Deep learning based localization and segmentation of wrist fractures in bone X-rays

A thesis submitted to the
University of Petroleum and Energy Studies

For the Award of
Doctor of Philosophy
In
Computer Science & Engineering

By
DEEPA JOSHI

Sept. 2022

Supervisor
Dr. THIPENDRA P. SINGH

School of Computer Science (SOCS)
University of Petroleum & Energy Studies
Dehradun- 248007: Uttarakhand

Deep learning based localization and segmentation of wrist fractures in bone X-rays

A thesis submitted to the
University of Petroleum and Energy Studies

For the Award of
Doctor of Philosophy
In
Computer Science & Engineering

By
DEEPA JOSHI
500064211

Sept. 2022

Supervisor
Dr. THIPENDRA P. SINGH
Professor, School of Computer Science
University of Petroleum & Energy Studies

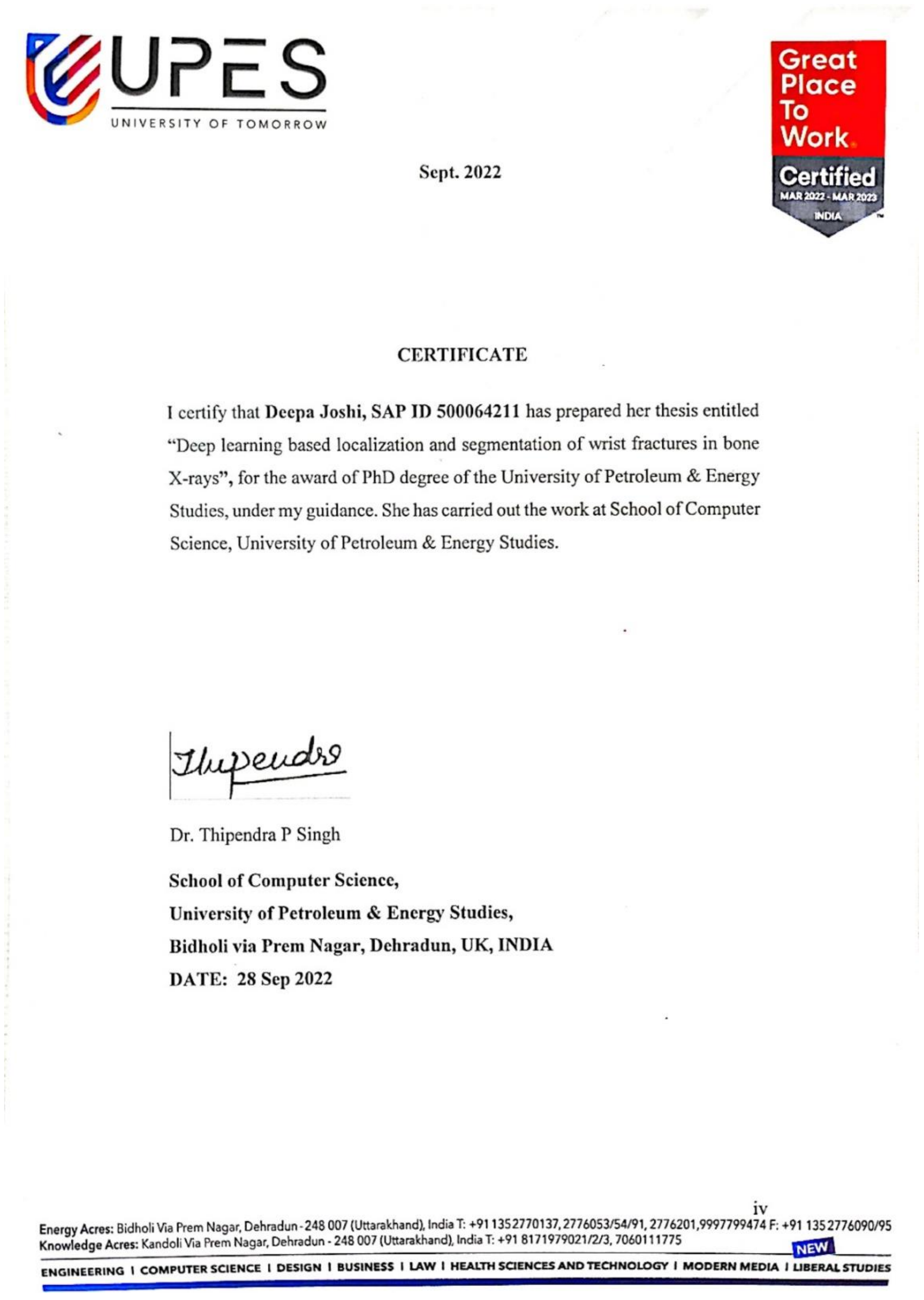
School of Computer Science (SOCS)
University of Petroleum & Energy Studies
Dehradun- 248007: Uttarakhand

DECLARATION

I declare that the thesis entitled “**Deep learning based localization and segmentation of wrist fractures in bone X-rays**” has been prepared by me under the guidance of Dr. Thipendra P Singh, Professor, School of Computer Science, University of Petroleum & Energy Studies. No part of this thesis has formed the basis for the award of any degree or fellowship previously.

Deepajoshi

DEEPA JOSHI
School of Computer Science,
University of Petroleum & Energy Studies,
Bidholi via Prem Nagar, Dehradun, UK, INDIA



UPES
UNIVERSITY OF TOMORROW

Great Place To Work. Certified
MAR 2022 - MAR 2023
INDIA

Sept. 2022

CERTIFICATE

I certify that **Deepa Joshi, SAP ID 500064211** has prepared her thesis entitled "Deep learning based localization and segmentation of wrist fractures in bone X-rays**"**, for the award of PhD degree of the University of Petroleum & Energy Studies, under my guidance. She has carried out the work at School of Computer Science, University of Petroleum & Energy Studies.

Thipendra

Dr. Thipendra P Singh

School of Computer Science,
University of Petroleum & Energy Studies,
Bidholi via Prem Nagar, Dehradun, UK, INDIA
DATE: 28 Sep 2022

iv

Energy Acres: Bidholi Via Prem Nagar, Dehradun - 248 007 (Uttarakhand), India T: +91 1352770137, 2776053/54/91, 2776201,9997799474 F: +91 1352776090/95
Knowledge Acres: Kandoli Via Prem Nagar, Dehradun - 248 007 (Uttarakhand), India T: +91 8171979021/2/3, 7060111775

ENGINEERING | COMPUTER SCIENCE | DESIGN | BUSINESS | LAW | HEALTH SCIENCES AND TECHNOLOGY | MODERN MEDIA | NEW LIBERAL STUDIES

ABSTRACT

In this day and age, X-rays are the principal instruments for assessing suspected fractures in humans. It takes significant time and requires experienced radiologists or trained orthopedic surgeons to examine X-ray images manually. Inability to diagnose and treatment delays owing to unnecessary referrals by primary care clinicians are induced by the excessive workload and shortage of radiologists in small settings and primary/community health centers. Furthermore, the lack of qualified radiologists and orthopedic surgeons in medically underserved regions, such as rural India, has driven us to create an automated fracture detection model. We have constructed a deep neural network to detect, localize, and segment the wrist region to find fractures near the wrist joint in radiographs. The orthopedic surgeon manually constructed a bounding box and segmented mask to annotate the fractures. We employed datasets from two separate domains to ensure that the model converges more quickly.

Wrist Fracture Dataset (WFD) and Surface Crack Dataset (SCD) have been created and annotated. The WFD was obtained from the Doon Hospital in Dehradun, India, between February 2019 and March 2020. The number of wrist fracture images obtained from the hospitals is 315 consisting of 733 annotations/cracks which is insufficient to generate accurate results using deep learning techniques. Therefore we have incorporated state-of-the-art COCO and self-collected Surface Crack Datasets (SCD) for better model generalization. COCO dataset does not include images from medical domain, more specifically there are no images which has crack like pattern in it. As a consequence, we have developed surface crack dataset. The surface crack dataset consists of 3,000 images collected by capturing the minute cracks, which has similar patterns as the bone fracture cracks. SCD consists of pictures taken from walls, pavements, and roads, created using a mobile camera. To overcome the obstacles in data collecting and labeling in diagnosing wrist fractures, a subset of the dataset is made freely accessible for research.

The suggested architecture substitutes the last-level max pool layer of the architecture with a concatenation of ACP AdaptiveConcatPool (ACP), AdaptiveAvgPool (AAP), and AdaptiveMaxPool (AMP) layers utilizing Feature Pyramid Network (FPN) as the backbone architecture. For improved model convergence, the notion of freezing and unfreezing the network is applied during the transfer learning cycles. Each radiograph is assigned a ground truth label to test the model's correctness. The principle of Intersection over Union (IoU) is utilized to evaluate the performance measure for fracture detection and localization using the Average Precision (AP) value. The output of the suggested model is contrasted with the radiologists' annotated ground truth label and results from related investigations. For fracture detection, an average precision of 92.278% on a scale of 50^0 and 79.003% on a strict scale of 75^0 was reported. For fracture segmentation, an average precision of 77.445% on a scale of 50^0 and 52.156 on a strict scale of 75^0 was reported.

ACKNOWLEDGMENT

Ph.D. journey is truly a life-changing experience and a confluence of multiple learning for me. It is not possible to do without the support and guidance that I received from many people. I am extremely thankful to all of them. I express my heartfelt gratitude to my research supervisor Dr. Thipendra P Singh for his unconditional support and motivation during the entire research program. His guidance always lent a hand to me not only in research but in real life also. His simplicity, positivity, honesty, and discipline inspire me in every second of my life. I wish and try to be as honest and disciplined as he is, for my entire life. Without his continuous support, guidance, and constant feedback, this Ph.D., would not have been achievable.

I gratefully acknowledge, Dr. Anil Kumar Joshi, Prof. and Head of Orthopedics, Government Doon Medical College, Dehradun for providing the X-ray samples. Being an interdisciplinary area of research, it is highly required to take input from Radiologists to work as an expert for validation of the model. He helped in providing and labeling the wrist fracture images for the development of the dataset used in the research.

My deep appreciation goes to Mr. Vidyanand Mishra, Ms. Shahina Anwarul, Ms. Tripti Misra, and Ms. Diksha Goel, Ms. Simran Oberoi, Mr. Gargeya Sharma for their much need support during this Ph.D. program. They were always helpful and provided me with their assistance throughout my research journey.

I am blessed with a beautiful and supportive family, I am unable to express my gratitude in words to my father Mr. R. D. Joshi, mother Ms. Kamla Joshi, elder sister Ms. Neha Pant, younger brother Mr. Deepanshu Joshi and sister-in-law Ms. Kavita Joshi for always believing in me and encouraging me to follow my dreams.

Above all, I owe it to all almighty *Parameshvar* for granting me the wisdom, health, and strength to undertake this research task and to shower his blessings on me to complete my thesis.

.

DEEPA JOSHI

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATION

Acronym	Meaning of Abbreviation
AAP	AdaptiveAvgPool
ACP	AdaptiveConcatPool
AI	Artificial Intelligence
AMP	AdaptiveMaxPool
AP	Average Precision
ASPP	Atrous Spatial Pyramid Pooling
AUC	Area under the ROC Curve
BPNN	Back Propagation Neural Network
CAD	Computer Aided Design
CI	Contextual-Intensity
CLAHE	Contrast Limited Adaptive Histogram Equalization Algorithm
CNN	Convolutional Neural Networks
COCO	Common Objects in Context
CT	Computed Tomography
DCNN	Deep Convolutional Neural Network
DL	Deep Learning
DT	Decision Tree
EMR	Electronic Medical Record
ESS	Efficient Subwindow Search
FCNT	Fully Convolutional Network Based Tracker
FN	False Negative
FP	False Positive
FPN	Feature Pyramid Network

FRRN	Full Resolution Residual Networks
GLCM	Gray Level Co-occurrence Matrix
GO	Gabor Orientation
GPU	Graphics Processing Unit
IBFDS	Intelligent Bone Fracture Detection System
ICA	Independent Component Analysis
IGD	Intensity Gradient Direction
ILSVRC	ImageNet Large Scale Visual Recognition Competition
IoU	Intersection over Union
IRB	Institutional Review Board
KDD	Knowledge Discovery in Databases
KNN	K-nearest neighbor
LSTM	Long Short Term Memory
MLP	The Multilayer Perceptron
MRF	Markov Random field
MRI	Magnetic Resonance Imaging
MURA	Musculoskeletal Radiographs
NB	Naive Bayes
NN	Neural Network
NSA	Neck Shaft Angle
PSNR	Peak Signal-to-Noise Ratio
PSNR	Peak Signal-to-Noise Ratio
R-CNN	Region-based Convolutional Neural Network
RPN	Region Proposal Network
RU	ResNet Unit

SACEN	Simultaneous Automatic Contrast adjustment, Edge enhancement and Noise removal
SCD	Surface Crack Dataset
SIFT	Scale-Invariant Fourier Transform
SIFT	Scale-Invariant Fourier Transform
SRF-FF	Stacked Random Forests Feature Fusion
SURF	Speeded Up Robust Features
SVM	Support Vector Machine
TN	True Negative
TP	True Positive
WEAD	Wavelet based Anisotropic Diffusion Algorithm
WFD	Wrist Fracture Dataset
WHO	World Health Organization

CHAPTER-1

INTRODUCTION

This chapter begins with a motivation for our study, followed by a description of the research difficulties and a summary of the contributions made by this thesis. Finally, we conclude this chapter with a thesis outline.

1.1 Motivations

An incomplete or full break in the bone is known as a fracture. The primary cause of fracture is high impact or force applied to a bone that is structurally capable of withstanding. Traumatic and stress are commonly found bone fractures in the human body. Stress fractures are common among sports (such as gymnasts, dancers, and long-distance runners) and military people and are caused by repetitive load-carrying strain on a healthy bone [1]. Traumatic fractures are caused by vehicle accidents, serious falls, or purposeful causes such as physical abuse. A fracture can also happen for several other reasons, such as osteoporosis (a disease that weakens bones), cancer, or the brittle bone condition known as ontogenesis imperfect. According to the World Health Organization (WHO) report, 1.66 million people suffer from hip fractures every year throughout the world, and the rate is expected to rise by three to four times by the year 2050 because of the worldwide increase in the number of older people [1]. All bone fractures are divided into seven major categories [2] that are depicted using Figure 1.1.

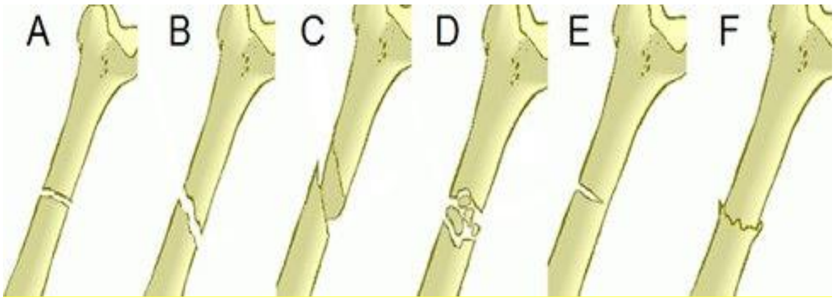


Figure 1.1 The primary bone fracture types- Transverse, oblique, spiral, comminuted, greenstick, and impact fractures are listed in alphabetical order from A to F [2].

A. **Transverse Fracture**: It is the simplest type of fracture where the bone is broken as a horizontal line.

B. **Oblique Fracture**: It is a fracture type where the break extends in a slanting direction, caused by indirect or rotational force.

C. **Spiral Fracture:** It is a fracture type where the break spirals around the bone, common in a twisting injury.

D. **Comminuted Fracture**: It is a fracture type where the bone breaks into several pieces.

E. **Greenstick Fracture**: It is an incomplete fracture type where the broken bone is not completely separated.

F. **Impacted Fracture**: It is a fracture type where the bone breaks, but the two ends of the fractured bone are forced together. It produces a rather stable fracture that can heal readily but at the cost of some length lost.

X-rays, Magnetic Resonance Imaging (MRI), and Computed Tomography (CT) are various medical imaging modalities used to capture images of the affected body area. A radiology specialist interprets these recorded images to make a medical diagnosis and then recommends therapy. The oldest, quickest, and most popular imaging technique is an X-ray, which analyses potential fractures by taking pictures of the body's interior organs [3]. It has emerged as the go-to analytical tool for examining patients for fractures because it is widely accessible in locations where other, more expensive imaging modalities would not be. Radiologists or physicians use visual inspection to evaluate X-ray samples to determine the existence and type of fractures in different bones. To acquire a more detailed, cross-sectional image of the bone that may be missed during an X-ray examination, the need for advanced imaging technologies such as MRI and CT scans emerges [3].

