tiny hairline cracks along with healthy bones for comparison. This way, the model is prepared to handle a variety of cases, making it reliable for real-world use.

Bone fractures are becoming round the world more due to things like aging populations, urban lifestyles and rising accident rates. Spotting even small fractures is crucial because a missed or delayed diagnosis can lead to serious complications or slower recovery times. This system is designed to help doctors and radiologists detect fractures quickly and accurately, saving valuable time and improving patient care.

What makes this system especially helpful is that it doesn’t just provide a yes or no answer. It also highlights the part of the X-ray it focused on, so doctors can see exactly why the system is a valuable tool rather than a “black box”.

This technology has the potential to make a big difference in healthcare. By reducing the workload on radiologists and cutting down on diagnostic errors, it can help doctors focus more on treatment. It’s also particularly useful in places where there aren’t enough experienced radiologists, making it a lifesaver in resource limited areas.

The system is designed to work seamlessly with the existing model hospital technology, like PACS (Picture Archiving and Communication Systems), and it’s useful and built to be fast and user friendly. In short, this project aims to transform how fractures are detected. By combining smart technology with real world, it’s a toward better, faster and more reliable healthcare for patients everywhere. Doctors or technicians can upload an X-ray and get results almost instantly, which means patients get the care they need without unnecessary delays.

# REFERENCES

Enhancing diagnosis: ensemble deep-learning model for fracture detection using X-ray <https://www.sciencedirect.com/science/article/pii/S0009926024004197>

## RESEARCH PAPERS:

1. Achawale, M. A. More, Y. Bankar, M. A. Ghuge. M. P and Zole, M. G, 2024 BONE FRACTURE DETECTION USING CONVOLUTIONAL NEURAL NETWORK (CNN). JOURNAL OF BASIC SCIENCE AND ENGINEERING.
2. Alkhatib, A. J, Alharoun. M and AlZoubi, A, 2024. A Deep Learning Framework for timely bone fracture detection and prevention. Information Sciences with Applications.
3. Bone fracture detection and classification using Deep Learning Techniques. In Driving and Smart Medical Diagnosis Through AI-Powered Technologies (AI) APPROACHES. In AIP Conference Proceedings (Vol. 2802. No, 1). AIP Publishing.
4. Khan. M. I, 2024 Bone fracture identification with using deep learning model Resnet50. International Journal of computing and digital systems.
5. Devi. M. S, Aruna. R, Alufti. S, Punitha. P and Kumar, R. L, Bone fracture quantization and systemized attention gate. UNet-based deep learning framework for bone fracture classification Intelligent Data Analysis.
6. M. Singh, S. Kumar, Ruchilekha and M. K. Singh,
7. M. Singh, S. Kumar, Ruchilekha and M. K. Singh, “A Comparative Study of Feature Extraction Techniques and Similarity Measures for image Retrieval,” 2022 International Conference on Futuristic Technologies (INCOFT), Belgaum, India, 2022,

pp. 1-7, doi:10,1109/INCOFT55651.2002.10094430.

1. Rinisha Bagaria, Sulochana Wadgwani, Arun Kumar Wadhwani, Bone fractures detection using support vector machine and error backpropagation neural network, Optik, Volume 247,2021,168021,ISSN 0030-4026
   * Detection of bone fracture based on machine learning techniques: Kosrat Dlshad Ahmed, Roojwan Hawezi I images: Information System Engineering Department, Erbil Polytechnic University, Erbil, Iraq.
   * Analysis of X-Ray Images with Image Processing Techniques: A Review
   * Fracture detection from X-ray images using different Machine Learning Techniques.

**YouTube links**: <https://youtu.be/9qCWNsdazn4?si=54s9NTFre7au0q9e>

<https://youtu.be/UdTGPXUrHx8?si=cFQObdbpQ7ORNced>

<https://youtu.be/aQHxCjb58QU?si=mP3g9FKrp4FRsvlk>

<https://youtu.be/XJpC8VI7Zio?si=Z7ylGDd23hhemLEH>