The manual examination and categorization of fractures involved in radiographic image interpretation take a lot of time and effort. False detection rates and poor fracture healing could be caused by a shortage of doctors in medically underserved areas, a shortage of experienced radiologists in overloaded clinical settings, and weariness from excessive workloads. Additionally, because rarely a second examiner is present, the chance of incorrect identification due to incomplete interpretation of the X-ray image is increased. A fracture diagnosis error resulting from incorrect fracture identification was described in 41% to 80% of instances [4-5]. The examiner's ability to detect anomalies may be greatly diminished due to weariness brought on by analyzing numerous musculoskeletal pictures, according to many studies [6–8]. Computer vision systems may be a potential answer to such issues if they can swiftly offer a reliable second opinion in identifying questionable fracture instances.

1.2 Problem Descriptions

In the past, numerous low-level pixel-processing methods for predicting human bone fractures, including noise reduction, segmentation, and feature extraction, were used [9]. Before extracting characteristics from the image, obtaining the region of interest was relatively common by separating bone regions from fleshy portions. Several aspects from the image were retrieved and provided to the classifier to forecast the occurrence of fractures, including textual, shape, edges, horizontal, and vertical lines. However, several deep learning techniques have since superseded this strategy.

Artificial intelligence is a subfield of computer science that develops computer systems that can simulate human intellect. This word is made up of the phrases "Artificial" and "Intelligence," which signify "a thinking ability developed by humans." Machine learning is a branch of artificial intelligence in which computers learn from historical experiences or information without being explicitly programmed. Deep learning, one of artificial intelligence's fastest-growing sub-fields, has achieved great success in medical imaging by offering greater accuracy than other methods [10]. Convolutional Neural Networks (CNN), which have fully linked, sub-sampling, and convolution layers, are types of deep learning architecture [10]. Since the creation of the CNN model, which can "learn the features" rather than manually programming them into the system, feature extraction approaches have undergone a significant transformation. While the fully connected layer is utilized for classification, CNN's convolution and subsampling layers are used for feature learning. These architectures have been quite successful since the model can learn features independently rather than manually adding them to the system. In order to improve patient care, Computer-Aided Design (CAD) systems can help medical professionals by recommending the type of treatment necessary for disease diagnosis [11]. When combined with X-ray machine software, a fracture diagnosis system might speed up the examination procedure by producing the best findings.

Research studies on the subject of fracture detection demonstrate that there have been significant increase in computer vision-based crack detection systems over the past ten years. However, there are a few obstacles in this field, and these challenges drive the investigation in this thesis.

Conventional image processing approaches in fracture identification struggle with image non-uniformity and variable lighting conditions [9]. Large bone fracture datasets with reliable ground truth labels are scarce for building and testing detection algorithms, particularly deep neural network-based models. The existing research articles focus on fracture detection and its localization using deep neural network architectures pre-trained on state-of-the-art non-medical datasets.

Fracture detection has incorporated traditional machine learning techniques like the Nearest Neighbor (NN) [12] and Support Vector Machines (SVM) [13]. Implementing such algorithms in fracture detection should provide a quantized visual representation that describes the crack that must be classified. Depending on handcrafted crack features, such algorithms benefit from a limited amount of images for learning. Therefore, establishing a reliable method for feature extraction, such as morphological operations and edge/line detection, is a vital step in traditional machine learning-based systems [14]. More sophisticated techniques include wavelet-based texture extraction methods. The established techniques for machine learning are simple to comprehend and diagnose. However, a fundamental issue with these models is that they are limited to generalizing since they require robust fracture features, which are challenging to achieve when dealing with a wide variety of real-world settings.

Deep convolutional neural network-based techniques have excelled in fracture detection methods over the past ten years [23-30]. These models do not need pre-defined, manually created features because they can learn highly complicated fracture characteristics straight from the raw images. However, deep learning-based approaches encounter two major difficulties.

First, they require sizable datasets with sparsely available ground truth annotations. This has significantly hampered any worthwhile research in this area. A deep network can be trained to detect and locate the cracks by being given a large bone fracture dataset.

Second, the research articles primarily focus on fracture detection and localization using pre-trained deep neural networks on non-medical datasets. Merely providing annotations in the form of a bounding box surrounding the crack does not somehow help to visualize its shape. If a fracture expanded horizontally or vertically has irregular shapes, then a bounding box would involve extraneous bone regions while the model is trained. Consequently, each image is annotated by drawing a more complex shape around the fracture, such as a polygon.

Instance segmentation, which integrates faster-RCNN (Region-based Convolutional Neural Network) and semantic segmentation, can be employed to overcome the abovementioned issues. Considering this, the following set of objectives have been drafted.

1.3 Objectives

Analysis and design of bone fracture localization and segmentation model for assisting radiologist in accurately detecting wrist fractures.

Sub-Objectives

1. To collect bone X-ray images, perform data labeling, data preprocessing, and dataset splitting into train, validation and test set.
2. To propose a model which classifies fracture into binary class (Healthy and Fracture).
3. To extend the model for localization and segmentation of suspected fractures in wrist bone and analyzing performance thereof.

1.4 Contribution of the Thesis

The general goal of this thesis is to propose an accurate fracture detection and segmentation model for wrist bone fractures from X-ray images. The research contribution is summarized below.

To begin, we propose two new datasets: the Wrist Fracture Dataset (WFD) and the Surface Crack dataset (SCD).

WFD consists of wrist bone X-ray images collected from the Government Doon Medical Hospital in Dehradun, India, between February 2019 and March 2020. SCD comprises photographs taken from walls, pavements, and roads with a mobile camera. We included a surface crack dataset with crack patterns similar to wrist bone fractures. SCD and WFD are both preprocessed before being fed into the network. Following that, the images go through a labeling and augmentation process. All of the images in our datasets were labeled manually, taking longer but less error-prone than the automatic annotation software. To avoid data collection challenges and wrist fracture labeling, a portion of the dataset is made publicly available for research.

Second, we demonstrate a novel fracture localization and segmentation model of three sub-architectures: the Backbone network, the Region Proposal Network

(RPN), and RoIAlign (Region of Interest). The fractures are identified and segmented using an instance segmentation technique that combines faster-RCNN and semantic segmentation. To our knowledge, this study is the first to concentrate on developing a segmentation mask around the class labels in order to identify wrist fractures. The idea behind combining segmentation and localization of wrist fractures is to improve the visualization of fracture shape. The fracture shape has been observed to extend in vertical and horizontal directions in most of the X-ray images collected. Automatic fracture detection is improved by creating a segmented mask around the fractures.

The backbone network, which consists of a top-down and bottom-up pathway, is in charge of extracting semantically significant features from the input image. We replaced the last-level max-pool layer in the backbone architecture with a linear combination of AdaptiveConcatPool (ACP), AdaptiveMaxPool (AMP), and AdaptiveAvgPool (AAP) layers. The fracture localization and segmentation techniques are then used to detect smaller and larger objects. The sub-architecture of this stage consists of two networks: RPN and the RoIAlign layer. The maximum and average activations are preserved in the proposed architecture to allow the neural network to choose the most effective strategy without needing to conduct individual experimentation. A modified version of RoIAlign proposed in the mask-RCNN architecture is used to crop the region of interest precisely using a new technique (RoIAlignv2). The neighboring indices are calculated precisely by subtracting the half-pixel offset (0.5) from RoI coordinates. This method overcomes the disadvantages of using bilinear interpolation with a slightly off-aligned pixel value. The mask around the cracks is then generated by combining a parallel layer with the existing object detection framework.

Third, the proposed methodology applies the transfer learning strategy [15] to transfer knowledge from the state-of-the-art non-medical dataset known as Common Objects in Context (COCO) to the wrist fracture dataset. It is of limited utility to train the models for bone abnormality detection using general-purpose datasets like COCO, which contain specific everyday object categories, including vehicles, animals, home objects, etc. Instead of a non-medical dataset to train the model, we used a dataset of surface cracks with fracture patterns resembling those of wrist bone fractures. A surface crack dataset previously fine-tuned with the COCO dataset is used to fine-tune the wrist fracture dataset.

Fourth, the proposed model is fine-tuned by choosing the relevant parameters and hyperparameters for analysis and experimentation. The experiments are carried out in three stages. In Phase I, the weight file from the COCO dataset pre-trained on the standard mask-RCNN architecture is used. In Phase II, the weight file from Phase I is used instead of a random weight initialization technique to fine-tune the proposed model on the SCD. We also used the concept of freezing and unfreezing specific layers of the proposed architecture. The first layers of the CNN architecture are intended to obtain generic features, such as the first layer identifying simple gradients of a line, the second layer discovering simple shapes, and the third layer combining line and shape features. The final layers, on the other hand, are more focused on the specific task, such as finding the image's crack patterns in our problem statement. It is unlikely that better features will be generated at the initial layers of a CNN architecture while updating the gradients because the features predicted by a CNN architecture will remain the same regardless of the dataset used. As a result, the proposed model's initial layers are frozen during the second phase of training (not trained). The entire architecture is unfrozen and trained in the third phase by updating the learned parameters. The network updates the parameters using a learning rate finder curve and a differential learning rate technique. For deep neural architecture segments, this method employs different learning rates.

1.5 Thesis Outline

The following is an outline of the thesis. The second chapter delves into the fundamental components of computer vision technologies, starting with traditional image processing techniques and progressing to deep convolutional neural network designs employed in bone fracture detection and segmentation. The self-collected wrist bone dataset and surface crack dataset are introduced in Chapter 3. A novel architecture for fracture identification and segmentation on wrist X-ray bones is put forward in Chapter 4. In Chapter 5, we discussed the hyperparameters required to fine-tune the model. Furthermore, the suggested architecture's performance is evaluated and compared to existing approaches based on wrist fracture detection and segmentation. In Chapter 6, we summarize the thesis, reach conclusions, and talk about future research.

CHAPTER -2

LITERATURE SURVEY

A subfield of artificial intelligence called computer vision enables computers and systems to gather information from digital images, videos, and other visual inputs and to execute actions or formulate predictions based on that information. Computer vision connects a variety of scientific fields, including Computer science (Theory, Architecture, Systems, Algorithms), Engineering (Image processing, Natural Language Processing, Speech Processing, Robotics, Image Processing), Biology (Neuroscience), Mathematics (Machine Learning, Information retrieval), and Physics (optics). The Key Elements of computer vision include visual recognition tasks like image classification, object detection, localization, and segmentation. Since machine learning first appeared, nearly every industry has changed significantly to make room for AI technologies. One important area that has experienced significant advancements in the healthcare industry is where computer vision has assisted in resolving some of the most crucial problems. Computer vision has made significant contributions to the healthcare industry, from X-ray analysis to fracture detection in critical organs. This chapter presents a detailed description of how human wrist bone fractures have been analyzed using object detection and localization techniques. Furthermore, the current state-of-art of fracture detection is presented.

2.1 Image processing methods for fracture detection

X-ray is the most frequently used imaging modality for fracture detection due to its painless, economical and non-invasive nature, which has gained enormous popularity in medical imaging. Poisson, Gaussian, and salt and pepper noise are various types of noise artifacts commonly found in radiographs, particularly when collected in large quantities from the public domain such as the internet [16]. The need for handling such images rises largely as reducing one type of noise sometimes affects the other. Edge detection is another useful step in determining the boundaries of objects in the image. Gradient, Laplacian, and Sobel have commonly used methods of edge detection. The shapes and sizes of bone are non-identical in X-ray images due to the patients' differences in age and gender [16]. Normalization could be used to deal with size variations, but its results are unsatisfactory as it removes important texture information in shrunken images and adds noise and artifacts in the case of larger images. Hence, adaptive sampling is employed in various kinds of literature to sample X-ray images instead of scaling them [16-19]. Adaptive sampling does not require accurate extraction of bone contours as done by the authors in [20]. A slight variation of shape is accepted here. Image transforms such as wavelets and curvelets are powerful algorithms to obtain decent quality compressed images with higher Peak Signal-to-Noise Ratio (PSNR) and compression ratio resulting in lesser memory requirements to store medical images. Both wavelet and curvelet transform (a multi-scale method originating from wavelets) are commonly used for medical image compression, contrast enhancement, edge detection, and image registration [21]. They extract an enormous set of coefficients or features from the input image, where insignificant features are eliminated via a feature selection algorithm for better or faster classification.

After smoothing and edge detection, the primary step in various image-processing applications is the extraction of essential features (informative representations) from the image. Feature extraction focuses on extracting image characteristics that acquire visual image attributes. The classifier's performance depends on the perfect set of features retrieved from the image. Texture can be a useful cue for detecting diseases or tissue types in medical imaging. Visual texture is used for segmenting and discriminating objects from a background that has a repeated pattern of elements with some variability in element appearance and relative position. The spatial features of an image are described by its gray level, spatial distribution, and amplitude, where the amplitude is the simplest feature that discriminates bone tissues from X-ray images [22]. Table

2.1 demonstrates relevant review findings of the image processing methods used for fracture detection.

With the rise of deep learning neural networks, deep layers of Convolutional Neural Networks (CNN) replaced the feature extraction task. CNN is a multilayered neural network consisting of convolution, sub-sampling, and fully connected layers. Convolution and sub-sampling layers of CNN are part of the feature learning process, while a fully connected layer is used for classification. ConvNets or CNN can learn various low-level (minor details of the image, e.g., lines, dots or edges, etc.) and high-level features (built upon low-level features to detect objects and larger shapes) through abstraction in the layers. Features are extracted using CNN in recent approaches to fracture detection and classification [23-30].

Table 2.1 The table demonstrates relevant review findings of the image processing methods used for fracture detection.

Author	**Bone type**	**Relevant Review Findings**
[16], 2004	femur	- Image features are extracted by performing texture analysis of trabecular patterns in femur X-rays. - Neck Shaft Angle (NSA) is considered the feature for fracture detection.
[17], 2004	femur and wrist	- Active shape and appearance models are used to extract the contours of the femur X-rays. - Trabecular patterns in femur X-rays are subjected to texture analysis to extract image features. - The four distinct image features extracted from X-ray samples are NSA, GO, IGD, and MRF. - Instead of scaling the X-ray images, adaptive sampling is used to sample them.
[18], 2005	femur and wrist	- GO, IGD, and MRF are extracted from X-ray samples to detect femur fractures. - Instead of scaling the images, adaptive sampling is used to sample X-ray samples.

[19], 2007	femur	- GO, IGD, and MRF are extracted from X-ray samples to detect femur fractures.
[20], 2003	femur	- The NSA is the only characteristic that is thought to be useful for identifying fractures.
[23], 2017	pelvis	Three CNN-based models are used for preprocessing to account for variations in medical studies. - CNN-frontal is trained to recognize anatomical details in pelvis images and distinguish them from other images such as the chest, lateral hip X-rays, spinal images, etc. - CNN-bounding is a regression-based model trained to locate the femur neck and potential fracture sites. - CNN-metal is trained to include only pertinent hip fracture cases by excluding cases where the metal implant has occurred or where there may be another diagnostic challenge,
[31], 2011	tibia	-To simultaneously adjust contrast, enhance edges, and remove noise in tibia X-ray images, the SACEN technique is proposed. - Edges in edge-detected regions of tibia images are enhanced using the CLAHE algorithm [38], and noise is eliminated in gray areas of the non-edge region using the WEAD algorithm [39]. - In the second stage, the bone image from the X-ray is segmented. Next, using a region-growing algorithm, the diaphysis region is extracted from the epiphyses and fleshy regions of the tibia X-ray images. - Various texture features were extracted from the processed tibia X-ray images, including GLCM, GO, MRF, and IGD.
[32], 2017	multiple bones	- Scale-Invariant Fourier Transform (SIFT)-based feature extraction is used after the Haar wavelet transform [42] to improve image quality by reducing noise in the X-ray images.
[33], 2011	femur	- The input image is converted to binary form to increase calculation speed and simplify the process. - A median filter is used to remove fine particles after the Laplacian edge detector has detected the edges of the femur.

		- The shaft and non-shaft regions of femur images are separated using the K-means unsupervised clustering approach.
[34], 2012	tibia	-The quality of X-ray images is enhanced by combining an ICA and wavelet-based hybrid denoising technique. - Combining texture and shape features improves the performance of the classification system. GLCM, GO, MRF, and IGD are the main tools for extracting texture features in tibia images. - Shape features in tibia images are extracted using the fast Hough transformation algorithm [40] after segmenting diaphysis regions.
[35], 2013	hand bones	- The median filter reduces salt and pepper noise in bone X-ray images. - GLCM entropy, contrast, correlation, and homogeneity are used to extract texture features. - Using Weka supervised attribute selection, 84 features are ultimately chosen from thousands of features extracted from hand X-rays for fracture detection.
[36], 2013	long bones	- Histogram equalization is used to handle intensity variations between X-ray images. - A quick and effective filtering algorithm is used to handle Gaussian noise; this algorithm outperforms mean, Weiner, k-means, alpha-trimmed mean, and trilateral algorithms in terms of PSNR and mean absolute error. - Bone images are improved using a well-known filtering algorithm called Haar wavelets. - The k-fill algorithm is used to handle salt and pepper noise. - A modified version of the Canny algorithm is used to find edges. - A well-known Harris algorithm and a tensor-based corner detection algorithm are used to detect corners (the intersection of two edges) [41].
[37], 2018	tibia	- A grayscale image is created from an RGB image. - Tibia bone regions are enhanced in images using USM (Unsharp Mask Filter).

		- To detect breaks, the Harris corner detection algorithm is used.

2.2 Conventional machine learning algorithms for fracture detection

Various features extracted, such as textual, shape, edges, and horizontal and vertical lines, are fed into classification algorithms, which predict the occurrence of bone fractures and classify them accordingly. Once the ideal set of features is fed into the classifier, the accuracy of fracture detection is determined by the classifier chosen. As a result, appropriate features must be extracted to create a powerful classification model. Table 2.1 displays relevant review findings of conventional machine learning-based fracture detection algorithms.

Neck-Shaft Angle (NSA) was the sole feature used in the initial work on automatic fracture detection [20]. Radiologists classified the image as fractured if the NSA is less than 116^0. Using this model, 94.4% of training samples and 92.5% of test samples could be correctly identified. The model's inability to pick up on slight variations in the femur neck-shaft angle is the main cause of the test cases' 7.5% error rate. Trabeculae are found in the upper extremity region of the femur, and when a fracture occurs, they significantly alter the orientation of the femur's neck and head. The NSA can be used to identify these changes, but this method leaves local disruptions undetected. Therefore, a novel method is suggested that uses feature extraction from femur X-rays followed by classification to perform texture analysis of trabecular patterns to find such minor disturbances [16, 17]. Researchers in [16] extracted Gabor features, while [17] and [18] used the Gabor Orientation (GO) they had previously obtained from [16] and also acquired Intensity Gradient Direction (IGD) and Markov Random Field (MRF) texture features from femur X-rays. The classifier of choice is then fed these features to identify fractures in X-ray samples. In 30 X-ray samples, Gray Level Co-occurrence Matrix (GLCM) is the only feature used to classify femur fractures, and it achieves sensitivity and accuracy of 80% and 86.67%, respectively [33].

An approach to machine learning called ensemble techniques combines many base models to create a single, optimal prediction model. Ensemble learning approaches have enhanced the performance of machine learning models, making the model more reliable.

2.2.1 Ensemble based classification system

An ensemble machine learning technique combines various models or classifiers to create the best model that can most accurately predict our desired outcome. The fundamental idea behind ensemble models is to use multiple learning algorithms concurrently to produce better predictions than a traditional individual model. The ideal set of features that can be retrieved or learned from the image will determine how accurate the classifier is. However, by combining different classifiers and integrating the outcomes of every independent classifier, the accuracy could be further increased. A wide range of fields, including face recognition [43], geospatial land classification [44], video-based face recognition system [45], medical image segmentation [46], wind power forecasting [47], etc., have shown involvement in ensemble-based classification systems.

By avoiding overfitting issues and lowering bias and variance error in contrast to its component classifiers, these models have demonstrated better accuracy (low error). The significance of these models can be understood by the fact that ensemble-based models were used to achieve the best accuracy in several prestigious machine-learning competitions, including the well-known Netflix challenge [48], the Knowledge Discovery in Databases (KDD) cup 2009, and Kaggle. From 2003 to 2015, the most prevalent ensemble-based models for diagnosing human bone fractures used the Neural Network (NN), Support vector machine (SVM), and Naive Bayes (NB) algorithms.

With the introduction of multiple-classifier-based systems, where the individual results from base classifiers are fused, it has been observed that the effectiveness of the classifier is significantly improved [16; 18; 31; 34]. The diversity of the models influences the choice of the best classifier out of all the competing models. It is flawed to select a classifier solely based on training data accuracy. The methodologies listed below can be used to achieve some degree of diversity among the classifiers that make up an ensemble system, which is necessary for the system to perform well [49]:

1. Utilizing various classification algorithms in an ensemble system.
2. Applying the same classification algorithm with various instantiations or hyper-parameter configurations.
3. Making use of various feature sets:
    (a) Feature selection
    (b) Random selection
4. Making use of various training sets:
    (a) Bagging
    (b) Cross-validation

a) Bagging or Bootstrap Aggregating

Non-hybrid classifiers combine the same classification algorithm with different instantiations or hyper-parameter settings to create ensemble models, a widely used ensemble modeling technique. Bootstrap aggregation is one of the earliest and most basic ensemble-based techniques. It trains several models of the same learning algorithms using subsets of randomly chosen datasets drawn from the initial training set with replacement [50]. The output of the multiple-classifier or ensemble is predicted based on the majority votes of the individual classifiers. This algorithm can be modified in many ways that improve the model's performance [51]. The most well-known use diverse training data for individual classifiers, and the other uses various classification algorithms. The bagging process involves running various classification algorithms in parallel while using training subsets chosen randomly from the training dataset. The outcome of this ensemble model is predicted by individual classifiers voting in favor of their predictions. The bagging process involves concurrently executing distinct classification algorithms using training subsets chosen randomly from the training dataset [52]. The outcome of this ensemble model is predicted by individual classifiers voting in favor of their predictions, as depicted in Figure 2.1.

b) Boosting

Boosting is a straightforward variation of the bagging technique that aims to enhance the classification model by sequentially turning weak learners into strong learners, each trying to improve its forerunner [53]. The primary distinction between bagging and boosting is that the former uses a parallel training stage in which each model is developed independently. In contrast, the latter uses a sequential approach in which the success of a previous classifier determines the architecture of the current model [54]. Similar weights are initially assigned to the data in a sequential process, and these weights are then redistributed after each training stage to allow subsequent learners to focus more on the misclassified cases now associated with higher weights. Boosting is a sequential process in which initial input points and data are given similar weights and chosen randomly from the training set. After each training and testing session, misclassified samples are identified and given higher weights. This enables later learners to emphasize cases incorrectly classified and more likely to be chosen for the next classification [55]. The boosting process is illustrated in Figure 2.2.

c) Stacked Ensembles

Base learners are the first layer of a multi-layer learning process known as stacking, followed by lower-tier meta-learner stages that incorporate base learners as input to create the optimal combination of first-level base learners. In 1992, the concept of a *super learner* was first put forth [56], but it was not until 2007 that it was put into practice with improved performance [57], illustrative of how stacked ensembles support the creation of the best learning model. A well-known machine learning algorithm called random forest uses bagging to combine weak learners (like decision trees) into a single, powerful learner.

With 145 X-ray images from different body parts, including the foot, knee, arm, hand, ankle, and lower leg, and 10-fold cross-validation, a fracture identification method is built using the Stacked Random Forests Feature Fusion (SRF-FF) technology [58]. As shown in Figure 2.3, the first layer of a four-layer random forest uses five decision trees, while the subsequent layers use fifteen trees. The Efficient Sub window Search (ESS) algorithm is used to determine the regions with the highest likelihood of fracture occurrences after the classifier has been trained to provide confidence score maps that reflect the likelihood of fractures in X-ray images [59]. The proposed model outperforms a single layer of stacked random forest and SVM in terms of locating and identifying fractures in X-ray images.

Divide and conquer is a different ensemble strategy based on the notion that each sub-problem is easier to solve than the main problem. It needs a large training set and challenging problems to form larger clusters and produce successful results. The challenging issue of fracture identification is split into the kernel space of the Gini SVM as opposed to the feature space due to the lack of a larger training set [19]. The Gini SVM is first trained on training set T, and on testing set V, the error is calculated. In the next step, a new SVM and training set T' of T is used to categorize the new validation set further V' (a subset of V), which was chosen using the calculated error. This architectural design increases the accuracy of the SVM by ensuring that the lower level SVM (child) always facilitates the performance of the higher level SVM (Parent).

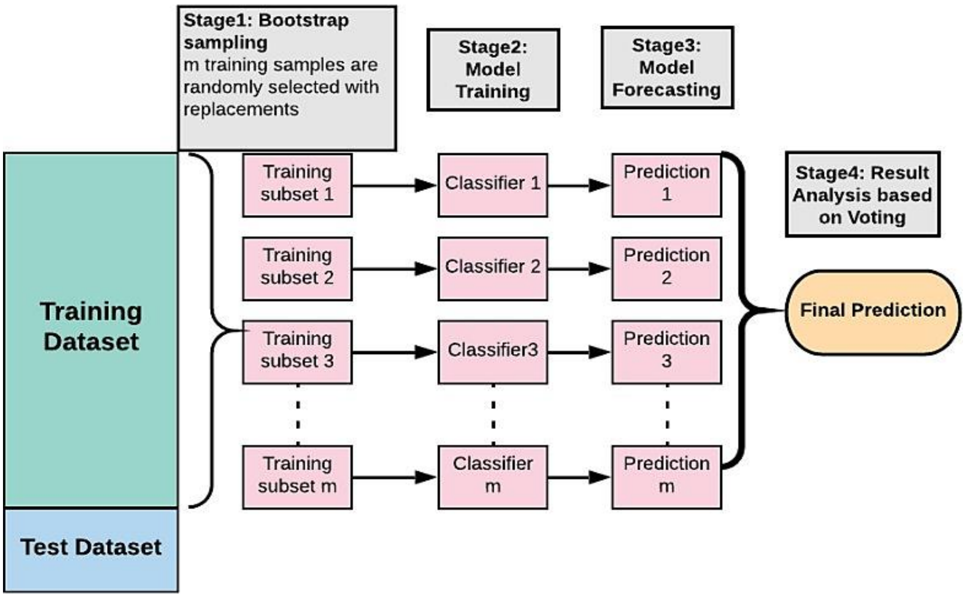


Figure 2.1 Various classification algorithms are run in parallel using the subsets drawn randomly from the training dataset during the bagging process. Individual classifiers in this ensemble model cast votes in favor of their predictions to determine the model's outcome.

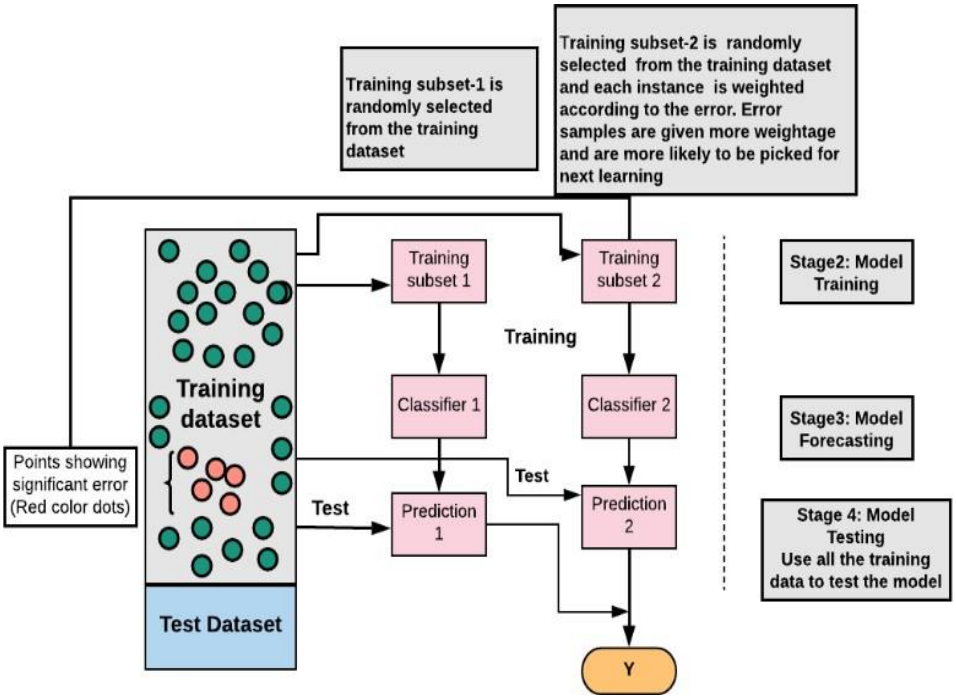

Training subset-1 is randomly selected from the training dataset
Training subset-2 is randomly selected from the training dataset and each instance is weighted according to the error. Error samples are given more weightage and are more likely to be picked for next learning
Training dataset
Test Dataset
Points showing significant error (Red color dots)
Training subset 1
Training subset 2
Training
Classifier 1
Classifier 2
Test
Test
Prediction 1
Prediction 2
Y
Stage2: Model Training
Stage3: Model Forecasting
Stage 4: Model Testing
Use all the training data to test the model

Figure 2.2 Boosting is a sequential process in which the initial input points and data are selected randomly from the training set and given similar weights. After each training and testing session, samples that were incorrectly classified are identified and given higher weights.

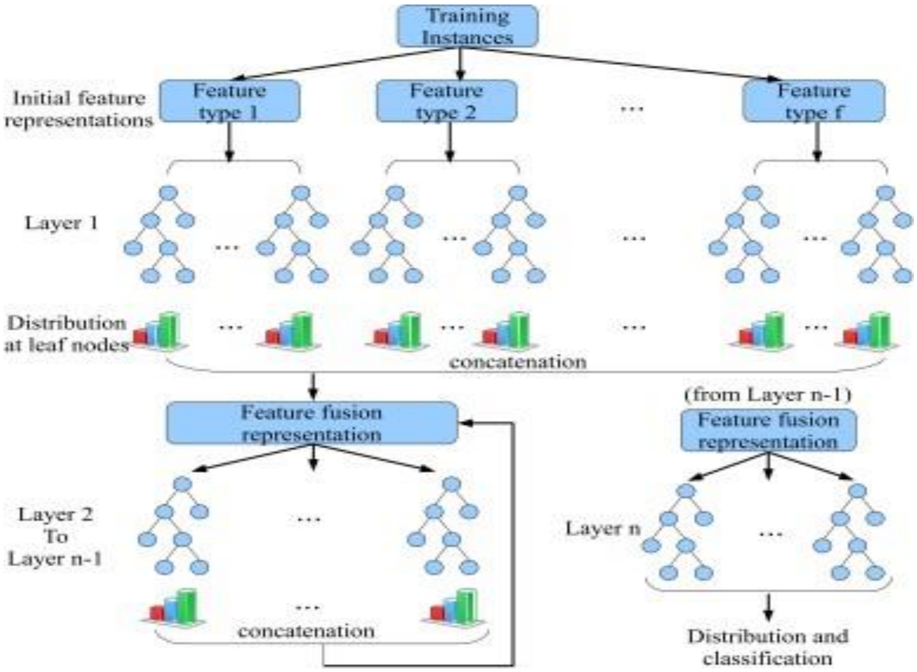

Training Instances
Initial feature representations
Feature type 1
Feature type 2
...
Feature type f
Layer 1
Distribution at leaf nodes
concatenation
Feature fusion representation
(from Layer n-1)
Feature fusion representation
Layer 2 To Layer n-1
Layer n
concatenation
Distribution and classification

Figure 2.3 Flow chart of stacked random forests feature fusion [58]

Table 2.2 The table presents pertinent review results of traditional machine learning-based fracture detection algorithms.

Ref	Bone type	Relevant review findings
[16]	femur	The following feature-classifier combinations have been trained to identify femur fractures: 1. NSA + Bayesian 2. SVM+NSA 3. SVM + NSA + NB (ensemble) Rules 1 of 2 and 2 of 3 provide the best fracture-classifier combination, which is as follows: If any 1 out of 3 or any 2 out of 3 combinations detects the fracture, the fracture has been detected.

[17]	femur and wrist	Classification is done after feature extraction to find fractures in wrist and femur images. The following feature-classifier combinations have been trained to find fractures in X-rays of the wrist and femur: 1. Thresholding + NSA 2. GO + Bayesian 3. GO + SVM 4. IGD + Bayesian 5. SVM + IGD 6. MRF + SVM If 2 out of 6 or 2 out of 4 combinations detect the fracture, the femur has a fracture. When predicting fractures in wrist images, a combination of MRF and SVM has demonstrated the best performance.
[18]	Femur and wrist	The following feature-classifier combinations have been trained to find fractures in X-rays of the wrist and femur: 1. GO + Gini-SVM 2. MRF + Gini-SVM (performed best in terms of sensitivity and accuracy for femur images) 3. IGD + Gini-SVM When texture features and various classifiers are combined, wrist fracture detection performance improves.
[19]	femur	- There are three classifications for fractures: healthy, fractured, and unknown. - To predict fractures, hierarchical SVM is combined with features like GO, MRF, and IGD, with the divide and conquer technique serving as the primary guiding principle.
[20]	femur	- The only feature for fracture detection is the adult femur's neck-shaft angle. - An adult femur is healthy if it has an NSA of 120 to 130 degrees; a fracture is identified if the NSA is less than 116 degrees. -Trabeculae are found in the upper extremity region of the femur, and when a fracture occurs, they significantly alter the orientation of the femur's neck and head. -The neck-shaft angle can identify these changes, but this method leaves local disruptions undetected.

[31]	tibia	The following feature-classifier combinations have been trained to find fractures in X-ray images of the tibia: 1. BPNN texture features 2. SVM-based texture features 3. NB texture features 4. BPNN + SVM + NB texture features (ensemble) where the GLCM Mean, GLCM Variance, Energy, Entropy, Homogeneity, GO, MRF, and IG are texture features A tibia fracture is present if 2 of 4 combinations do. Gradient analysis combined with a modified Hough transform is used to locate the fracture.
[33]	femur	GLCM is used to classify fractured and healthy cases with 86.67 percent accuracy.
[34]	tibia	The following feature-classifier combinations have been trained to find fractures in X-ray images of the tibia: 1. Texture features + ensemble (BPNN+SVM+NB) 2. Shape features + ensemble (BPNN+SVM+NB) 3. Shape features + texture features+ ensemble (BPNN+SVM+NB) Best combination
[35]	Hand bones	- To identify bone fractures in hand X-rays, the base classifiers NB, DT, NN, and BN are chosen. - Wavelet, curvelet, and GLCM feature set performances of the individual and ensemble classifiers are reported. -When wavelet features are used independently with an NB classifier and combined feature sets, the accuracy is at its highest.
[36]	long bones	- DT, SVM, NB, and NN classifiers were selected to train the fracture detection and classification model. - For both binary and multiclass classification tasks, SVM outperformed all other classifiers with an accuracy rate of more than 85% when using the ten-fold cross-validation technique.
[37]	tibia	- DT and KNN classifiers are used for fracture detection and classification respectively.
[58]	multiple bones	- Fracture detection and classification are performed using the DT and KNN classifiers, respectively. - A multi-layer classifier with different random forests on each layer is used to detect the fracture.

		- The Efficient Subwindow Search (ESS) algorithm is used to find the fracture. 81% of the detection rate is contained in the top seven bounding boxes produced due to fracture localization. -The proposed model performs better at locating and detecting fractures in bone X-ray images than SVM and a single layer of stacked random forest.

2.3 Deep learning-based algorithms for fracture detection

This section discusses deep learning-based algorithms developed for bone fracture detection. Before that, the next subsections introduce deep learning and explore various architectures developed from early to advance deep CNNs.

2.3.1 Deep learning

Deep learning, a subset of machine learning and Artificial Intelligence (AI), is the process of continually training data to produce predictions as depicted in Figure 2.4. These trained models can autonomously pick up new skills, improve with practice, and make predictions about unknowable facts [26]. In a machine learning technique, finding key features that show anomalies or patterns in the data is crucial. These features are often created primarily with human experience, but models can now automatically learn these properties as machine learning techniques develop.

There are two basic groups within which different DCNNs fall. Convolutional and Fully Connected (FC) layers are the first layers in traditional architectures. Down-sampling layers are the most current structural variant. The former comprises networks like AlexNet [60] and VGGNet [61], while the latter mostly contains more modern systems like GoogLeNet [62] and ResNet [63]. The primary difference between these two types of networks is that more current networks typically adopt some unique network topologies, such as inception in GoogLeNet or residual blocks in ResNet, and substitute the FC layers with an average global pooling layer [63, 64]. More thorough explanations of these networks will be provided in the subsequent subsections, followed by a discussion of sophisticated DCNN models and how fracture detection have used these models.

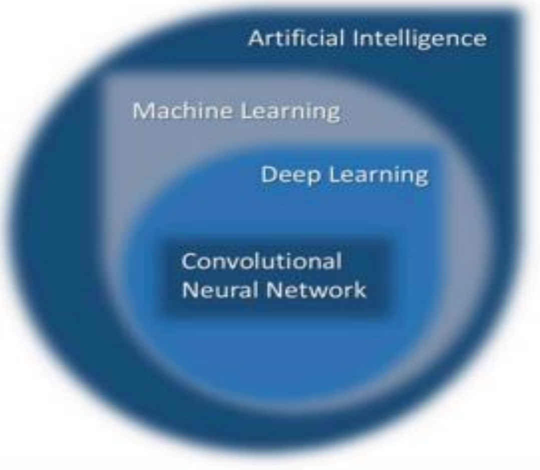


Figure 2.4 Deep learning is a subset of machine learning, which is a subset of artificial learning, and it is capable of performing tasks that require human intelligence [95].

2.3.1.1 Early deep CNN$_s$

A CNN or ConvNet is a unique, multilayered neural network created specifically for pattern recognition that allows it to identify visual patterns straight from pixel pictures with little to no pre-processing. A sizable visual database created for use in image classification and object detection was made available by the ImageNet project [65]. In order to promote the development and assessment of cutting-edge algorithms, this project also ran the ImageNet Large Scale Visual Recognition Competition (ILSVRC), an annual software competition [65The revolutionary CNN architecture LeNet-5 is presented in this section, followed by discussions of the leading CNN architectures of the ILSVRC: AlexNet, Network in Network (NIN), VGGNet, GoogLeNet, and ResNet. In this thesis, the collection of specified CNN architectures is referred to as L-A-N-V-G-R.

a. **LeNet-5 (1998)** - Comparing conventional architecture to traditional neural networks has resulted in a series of advancements in image classification. LeNet-5 [66], the first CNN model released in 1998, had seven layers, only three of which were convolutional (C) and one of which was Fully Convolutional (FC), with a total of 60,000 parameters. In Figure 2.4, this network is displayed.

The output of this network is a digit between 0 and 9, which is used to classify and identify 32 x 32-pixel greyscale handwritten numerals.

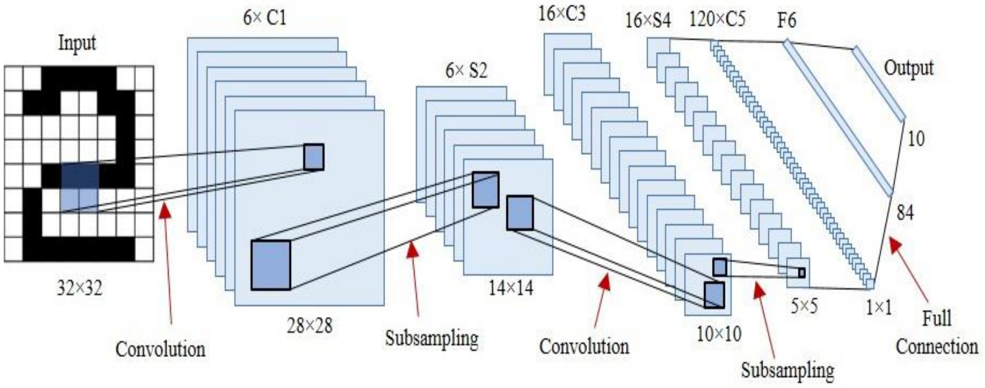


Figure 2.5 LeNet-5 architecture consisting of 7 layers [66].

b. **AlexNet (2012)** - Higher resolution images need to be processed using larger convolutional layers. Thus, AlexNet, which had 60 million characteristics in five convolution layers and three fully-connected layers, is credited with starting the background of deep learning [60]. Figure 2.5 depicts the AlexNet architecture. The reasonably quick and simple AlexNet is slightly changed into ZF-Net [67]. This network performed substantially better than its predecessors [60, 66]. In a conventional classification network, AlexNet has been applied after downsizing the input image and applying convolutional and FC layers. The output would then be the expected class label for the input image.

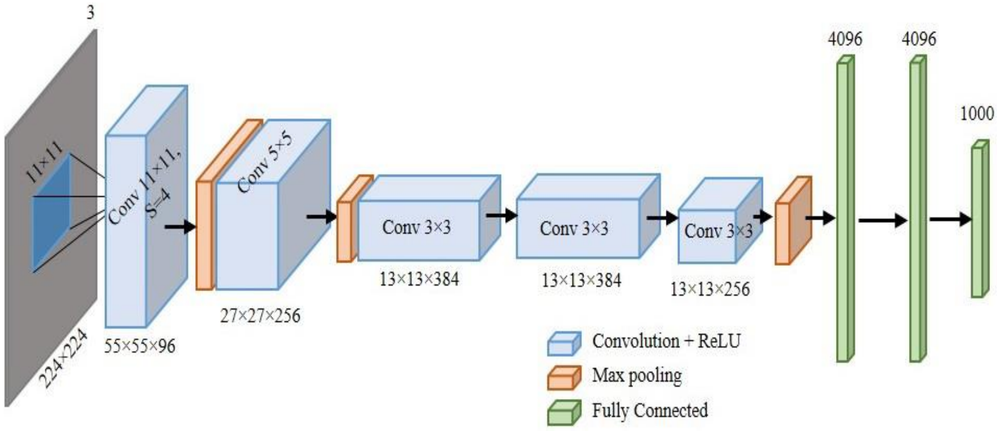


Figure 2.6 AlexNet architecture [60].

c. **NIN (2013)** - The capacity to distinguish between local patches within the input patch was improved by a Network in Network (NIN) design [64]. Three micro neural networks, essentially a nonlinear function approximator, are stacked to generate this model. The Multilayer Perceptron (MLP) is used to create the tiny neural networks. As shown in Figure 2.6, the filter size for each layer of the MLP structure is 1x1, except for the first layer.

Like CNN, the micro-networks are slid over the input to produce the feature maps, which are then supplied into the following layer. Multiple MLP structures are stacked to provide deep NIN, while the classification layer uses global average pooling.

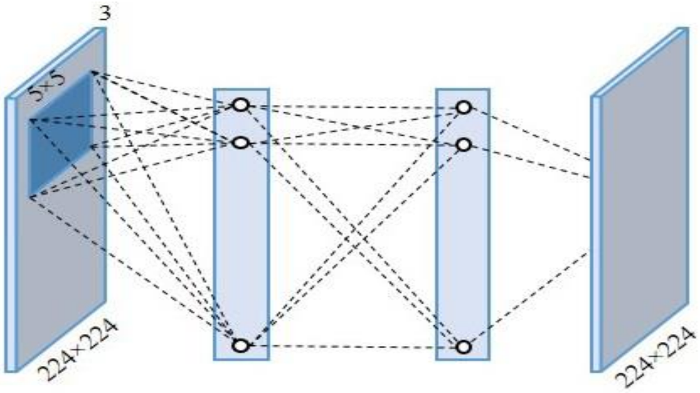


Figure 2.7 MLP structure [64].

d. **VGGNet (2014)** - This network's primary contribution is to assess correctness through deepening the network. Mini batch gradient descent with speed and dropout was used to increase the classification accuracy of this network, which had up to 19 layers and 138 million parameters [61]. Six VGGNet configurations have been proposed, ranging from 11 weight layers (eight convolution and three fully linked layers) to 19 weight layers (with 16 convolution and three fully connected layers). The total number of filters (depth of each layer) reaches 512 after starting with 64 in the first layer and growing by a factor of two after each max-pooling layer. Figure 2.7 depicts the VGGNet-16 design. Due to its extremely homogeneous design, VGGNet placed first in the single-object localization test at ILSVRC2014 [65].

e. **GoogLeNet (2015)** - The first section of the GoogLeNet design is similar to LeNet (Figure 2.4) and AlexNet (Figure 2.5), as shown in Figure 2.10, while the block's stack is derived from VGGNet (Figure 2.7). LeNet, AlexNet, and VGGNet's stack of FC layers are swapped out for GoogLeNet's worldwide mean pooling at the network's end. Google's top-5 error rate was 6.67%, which is quite near the level of human performance. It won first place in the ILSVRC2014's classification and detection task [65]. The subsequent adoption of Batch Normalization (BN) speeds up the training process for GoogleNet [69]. Figure 2.9 shows the GooLeNet model with 22 layers.

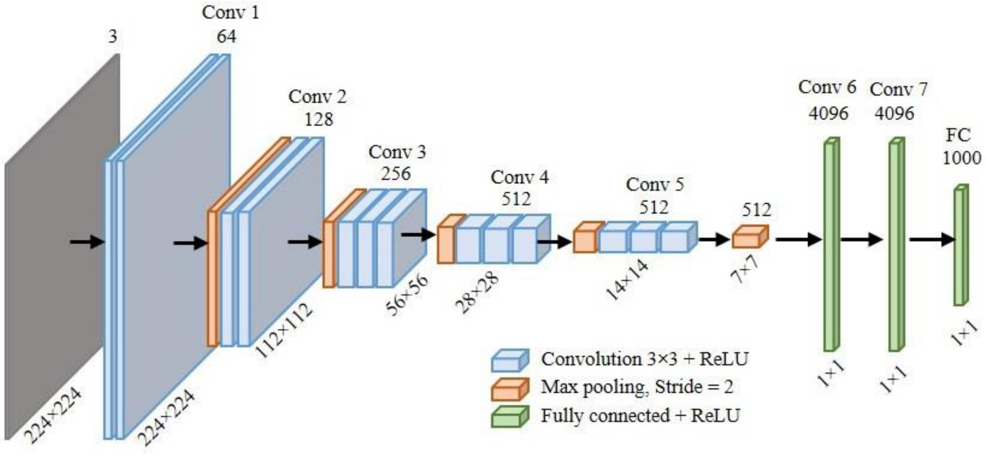


Figure 2.8 VGGNet-16 architecture [61].

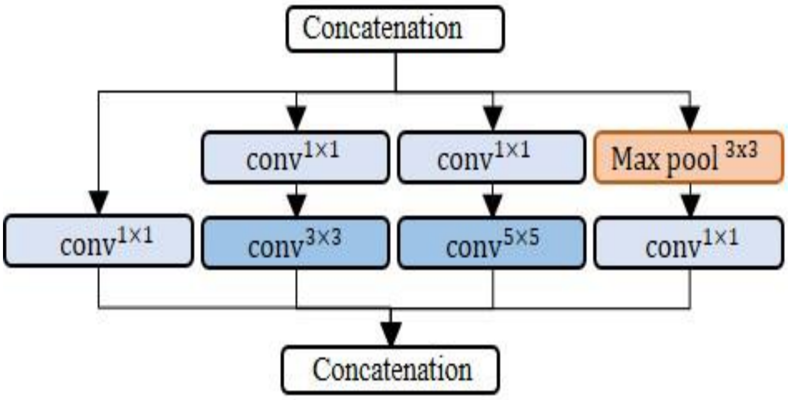


Figure 2.9 Inception module architecture [62].

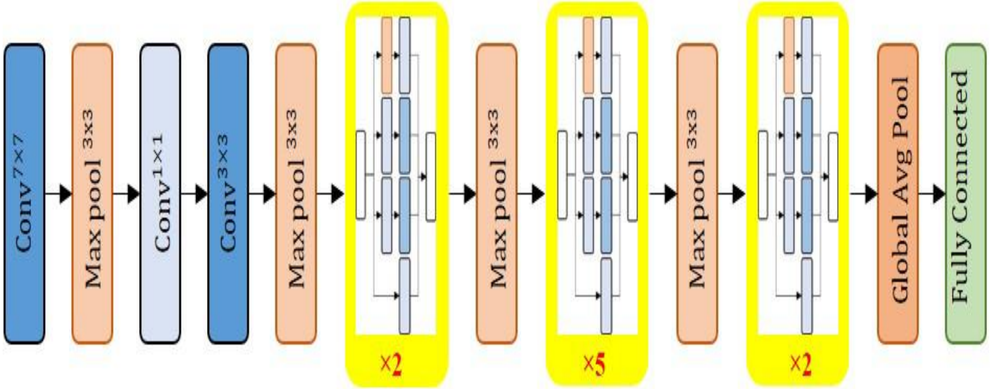


Figure 2.10 22-layer GoogleNet architecture [68].

f. **ResNet (2016)** - Since it is more difficult to train deeper neural networks than shallower ones, the development of ResNet marked the start of a new phase in deep neural network training efficiency [63]. In order to facilitate training and optimize the significantly deeper networks, which produced greater accuracy, a residual learning system was developed. Instead of learning unsourced functions, the layers were deliberately reformed to learn residual operations concerning the layer inputs. The introduction of the ResNet Unit (RU), shown in Figure 2.11, was made to address the critical issue [70]. This occurs when adding more layers to a powerful deep model causes the training error to increase. By creating the shortcut interconnection as identity mapping, the

ResNet solved this issue. The depth of the residual networks might range from 18, 34, 50, 101, or 152 layers. The most complex ResNet is less complex while being eight times larger than VGGNet. This network demonstrated easier optimization than VGGNet while achieving an increase in an object accuracy rate of 28% [62]. In Figure 2.12, the ResNet with a 34-layer residual is displayed. This network has four building blocks, and each has a stacking of RU building blocks.

ResNet-34 consists of 18 RU building components in total. Comparing the VGGNet to AlexNet, which has nearly three times as few parameters, involves much processing. Compared to AlexNet, which has over 60 million parameters, GoogleLeNet's Inception architecture has about 7 million parameters, which is a 9-times reduction. The ability to transport gradients back across all levels in an efficient manner is a worry, though, considering the relatively enormous depth of Google Net's 22 layers. The great performance of shorter networks in this task leads to the conclusion that the features generated by the middle layers of the network should be highly discriminative, which might be used by connecting auxiliary classifiers to the intermediate levels [62]. A deeper system would produce the same classification error as its shallower counterpart using ResNet's shortcut identity mappings [69]. By employing this method, networks containing the Inception module can achieve comparable accuracy while being less expensive [62].

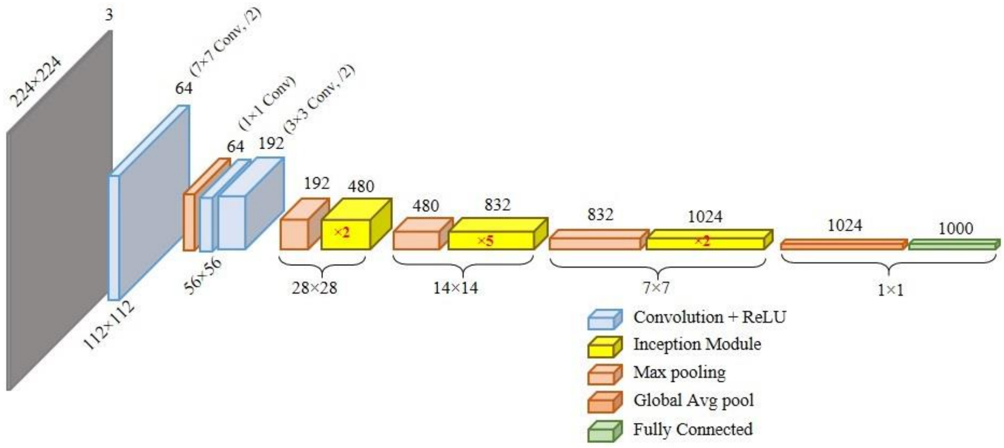


Figure 2.11 The architecture of GoogLeNet [62].

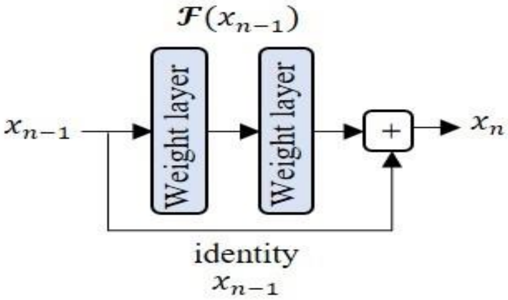


Figure 2.12 A ResNet Unit (RU) [63].

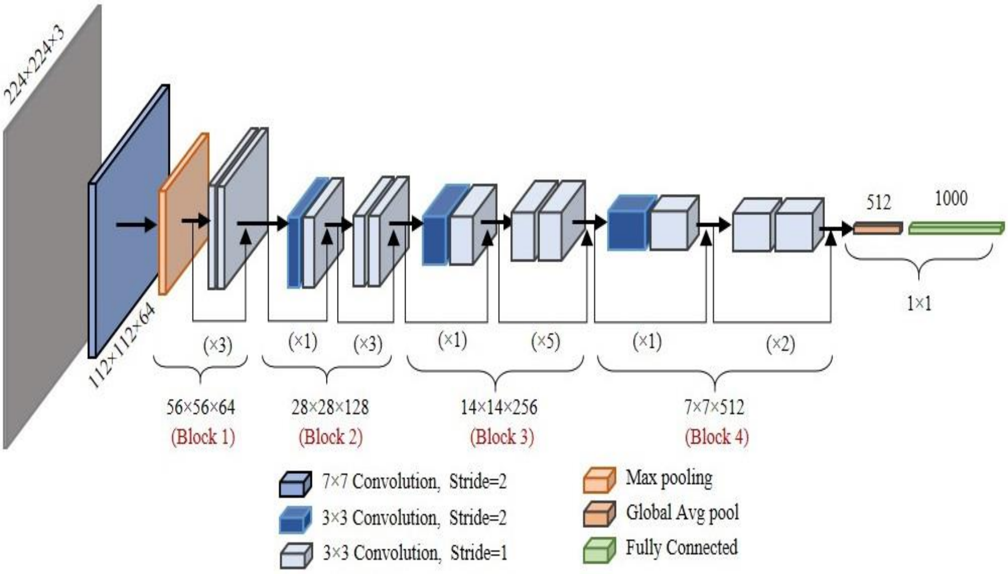


Figure 2.13 A 32-layer ResNet architecture [63].

2.3.1.2 Advanced deep CNNs

More sophisticated DCNN architectures have adapted the basic L-A-N-V-G-R networks for various purposes. Following is a list of several of these advanced DCNNs:

a) Object detection

RCNN, a region-based technique using CNN characteristics, was proposed by Uijlings et al. [70]. The ConvNet structure of AlexNet [60] is swapped out for a 16-layer GoogLeNet [62] model to build this architecture, resulting in a straightforward and scalable object detection technique. To accurately identify human faces, Taigman et al. suggested the nine-layer DeepFace CNN model [71].

With a network known as DeepID-Net, Ouyang et al. [72] tackled a specialized identification problem for distorted objects. To aid in understanding the distortion of object pieces, this framework provides a deformation-limited pooling layer. Although this method is based on RCNN, it is significantly more complicated because the deformation is specified as the visual features at many semantic levels. A subsequent method for modeling transformation matrices was published by Dai et al. [73]. Liu et al. proposed the Single Shot Detector (SSD) object identification model, which included predictions from several feature maps with different resolutions to recognize objects of various sizes. SSD is much faster than RCNN because it eliminates proposal development and integrates coordinates regression and region classification into a single network [74].

Wang et al. recommended the Fully Convolutional Network Based Tracker (FCNT) to address the visual tracking problem [75]. FCNT is a tracker network built on FCN that focuses on high-level features to recognize the semantic class of the object and low-level characteristics to acquire more exclusionary data to more effectively distinguish the same appearance from the background.

b) Classification

Instead of providing supervision solely at the output nodes and transmitting this supervision back to earlier levels, Lee et al. suggested Deeply-Supervised Nets (DSN) to give a close integrative oversight of the hidden layers [76]. They applied the auxiliary classifier to each buried layer, regarded as an additional regularizer. However, Szegedy et al. [62] had already introduced the importance of the auxiliary classifiers. A highly difficult job of fine-grained recognition to differentiate among visually very similar things such as kinds of birds, breeds of dogs, or types of airplanes, was handled by the DeCAF network presented by Donahue et al. [77]. Classification tasks like fine-grained recognition have substantial intra-class and low inter-class variation [77, 78]. Although introducing a residual learning framework with 152 levels made it easier to train deeper networks, the high computing cost of deeper neural networks still makes them difficult to deploy. At that point, the two main issues that need to be handled are disappearing gradient and model size. Using a feed-forward ResNet technique, Huang et al. [79] addressed the gradient vanishing issue by connecting every layer to every other. Their model, DenseNets, also decreased the number of variables. Both ResNet and DenseNet's designs fall under the categories of pre-activation and cross-layer connections. A batch normalization layer follows the convolutional layer in these networks, and the output of one layer can be utilized as the input for numerous following layers. These two well-known deep learning networks were developed to recognize various classes, including the 1000 classes in ImageNet.

c) Pixel Classification

By fusing several low-level image data with high-level context, Girshick et al. suggested an object recognition and semantic segmentation network [80]. This network uses bottom-up region recommendations in conjunction with CNNs to localize objects and segment them.

Another deep neural network, dubbed DeepLab, which enhances the localization of object boundaries, also addresses semantic segmentation [81]. This model incorporates two new elements: the Atrous Spatial Pyramid Pooling (ASPP) module to partition the objects at various scales and the Atrus convolution, a potent tool for controlling the resolution in dense prediction. The Full Resolution Residual Networks (FRNN) model, another DCNN-based model for semantic segmentation, improves localization accuracy while offering remarkable recognition performance [82].

Most DCNNs have excessive parameters and need millions or even billions of starting point operations. Therefore, deep network designers' primary concerns are storage and computing capacity. One of the primary drivers for reducing the number of these networks' parameters is to increase the effectiveness of their deployment on mobile apps like MobileNet [83] or their training in Internet-scale clusters, which results in lower computing costs and storage requirements.

An overview of dimension reduction methods used with deep networks is provided in the next section.

2.3.1.3 Dimensionally reduced deep CNNs

Although the deep networks have dramatically increased accuracy, there is a significant processing overhead due to the deep networks' enormous number of parameters. Implementing a deep network on hardware systems with constrained processing resources, such as mobile phones, is challenging because of the high storage requirements and computationally expensive floating-point matrix multiplications. Considering ways to lower the memory and compute costs in deep network topologies is crucial.

In order to speed up just the testing phase of the large-scale training network, Denton et al. devised a linear compression algorithm [84]. This method cuts the test time two-fold by taking advantage of CNNs' linear nature. The RCNN object detection network requires a lot of computing power. Two improved versions of this network are developed to increase its efficiency: the first, fast-RCNN, extracts the Region of Interest (RoI) [85]; the second, faster-RCNN, builds the Region Proposal Networks (RPNs) on top of the RCNN convolutional feature mappings [86]. FitNets is a framework that Romero et al. created to condense a wide, deep network with many parameters into a deeper, thinner network with fewer parameters [87]. The compressed network is trained using the intermediate-level cues from the larger network. FitNets has shown that deeper, narrower networks can generalize and operate more quickly than wider ones.

Han et al. established a parameter reduction technique to minimize computational time and memory consumption in CNNs by deleting extraneous links in the first round of learning and then fine-tuning only the critical connections in the second [88]. This method maintained the accuracy of

AlexNet, which had nine times fewer parameters, and VGG-16, which had 13 times fewer parameters. By removing the less significant filters, ThiNet used filter level pruning as another optimization strategy [89]. Instead of using the statistics from the current layer, ThiNet prunes the filters depending on the statistics from the next layer. It reduced the VGGNet-16 model's size by a factor of 16 while just slightly decreasing accuracy. Landola et al. SqueezeNet is a different compressed version of AlexNet that keeps accuracy while having 50 times fewer parameters [90]. The re-module, which this technique introduced, has two different sorts of layers: the compress convolution and the expansion. In a different architecture known as Deep Fried Convnets, the fully connected layers are re-parametrized using an adaptive Fastfood transform algorithm because they contain high and over 90% of the CNN parameters [91].

Artificial Neural Networks (ANN) are a commonly used computational model in machine learning for identifying intricate patterns in the data. These are brain-inspired systems that think about replicating how people learn. Although perceptrons, also known as neural networks, have been around since the 1940s, they have only just begun to play a significant role in artificial intelligence. The development of a method known as "backpropagation" is one reason they have grown prominent in machine learning. With backpropagation, neural networks can modify the weights of their hidden layer neurons to produce the desired results [93].

In radiology, professional radiologists can extract important details from scans and evaluate them using their skills, knowledge, and experience. As a result, it presents a fantastic opportunity to use machine learning algorithms to automatically forecast the data with accuracy comparable to that of a radiologist expert [92]. For bone fracture identification, the Intelligent Bone Fracture Detection System (IBFDS) integrates image processing and neural network approaches [32]. First, a feature extraction technique based on the Scale Invariant Fourier Transform (SIFT) algorithm [94] is used to improve the image quality by minimizing noise in the X-ray images. The retrieved characteristics are then used to classify images into the fracture and non-fracture categories using a 3-layer backpropagation ANN.

Artificial Neural Networks (ANNs), which have numerous hidden layers and offer higher degrees of abstraction, have advanced thanks to deep learning. By adding deep layers to the model that enable the system to learn complex data, deep neural networks have significantly increased the accuracy of task prediction [96]. Many factors, including the availability of large datasets made possible by the rapid accumulation of electronic data in the form of Electronic Medical Records (EMRs), Graphics Processing Unit (GPU) advancements that improved performance with graphics and videos, and advancements in the deep learning algorithm made possible by the incorporation of multiple layers in deep learning architecture, are driving the growth of deep learning in the healthcare sector [97].

Convolutional Neural Networks (CNN) is a type of deep learning architecture that includes fully linked, sub-sampling, and convolution layers, as shown in Figure 2.14. When the CNN model was created, feature learning approaches radically changed since it can *learn the features* rather than manually program them into the system. While the fully connected layer is utilized for classification, CNN's convolution and subsampling layers are used for feature learning. It has achieved enormous success in several areas of medical image analysis, including image segmentation, image registration, image fusion, image annotation, genomics, etc. It has recently emerged as a machine learning breakthrough.

To interpret radiographic data, radiologists combine sense, memory, pattern recognition, and cognitive thinking. Their performance is impacted by numerous distractions, which ultimately increases workloads and weariness. Therefore, improving patient security through developing systems or technologies automatically identify abnormalities or patterns in musculoskeletal radiographs without human intervention [98]. The AlexNet [60] model, created by Alex Krizhevsky et al. in 2012, sparked a revolution in computer vision and provided deep CNN with fresh knowledge. It was first created to participate in the ImageNet competition, whose overall architecture is comparable to LeNet-5 [66] but far larger. After first placing in the ImageNet competition, it effectively persuaded the computer vision community of its value. This revolution was made possible by the employment of efficient regularization parameters, data augmentation techniques, rectified linear units, and the usage of graphics processing units to meet computing requirements. It was listed among the top ten deep learning milestones of 2013 [99]. The greatest strength of a CNN is its deep design, which enables the extraction of fine characteristics at various abstraction layers [100].

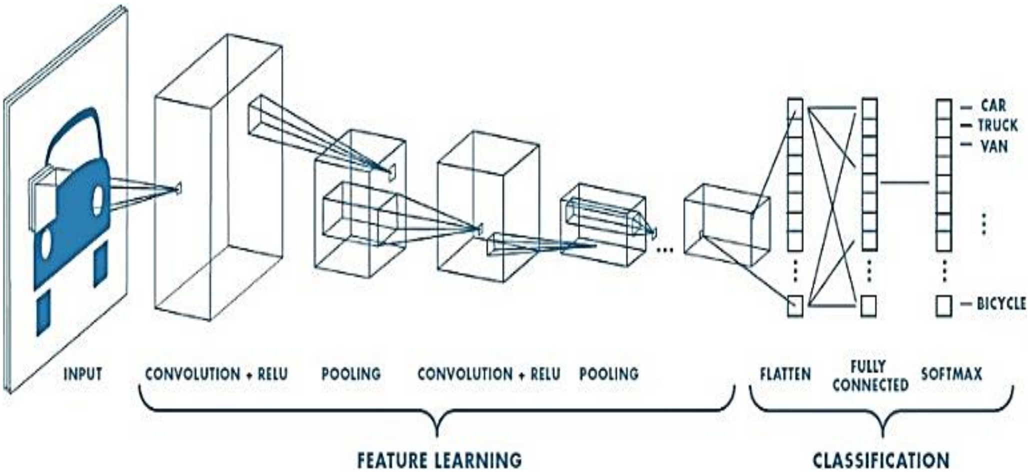


Figure 2.14 A schematic representation of the CNN architecture is shown. Convolution, subsampling, and fully connected layers make up its three layers. Convolution and subsampling layers are used to classify data, while fully connected layers are used to learn features [101]

2.4 Deep learning-based algorithms for fracture classification and localization

A Deep Convolutional Neural Network (DCNN) needs training data to be properly trained. When there is not enough of it, it might be challenging to guarantee appropriate convergence of the model, especially in the highly-restricted field of medical imaging. Data augmentation, which reduces data insufficiency and overfitting issues, is the act of generating new data from our existing dataset with tiny adjustments such as flips, rotations, mirroring, translations, etc. Data scarcity problems, which are the main obstacle to deep neural network architectures, are mitigated by CNN architectures' capacity to identify and categorize fractures even when they are oriented differently. From 2003 to 2018, the number of radiographs used to diagnose fractures increased from 500 to hundreds and thousands due to data augmentation approaches. Only a few published publications (included in Table 2.3) on fracture diagnosis have used augmentation methods on their datasets. Future work could successfully implement this strategy to improve the classifier's performance. A CNN that has already been trained on a separate network can be fine-tuned, another possible solution to the data scarcity issue.

Many researchers have been inspired to use transfer-learning techniques because of the dearth of datasets, particularly in the medical field [105]. Pre-trained CNN is a state-of-the-art image classification network trained on millions of images from a specific domain over many weeks on various servers before being applied to a different area of interest. This strategy has proven helpful for researchers when a lack of resources makes it difficult to build a viable model from the start. The very sophisticated and potent set of characteristics required for the relevant domain of interest can be obtained using these huge pre-trained models [102]. Two different CNN models are compared in a study made by Birks et al. [99]. One model is trained entirely from scratch. The other is a pre-trained model that has been further adjusted on the necessary domain. Different medical imaging applications, such as classification, detection, and segmentation, are used to compare the performance of the two models. In the best situation, a tweaked CNN model effectively outperformed the CNN model trained from scratch, and in the worst case, it performed similarly to the CNN model. Therefore, based on the quantity of data available, fine-tuning an existing model might present a useful technique to achieve the greatest performance for the application. The pertinent review findings of the deep learning-based fracture detection systems are shown in Table 2.4.

Table 2.3 Researchers' methods for data augmentation in bone fracture detection are shown.

Author	Images before augmentation	Images after augmentation	Augmentation techniques
[23], 2017	Not available	45,492	Translation, rotation, histogram matching, shearing
[24], 2018	Not available	31,590	cropping, horizontal mirroring, rotation, lighting and contrast adjustment
[25], 2018	1,891	>40,000	Shifting, scale transformations, rotation
[26], 2018	1,389	11,112	Horizontal flip, rotation, width and height shift, shearing, zooming
[63], 2017	Not available	40,561	Lateral inversion, rotation

Table 2.4 The pertinent review findings of the deep learning-based fracture detection systems are shown.

Ref	Target body region	Fracture detection type	Architecture used	Relevant review findings
[24]	31,490 wrist radiographs	Fracture is detected and heatmap is generated	Extension of U-Net architecture	Radiographs of ankles, knees, spines and other body parts totaling 100,855 were used for pretraining, and 31,490 radiographs were used for model fine-tuning.
[25]	1,891 humerus radiographs	Fracture classified into 4 categories	ResNet-152	-Network is fine-tuned on medical images. -The network's performance is evaluated against the labels produced by more than 50 senior orthopedic surgeons.
[27]	256,000 wrist, hand, and ankle radiographs	Fracture is classified into 4 classes: fracture, laterality, body part, and exam view.	5 state-of-the art deep neural networks	The network's performance is compared to the labels created by two senior orthopedic surgeons.
[28]	38 Distal radius fractures	Fracture is detected and localized	faster R-CNN	- For training, 4,476 augmented images are used. - Numerous augmented images were used to test the object detection precision.

[29]	7,356 Wrist radiographs	Fracture is detected and localized	faster R-CNN	The network's performance is compared to the labels created by two senior orthopedic surgeons.
[30]	1,946 Wrist radiographs	Fracture is detected and heatmap is generated	“DeepWrist” model is proposed.	- The model is assessed using two test sets: one for the general population and one difficult test set with only cases requiring confirmation by CT. - The radiograph is cropped to reveal the presence of the fracture at three landmark locations. - The ImageNet dataset is used to pre-train the model.

The recent improvements in deep learning methodologies and hardware processing have made object localization and identification intuitive. As a result, numerous industries have seen a rapid increase in the widespread adoption of object detection algorithms. However, the traditional methods of object detection using the sliding window approach produced thousands or even millions of bounding boxes, making them computationally expensive. By superimposing the image regions and comparing pixels one by one, the sliding window method [59] compares the images. Because of the overall complexity of the method, it is thought to be perplexing. If the source image is stretched, rotated, given different contrast levels, cropped, or zoomed using this technique, it also gets harder to compare.

A few earlier studies relied on the radiographs' binary classification to determine whether a fracture existed or not [23, 25, 26]. However, object detection models have been effectively used in recent years to identify the presence of fractures and their locations, providing better visual interpretability for healthcare professionals than binary classification. The wrist fractures are discovered and localized by extending the U-Net architecture, which recognizes the fracture and further localizes it by producing a heat map. On two different test sets, the model produced AUC values of 96.7% and 97.5%, respectively

[24]. Using faster-RCNN architecture, the radius and ulna fractures in wrist X-rays are identified, producing a bounding box and likelihood of fracture occurrences [28, 29].

Instance segmentation is a technique that uses a bounding box to mask the objects' actual shape rather than their location. These tasks combine localization, classification, and segmentation tasks into a single output that is a polygon mask encircling the defined target. Some of the most well-liked methodologies for performing instance segmentation include mask-RCNN [101], U-Net [104], and DeepLab [81]. Compared to object detection and image classification, instance segmentation demonstrates more promise and accuracy in tracking the defects/objects in the image. In addition to the current branch for forecasting an object mask and a bounding box, it tends to add a parallel branch. Instance segmentation, which combines faster-RCNN and semantic segmentation, has been used in this research. To our knowledge, this is the first study that focuses on fracture detection by creating a bounding box and segmentation mask around the fracture. We have determined that drawing a box only around the fracture does not accurately depict the structure of the fractures. Each image is labeled by drawing a more intricate shape, like a polygon, around the fracture.

2.5 Conclusion

The development of reliable fracture detection systems utilizing computer vision techniques has revolutionized due to the rise of deep learning. With little pre-processing, multi-layer neural networks can detect visual features directly from image pixels. Deep learning's primary benefit is that this does not require manually created features. As part of the classifier learning process, it instead performs automated and optimized feature extraction, which does not compromise the correctness of object recognition. According to the research, the deep learning-based method generates bone-fracture detection systems with high accuracy.

Current research priorities include object identification, localization, and other computer vision tasks like image classification and object categorization. Since errors might occur during manual interpretation of radiographs, automated inspection is often necessary for quality or defect evaluation. Automated fracture identification has numerous advantages over manual inspection, which is prone to human mistakes, complexity, time, and cost. Intensity, color, scale-invariant features, and other contemporary feature extraction methods are frequently used for object detection applications. Without a plethora of prior knowledge, salient feature extraction helps find the salient target. This method can identify fractures in the apparent foreground and distinguish them from the background regions. Although the saliency strategy provides adequate separation between the fracture and the background, additional sophisticated methods are needed due to difficulties, including uneven lighting and unessential bone regions.

In contrast to machine learning models like SVM and PCA, which are primarily based on the texture and color of each patch, multi-layer deep convolutional neural networks such as ResNet, VGGNet, GoogleNet, and FCN have the potential to perceive a non-linear relationship between different variables and accomplish influential object detection and localization performance. A fracture detection model built on a semantic segmentation network learns from non-uniform and sophisticated bone regions to extract specific high-level information about cracks.

CHAPTER -3

DEVELOPMENT OF TWO NOVEL DATASETS FOR FRACTION LOCALIZATION AND SEGMENTATION

3.1 Introduction

Fracture detection is increasingly incorporating computer vision and image processing-based approaches to identify the fracture cracks/patterns in bone X-ray samples. Researchers in the existing articles have proved the performance of the suggested model in their private datasets [23-30]. However, one of the key limitations in comparing the existing systems is the absence of the freely available standard dataset. Most of the studies mentioned concentrate more on classification and detection than segmentation, i.e., these models can determine whether a radiograph is fractured or not and can pinpoint the exact locations of the fracture by predicting bounding boxes, but they cannot exactly predict the shape of the fractures. To our knowledge, there is not a sizable collection of uniform images of wrist fractures along with their pixel-by-pixel labels. We created a dataset of wrist fracture images with their labels, a part of which is now accessible online on the GitHub platform [118].

The availability of the labeled dataset is the main obstacle to achieving multi-label classification on various anatomical locations. A promising solution to this issue is to improve a CNN that has already been trained on a different network. These pre-trained models enable researchers to gain the very sophisticated and potent features required for the topic of interest. The model can be trained on numerous images rather than millions of non-radiology images. The researchers have pre-trained the model on non-medical datasets and fine-tuned the model on wrist radiographs via transfer learning approaches. Training the models on general-purpose state-of-the-art datasets such as COCO [119] or PASCAL VOC [120] for detecting bone abnormalities is of limited use.

Instead of training the model on a non-medical dataset, we have incorporated a surface crack dataset with similar crack patterns to wrist bone fractures (Figure 3.1). The proposed approach does not directly use transfer learning on the wrist fracture dataset. The wrist dataset was fine-tuned using a surface crack dataset, which had previously been fine-tuned using the state-of-the art COCO dataset. This process is pictorially represented in Figure 3.2 along with the description. We have used an extension of the faster-RCNN model (mask-RCNN) and customized it to localize and segment the fractures.

3.1.1 Dataset Preparation

We have created and labeled two distinct datasets. The Surface Crack Dataset is the first dataset, and the Wrist Fracture Dataset is the second dataset. The surface crack and wrist fracture datasets are referred to as SCD and WFD, respectively, in this thesis. We developed both datasets for research purposes and expanded the fracture detection task to fracture classification and segmentation. Table 3.1 presents a list of publicly accessible datasets labeled for classification issues where we can determine whether or not a fracture is visible in the images.

The rest of the chapter is divided into four stages of data preparation. A discussion of the novel surface crack and wrist fracture dataset is found in Section 3.2.1. Next, the images in the datasets are preprocessed before feeding them to the network. Afterward, the images are undergone a labeling process followed by augmentation. All the images in our datasets are manually labeled, which was time-consuming but less error-prone than the automatic annotation software. Finally, multiple copies of the labeled images are generated using several augmentation techniques.

3.1.2 Data collection

Data collection is the first step, which entails gathering datasets from different hospitals or the public domain, followed by dataset labeling. Before feeding a dataset into a classifier, it must be prepared to predict fracture occurrence, its labels, and segmented masks. The dataset used for the study comprises self-collected and labeled Surface Crack Dataset (SCD) and Wrist Fracture Dataset (WFD). Instead of pretraining the model on a non-medical dataset, we have incorporated a surface crack dataset with similar crack patterns to wrist bone fractures.

a) Surface Crack Dataset (SCD)

SCD consists of pictures taken from walls, pavements, and roads, created using a mobile camera. Real-time images are captured using a mobile camera that can include variations like obstacles, shadows, partially visible cracks, background clutter, holes, and surface roughness. We worked with 3,000 surface crack images (1044 pavements, 1045 walls, and 911 roads) that we labeled for the presence of cracks. The dataset is manually annotated for the two distinct tasks of object detection and instance segmentation. The dataset is labeled by tracing a bounding box around the areas of the image's crack. When using the instance segmentation technique, a segmented mask is created that delineates the edge of the polygonal crack. Figure. 3.3 shows an example of original and annotated images displaying the bounding box and segmented mask around the cracks.

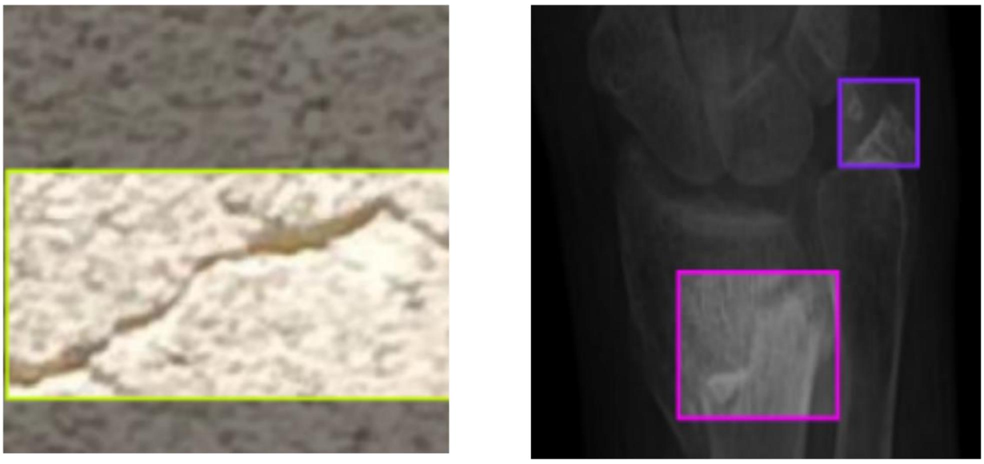


Figure 3.1 The crack patterns in the wrist fracture dataset and the surface crack dataset are comparable

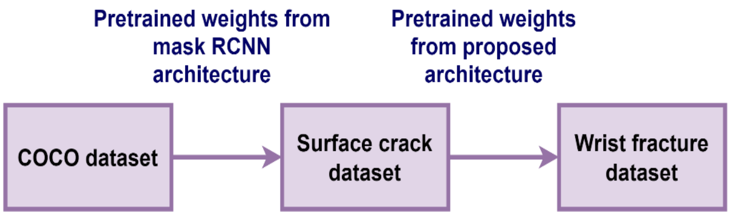

Pretrained weights from mask RCNN architecture
Pretrained weights from proposed architecture
COCO dataset
Surface crack dataset
Wrist fracture dataset

Figure 3.2 Knowledge transfer from COCO dataset to the wrist fracture dataset.

Table 3.1 A dataset of publicly accessible bone X-ray images that could be used to identify fractures.

S No.	Name	Source
1	Stanford ML group	MURA: For research purposes, a dataset of 40,561 bone X-ray images, including elbow, finger, hand, humerus, forearm, shoulder, and wrist, is available to the public. Six board-certified expert radiologists from Stanford Hospital categorized the dataset into normal and abnormal cases after labelling it. https://stanfordmlgroup.github.io/competitions/mura/
2	medpix	Medpix is an online database of medical images https://medpix.nlm.nih.gov/search?allen=true&allt=true&alli=true&query=fracture
3	Radiopaedia	More than 2800 fracture cases with diagnosis details are freely available. https://radiopaedia.org/search?lang=us&q=fracture&scope=cases
4	IIEST, Shibpur Indian Institute of Engineering Science and Technology	A diagnosis report compiled from case report forms is available, along with X-ray and MRI images of the knee joint. http://oldwww.iiests.ac.in/component/content/article/155-itcategory/3282-medical-image-database
5	MOST: Multicenter Osteoarthritis Study (MOST)	X-ray and MRI images of the knee joint are available, along with a diagnosis report compiled from case report forms. http://most.ucsf.edu/datadocs.asp
6	aylward.org	There are 10,000 chest X-ray images available, along with diagnosis information. https://nihcc.app.box.com/v/ChestXray-NIHCC/folder/37178474737

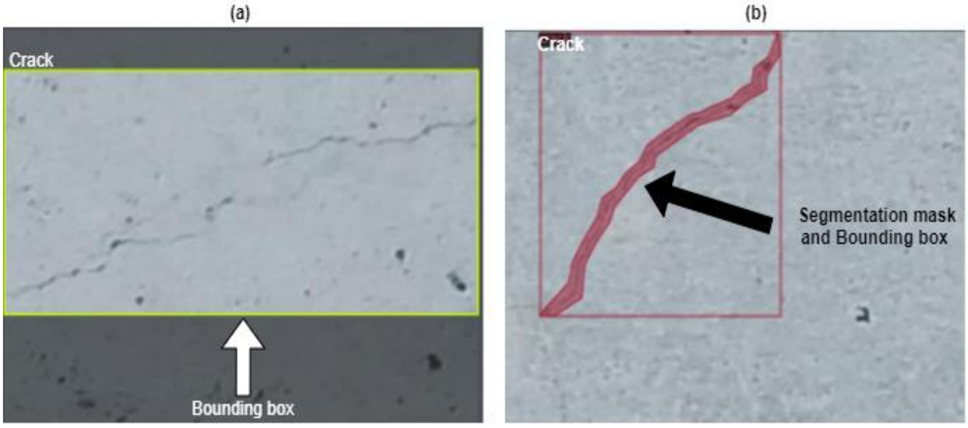


Figure 3.3 (a) A sample of the labelled image with a bounding box.
(b) An illustration of an input image that has been labelled by drawing a segmented mask covering only the critical crack regions.

b) Wrist Fracture Dataset (WFD)

Between February 2019 and March 2020, the radiographs are collected from the Government Doon Medical Hospital in Dehradun, India. Under the Ethical Conduct in Human Research and Related Activities Regulations, the dataset was obtained without revealing the participant's identity or any of their demographic information. The wrist bone X-ray images in the dataset totaled 315 images (296 Distal Radius, 19 Ulna).

The radiographs were disregarded if a plaster cast was in place, the wrist's growth plates had not yet fused, or the study revealed any fracture other than a distal radius or ulna fracture. Additionally, images were discarded if a single lateral projection could not determine whether a fracture existed or not. A radiology specialist labeled the radiographs with more than ten years of experience who serves as the Head of Orthopedics at Doon Hospital, Dehradun, India. Two additional Radiologists have been invited to participate in the verification of the annotations generated on the images. It was noted that 14 radiographs were found in disagreement regarding the diagnosis of the fractures. As a result, the dataset was updated to remove these 14 radiographs. 315 labeled images were finally chosen for training, where 210 wrist radiographs showed fracture occurrences and 105 wrist radiographs showed no fracture, yielding an initial data set. The summary of the dataset used in the current work is mentioned in Table 3.2.

Table 3.2 Summary of the datasets used.

Dataset Type	**No. of images**	**Dataset source**	**Type**	**No. of cracks**	**Labelling technique used**
COCO dataset	3,30,000	[119]	NA	NA	NA
Surface Crack Dataset (SCD)	3,000	Collected by capturing surface crack images using mobile camera	pavements-1044, walls-1055, and roads-911	8,241	We manually label all the images, creating a bounding box and segmentation mask around the surface crack.
Wrist Fracture Dataset (WFD)	315	Collected from Hospitals	Distal Radius-296, Ulna- 19	733	An orthopedic surgeon labels all the images by creating a bounding box and segmentation mask around the wrist crack.

3.1.3 Data Preprocessing

We have performed two separate sets of operations for SCD and WFD datasets, the details of which are mentioned in Table 3.3. The finger bone area was removed by the radiologists when cropping the region of interest from wrist X-ray samples. Next, we converted the DICOM images to 24-bit lossless JPEG format while making sure the best windowing was chosen under the doctor's supervision. Next, the images are undergone a labeling process followed by augmentation. The SCD had not undergone any preprocessing steps because the captured images were already in JPEG format.

The wrist fracture and surface crack datasets were provided as the subjects of our experiments. For research purposes, we have created and annotated both datasets and expanded the fracture detection task to include fracture localization and segmentation. Table 3.4 displays the dataset's origin, split ratio, types of data gathered, and annotation process used to categorize the datasets in the existing research articles.

Table 3.3 Preprocessing techniques applied on the datasets used.

Dataset	No. of images	Preprocessing techniques		
COCO dataset	3,30,000	The dataset is not processed. Only the pre-trained weights are obtained.		
SCD	3,000	No cropping required	No conversion is required since the images are already in JPEG format	Images after augmentation- 21,000 14,700 (Train), 4200 (validation), 2100 (Test)
WFD	315	Finger bone area and extra annotations are removed	DICOM images are converted to 24-bit lossless JPEG format	Images after augmentation- 2,205 1543 (Train), 441 (validation), 221 (Test)

Table 3.4 The table displays the dataset's origin, split ratio, types of data gathered, and annotation process used to categories the datasets.

Ref	Bone type	No. of images	Dataset split ratio	Hospital Name	Fracture prevalence	Annotation process
[20], 2003	femur	446	Training set- 126 Test set- 320	Local hospital, Singapore	Not available	Doctors primarily used the neck-shaft angle of the femur as a diagnostic tool.
[16], 2004	femur	432	Training set- 324 Test set- 108	Local hospital, Singapore	12% in training and test set	Not discussed
[17], 2004	femur	432	Training set- 324 Test set- 108	Local hospital, Singapore	12% in training and test set	Not discussed
	wrist	145	Training set- 71			

			Test set- 74			
[18], 2005	femur	432	Training set 324 Test set- 108	Local hospital, Singapore	12% in training and test set	Not discussed
	wrist	145	Training set- 71 Test set- 74		30% in training and test set	
[19], 2007	femur	420	Training set- 200 Validation set- 160 Test set- 60	Local hospital, Singapore	12% in training , validation and test set	Not discussed
[23], 2011	femur	30	Not available	Biomedical research center, Malaysia	50% in input images	Not discussed
[31], 2011	tibia	1650	Training set- 650 Test set- 1000	Not available	60% in training and 62% in test set	Not discussed
[34], 2012	tibia	NA	Not available	Not available	Not available	Not discussed
[35], 2013	hand	98	Training set- 116 Test set- 29	Jordanian Royal medical services, Jordan, Public domain	50% in input images	Radiology reports provide annotations for the X-ray samples.
[36], 2013	long bones	300	Not available	Hashemite kingdom of Jordan, public domain	33% in training and test set	Medical professionals who have been consulted verify 300 images and their labels.
[58], 2015	multiple	145	Training set- 116	Public domain	Not available	Radiology reports provide annotations

			Test set- 29			for the X-ray samples.
[23], 2017	pelvis	Dataset collected from hospital is not available, dataset developed after augmentation is 53,278	Training set- 45,492 Validation set- 4,432 Test set- 3,354	Royal Adelaide Hospital, SA	12% in training and validation set, 19% in test set	- Fracture labels from radiographs are gathered from reports from the orthopedic surgical unit and radiology departments. - Only 7.4% of the dataset needs to be manually labelled, which reduces the need for manual labelling of the entire dataset.
[32], 2017	multiple	100	Training set- 30 Test set- 70	Orthopedics traumatology hospital, Turkey	Not available	A benchmark database is used to acquire 100 images.
[24], 2018	wrist	Dataset from the hospital was not available, 31,590 were created after augmentation.	Training set- 28,341 Validation set- 3,149 Test set 1- 3,500 Test set 2- 1,400	HSS (Hospital for Special Surgery), United States	Not available	A group of 18 senior orthopedic surgeons manually annotated 1,35,409 radiographs, of which 1,00,855 were bone images of 11 different body parts and 34,990 were wrist images.
[25], 2018	humerus	1,891	Training set- 40,000 Test set- 181	Multiple Hospitals, Korea	73% in training, validation and test set	1,891 images gathered from various hospitals in Korea are divided into 4 categories by 3 experts.
[26], 2018	wrist	1,389	Training set- 8,890	Royal Devon & Exeter	F~50% in training,	- A trained radiologist was responsible for

			Validation set- 1111 Test set- 1111	Hospital, UK	validation and test set	selecting the appropriate region of interest and converting the images into JPEG format. - Using radiological reports, wrist X-ray images are categorized into fractured and healthy categories and are confirmed by a radiology registrar with three years of experience.
[37], 2018	tibia	NA	Training set- 40 for detection Test set - 52 for classification	Yangon Orthopedic Hospital, Myanmar , radiology websites	Not available	Not available

3.1.4 Data Labelling

The labeling process, a crucial step in data pre-processing, comes second in importance after the dataset collection. Labeling requires annotation of radiographs by experienced radiologists, clinicians, or orthopedic surgeons that should be done with extreme care; otherwise, it will reflect poor dataset quality and might reduce the overall performance of the model [121]. Making a complete dataset for any classification activity is difficult because of who is in charge of labeling and how long it takes him or her to do it. If the dataset is correctly mapped with exceptional care and precision by a team of experts, a classification-based algorithm can accurately predict the outcome. Even though every industry faces governance and regulation challenges in data collection, data management, and labeling that could take several months to complete, the challenges facing the healthcare sector are unique due to the complexity of the data and the extremely strict regulations. To address these issues, a health institute may request a waiver of consent from an Institutional Review Board (IRB) study, or researchers may process and anonymize DICOM data to remove any patient health information.

The radiologists have utilized LabelMe [122] software to annotate the wrist bone images. The skilled radiologist labels the wrist fractures by tracing a box around the fracture. Drawing a bounding box may not accurately depict the shape of the fractures, as it involves non-essential bone areas when the model is being trained. In order to further label each image, a mask is made by drawing a more intricate shape, such as a polygon, around the crack. It locates the fracture in the image, detects it, and then builds a segmented mask around it. Figure 3.4 illustrates an example of labeled images, including bounding boxes and segmentation masks. It can be observed that the segmentation mask constructed using the red color improves the fracture patterns' visibility, which is used to segment the fractures. A portion of the dataset is made publicly available for research to circumvent data collection challenges and wrist fracture labeling [118].

3.1.5 Data Augmentation

The data was amplified using a data augmentation approach. Several non-exact copies or transformations of each image had to be made to accomplish this, including the salient features in various orientations, which aided in providing the CNN with more training examples. The goal was to more accurately represent the wrist radiograph population in the real world so that variables like limb size, handedness, and small differences in wrist positioning could be better considered. The images are amplified in numbers using the data augmentation technique, which employs the transformations selected for SCD and WFD datasets (Table 3.5). Finally, the augmented images are divided into three parts (train, validation, and test set), having a split ratio of 70%, 20%, and 10% for both datasets. An illustration of the outcomes of augmentation on a source image from SCD and WFD is shown in Figure 3.6 and Figure 3.7, respectively.

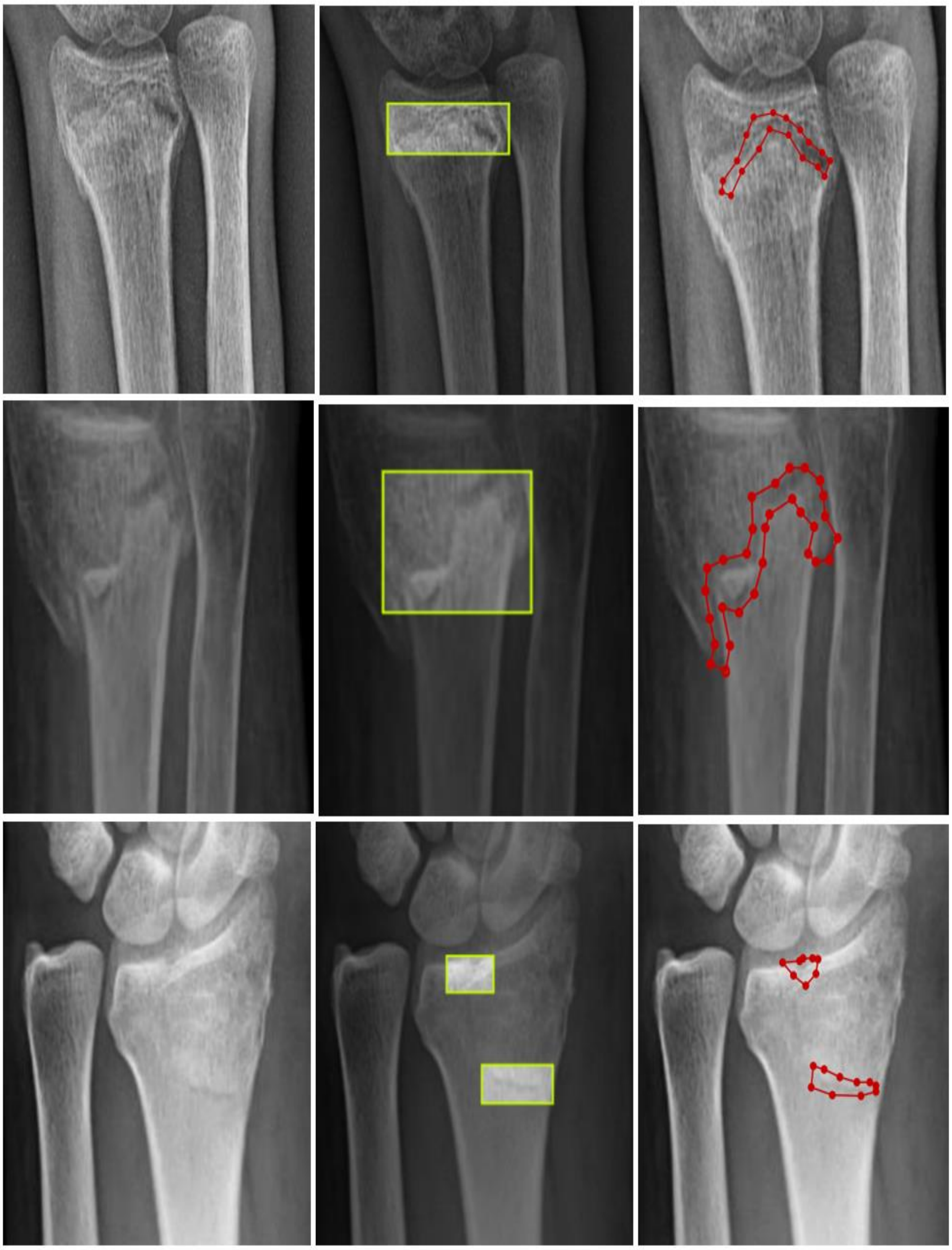


Figure 3.4 An example of original images that have been labelled with the aid of a segmentation mask (highlighted in red) and a bounding box (highlighted using the yellow box).

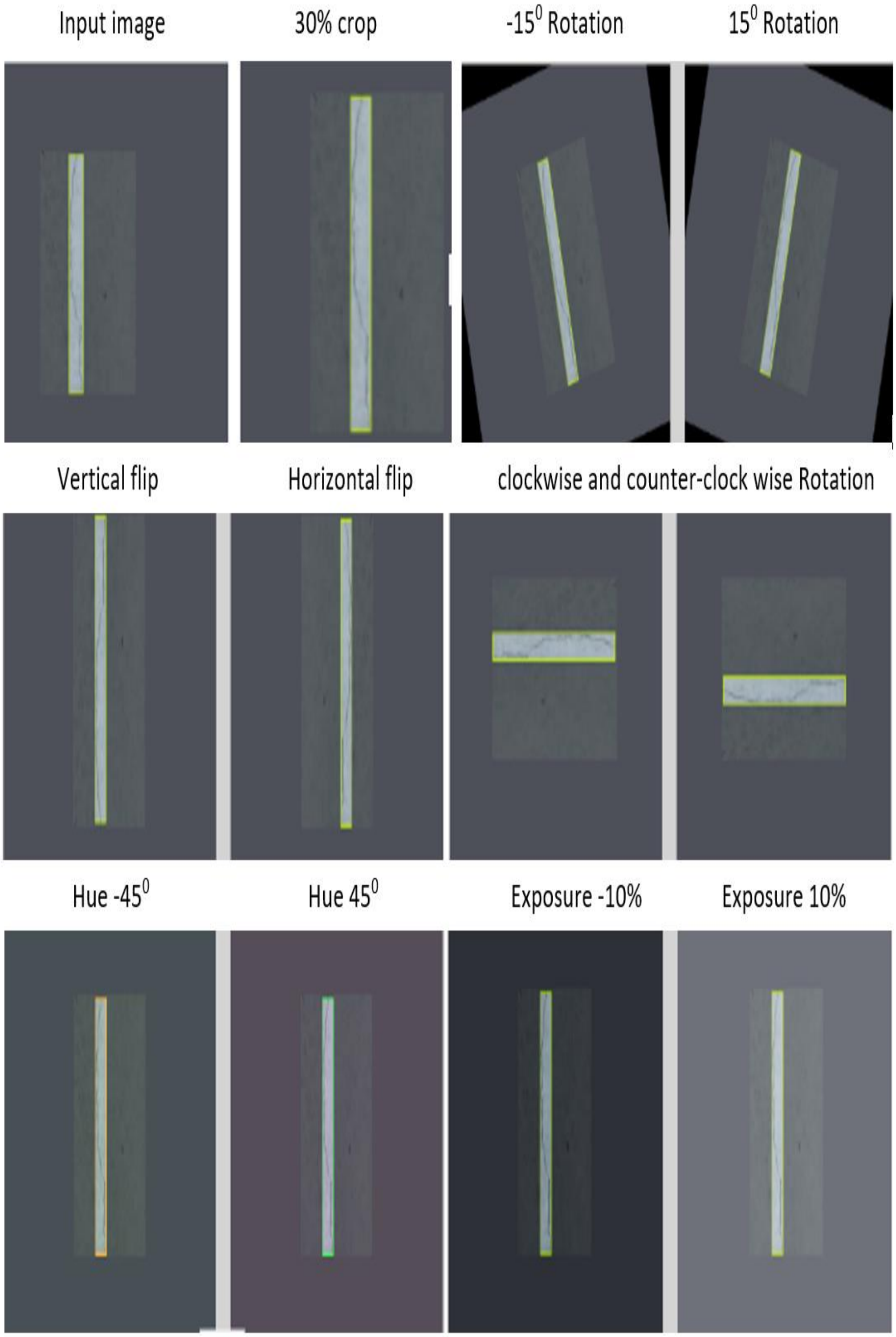


Figure 3.5 An illustration of the effects of augmentation on a SCD input image.

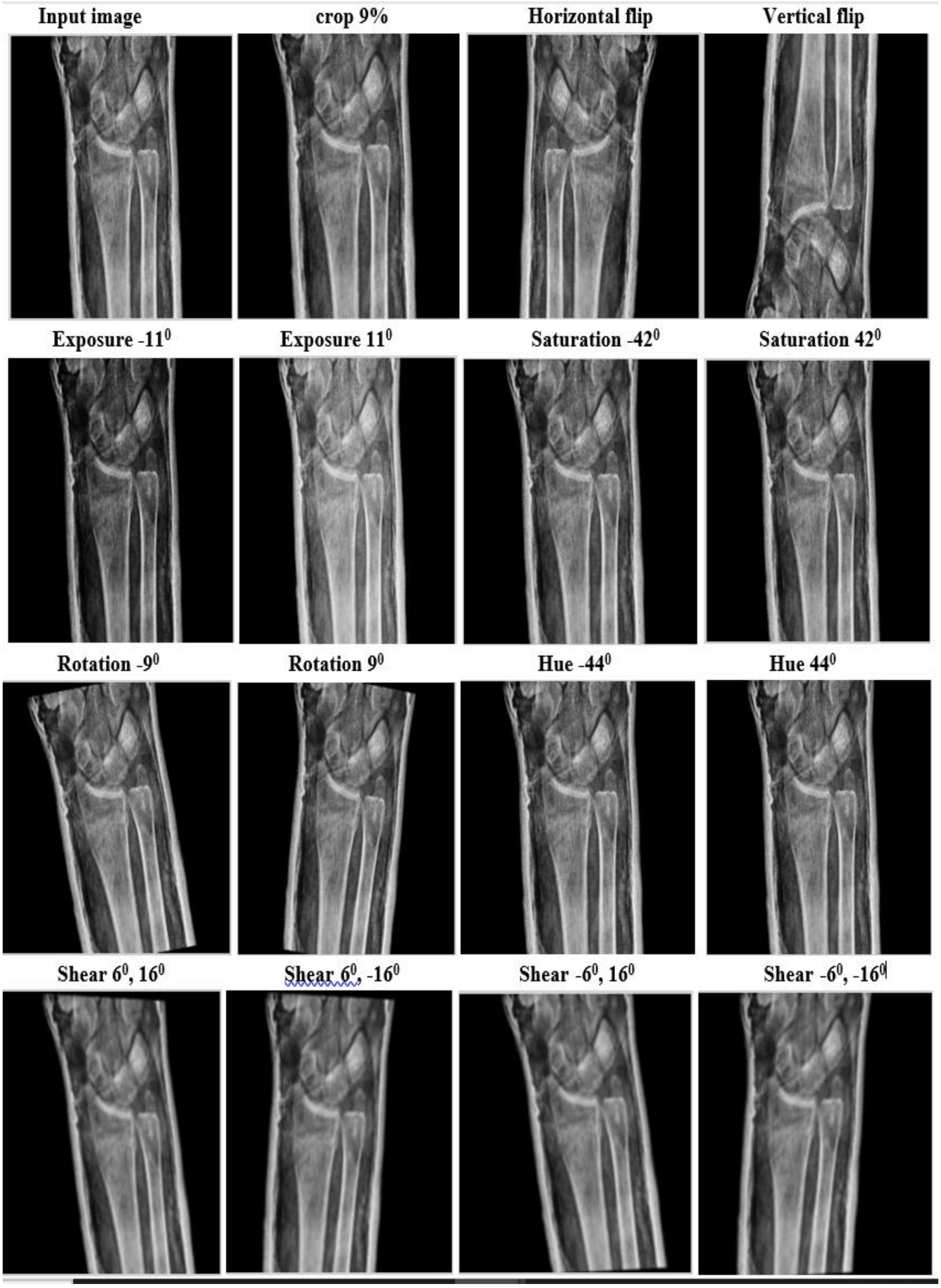


Figure 3.6 An illustration of the effects of augmentation on a WFD input image.

Table 3.5 Augmentation techniques implemented for SCD and WFD datasets.

Technique applied	Use	SCD	WFD
Crop	Aid model generalization with various subject translations and camera positions,	30%	0% Minimum Zoom, 9% Maximum Zoom
Rotation	To make the model resilient to object translations and camera position, it adds variability to position and size.	-15^0 and +15^0, 90^0 clockwise and counterclockwise rotations	Between -9° and +9°
Flip	To make it insensitive to object orientation.	Horizontal, Vertical	Horizontal, Vertical
Hue	To randomly adjust the colors of the input image.	Between (- 45° and + 45°)	Between (-44° and +44°)
Exposure	The brightness of the image is made more variable, which helps the model become less sensitive to changes in lighting and camera settings.	Between (-10% and +10%)	Between (-11% and +11%)
Shear	Helps model be more resistant to camera and subject pitch and yaw, add variability to perspective.	Not applied	±6° horizontal, ±16° Vertical
Saturation	Make arbitrary changes to the images' color vibrancy.	Not applied	Between (-42% and +42%)

3.2 Conclusion

The novel surface crack and wrist fracture dataset used in the research are discussed. Several stages involved in generating the datasets are data collection, preprocessing, data augmenttaion, and data labeling. The wrist radiographs used for conducting experiments were collected between February 2019 and March 2020 from the Doon Hospital in Dehradun, India. The dataset was obtained without disclosing the participant's identity or demographic data following the regulations governing Ethical Conduct in Human Research and Related Activities.

The number of wrist fracture images obtained from the hospitals is 315 consisting of 733 annotations/cracks which is insufficient to generate accurate results using deep learning techniques. Therefore we have incorporated state-of-the-art COCO and self-collected Surface Crack Datasets (SCD) for better model generalization. COCO dataset does not include images from medical domain, more specifically there are no images which has crack like pattern in it. As a consequence, we have developed surface crack dataset. The surface crack dataset consists of 3,000 images collected by capturing the minute cracks, which has similar patterns as the bone fracture cracks.

SCD and WFD are both preprocessed before being fed into the network. The WFD is cropped to exclude the finger bone regions from the hand X-rays. The DICOM image was then converted to the 24-bit lossless JPEG format. Following that, the images go through a labeling and augmentation process. The radiologists annotated the wrist bone images with LabelMe software. Drawing a bounding box may not accurately depict the shape of the fractures because it includes non-essential bone areas when training the model. A mask is created to label each image further by drawing a more intricate shape, such as a polygon, around the crack. A portion of the dataset is made publically available for research to minimize difficulties with data collection and wrist fracture labelling. [118].

CHAPTER -4

A NOVEL METHOD DEVELOPED FOR FRACTURE LOCALIZATION AND SEGMENTATION

4.1 Introduction

Deep learning techniques have made computer-aided design (CAD) systems increasingly capable of assisting radiologists in medical settings. Kim et al. [26] used lateral wrist radiographs to retrain the Inception-v3 network and create a model to ascertain if a new case is fractural. Raghavendra et al. [106] developed a unique CNN classification model to identify thoracolumbar fractures automatically. The automatic classification of osteoporotic vertebral fractures was studied by Tomita et al. [107] utilizing a deep residual network (ResNet) and a long short-term memory (LSTM) network. A straightforward binary classification model was trained on MURA [103], a sizable dataset of musculoskeletal radiographs made up of 40,895 radiographs, by Rajpurkar et al. using a 169-layer DenseNet. A CNN was trained to recognize wrist fractures using lateral and posteroanterior radiographs by Ebsim et al. [108]. England et al. [109] employed deep CNN to find traumatic pediatric elbow joint effusion. Urakawa et al. [110] reported that their fine-tuned model detects intertrochanteric hip fractures by using VGG-16 to analyze whether the proximal femurs cropped from an anterior-view hip radiograph are fractured or non-fractured. Badgeley et al. [111] employed Inception-v3 to forecast hip fractures by confounding patients and using healthcare information. Using AlexNet and GoogLeNet, Adams et al. correctly identify femoral neck fractures in X-rays with a 94.4 percent accuracy rate [112].

Real-world object detection is a challenging problem with two major responsibilities. In order to distinguish between foreground and background objects and apply the relevant object class labels, the detector must first solve the recognition problem. The detector must also assist in solving the localization issue, which necessitates assigning exact bounding boxes to diverse objects.

To illustrate the efficiency of computer-aided identification and classification of calcaneus fracture location in CT images, Pranata et al. combine a ResNet-50 with a SURF approach [113]. Faster-RCNN and Inception-v4 were utilized by Gan et al. to detect distal radius fractures, and the suggested network performed better than orthopedists [114]. In order to locate wrist fractures in radiographs, Lindsey et al. created extensions of the U-Net architecture; a controlled experiment shows that the help of the deep learning model significantly improves emergency medicine physicians' diagnostic accuracy [24]. In order to locate and diagnose thighbone fractures in X-ray images with an Average Precision (AP) of 82.1 percent, Guan et al. [115] created a novel CNN with dilated convolutions.

Recently, generic object detection has gained popularity attributable to single-stage and two-stage detectors. Fast R-CNN [85] and faster-RCNN [86], the first two-stage detectors, were introduced by the region-based convolutional neural network (R-CNN), which progressed the developments. In order to increase the effectiveness of detectors and enable end-to-end training of the detectors, faster-RCNN introduced a Region Proposal Network (RPN). Numerous techniques to improve faster-RCNN from various angles were introduced following this significant milestone. For instance, FPN [116] reduced scale variation using the architecture of a multi-scale feature pyramid. Cascade RCNN [117] enabled faster R-CNN to be extended to a multi-stage detector. The most successful generic object detection method to date is the two-stage detector, widely applied across numerous sectors.

Researchers have been using deep learning algorithms to recognize the fractures on bone X-ray images for fracture classification, detection, and instance segmentation. Regardless of where the cracks are in the images, the classification algorithms assist in determining whether they are present or absent. Finding the exact location of fractures in an image requires localization that goes one step further with object classification. Instance segmentation is a technique that uses a bounding box to mask the objects' exact shape rather than their physical location. These tasks combine classification, localization, and segmentation tasks into a single output that is a polygon mask encircling the defined target. Figure 4.1 displays a sample of image, annotated for object classification, localization and segmentation problem. According to our knowledge, this study is the first one to concentrate on creating a segmentation mask around the class labels to detect wrist fractures. Numerous studies have been based on problems with fracture classification that could only identify a fracture in an image without identifying its location. This task was expanded to include finding the image fractures. Our study also segmented the fractures using the instance segmentation method.

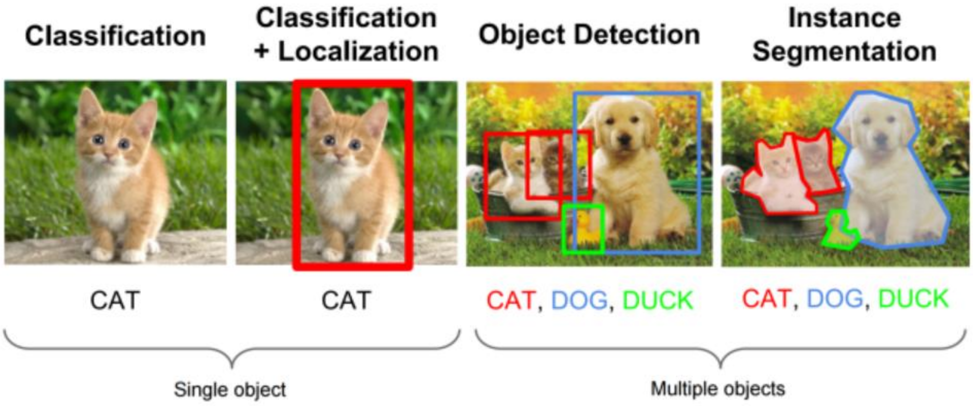


Figure 4.1 A sample of image, annotated for object classification, object localization and instance segmentation problem [132].

The following portions of this chapter are organized as follows. Section 4.2 discusses the suggested methodology for training the model, divided into three phases. Section 4.2.1 addresses the proposed feature extraction methodology for fracture detection, followed by Section 4.2.2, which details fracture localization and segmentation in wrist bones. The conclusion is stated in Section 4.3.

4.2 Proposed Methodology

We have used Instance segmentation, which localize and segments every fracture in the image by assigning a label to each image pixel. This study focuses on constructing the segmentation mask around the class labels to detect wrist fractures. Instance segmentation is the integration of faster-RCNN and semantic segmentation. The intuition behind involving segmentation along with the localization of wrist fractures is better visualization of the shape of the fractures. It has been observed that the fracture shape is extended in vertical and horizontal direction in majority of the X-ray samples collected. The automatic localization of the fractures is improved by creating a segmented mask along with the bounding box to exactly locate the fractures and its shape. This process eliminates the unessential bone area included in the bounding box during model training thus providing better results. The proposed methodology adopted for training the model is divided into three phases I, II, and III.

Phase I

Transfer learning is employed in the proposed work due to the limited availability of the wrist fracture dataset in the public domain [105]. By drawing on prior knowledge of similar tasks, transfer learning with convolutional neural networks aims to enhance performance on a new task. It has significantly improved medical image analysis by overcoming data scarcity and saving time and hardware resources. The number of wrist fracture images acquired from the hospitals is 315 consisting of 733 annotations/cracks, which is insufficient to generate accurate results using deep learning techniques. We have applied the transfer learning concept since the pre-trained weights can be transferred from one domain to another [123]. The first phase uses the state-of-the-art object recognition and segmentation dataset MS COCO (Microsoft Common Objects in Context) [119]. There are 3,28,000 photos in the COCO dataset of complex daily objects, including bottles, cars, chairs, cows, dogs etc. The photos contain 2.5 million instances with labels and 91 different object types. The COCO dataset is used because it consists of 2.5 million images, which helps in better model convergence and generates accurate results when applying deep learning techniques.

In phase I, the pre-trained weights for the COCO dataset trained on the mask-RCNN architecture built on Feature Pyramid Network (FPN) and ResNet-50 architecture are acquired from the publicly available resources [124]. These weights are utilized for fine-tuning the surface crack dataset in Phase II.

Phase II

The COCO dataset does not include images from the medical domain; more specifically, there are no images with a crack-like pattern. Consequently, we have developed a new-found surface crack dataset (SCD) dataset. The dataset consists of 3,000 images of pavements, walls, etc., having similar patterns as bone fractures illustrated in Figure 3.1. Transfer learning is utilized in the proposed methodology to transfer the knowledge from the non-medical (COCO) dataset to the medical domain dataset via surface crack images. This technique is efficacious for the limited dataset, where we can effectively apply pre-acquired knowledge for executing a task.

Layered architectures used in deep learning systems and models allow for the learning of various features at different layers. The final output is obtained by connecting these layers to a final layer. We can use a pre-trained network without its final layer as a fixed feature extractor for different tasks due to this layered architecture. The inductive learning method is exemplified by deep learning models. To infer a mapping from a set of training instances is the goal of inductive learning methods. The fundamental concept is to simply use the weighted layers of the pre-trained model to extract features without adjusting the weights of the layers while training with the new dataset [15].

Since COCO and wrist fracture dataset are vastly different, using COCO characteristics for fracture detection is less effective. As a result, we included surface crack images in place of directly transferring the pre-trained weights from the COCO dataset to fine-tune the model on the wrist fracture dataset. Figure 4.2 illustrates knowledge transfer from the COCO dataset to the WFD through SCD. A schematic diagram representing the overall steps involved in fracture detection is shown in Figure 4.3. In phase II, an inductive transfer learning mechanism is employed because the source (COCO dataset) and destination domains (Surface crack dataset) are different. The learning task domains of the COCO dataset and SCD are also different. Therefore, the surface crack images are fine-tuned to accurately detect and segments cracks in the input images, based on equation 4.1.

$$if \qquad Ds \neq Dt \; or \; Ts \neq Tt \qquad 4.1$$

It improves the learning of $ft(.)$ in Dt by applying the knowledge in $Ds \; and \; Ts,$ where $Ts \neq Tt$

The source domain is Ds, and the source domain's learning task is Ts. Tt is the target domain's learning task, and Dt is the target domain. The predictive function is $ft(.)$

Phase III

Phase III utilizes a transductive transfer learning mechanism where the source (Surface crack dataset) and destination domains (wrist fracture dataset) are different. The learning task of both the domains is same, which is to identify cracks in the images, therefore equation 2 is applicable for identifying fractures in wrist images.

$$if \qquad Ds \neq Dt \; or \; Ts = Tt \qquad 4.2$$

It improves the learning of $ft(.)$ in Dt by applying the knowledge in $Ds \; and \; Ts,$ where $Ts = Tt$

The source domain is Ds, and the source domain's learning task is Ts. Tt is the target domain's learning task, and Dt is the target domain. The predictive function is $ft(.)$

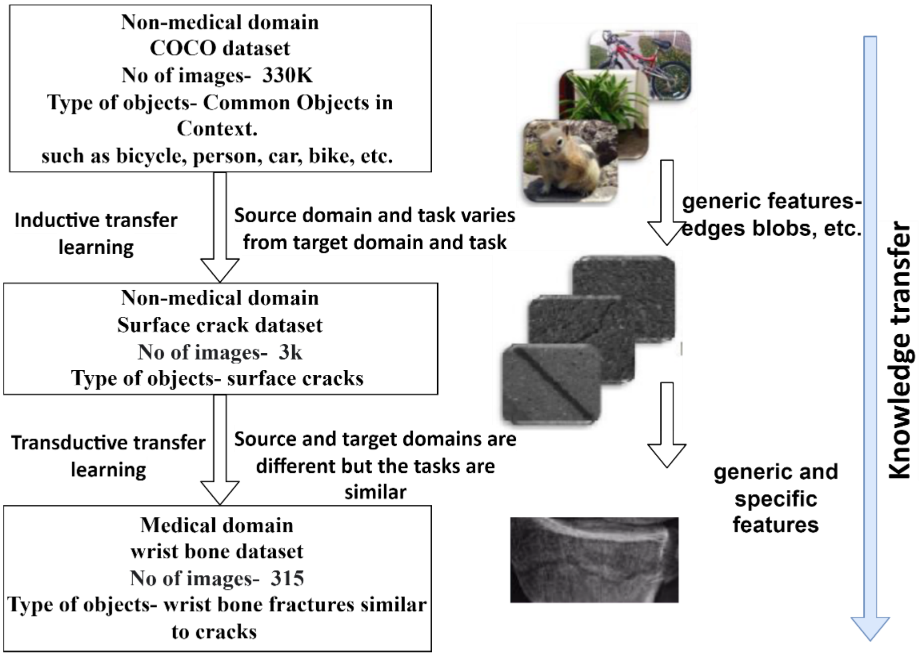

Non-medical domain
COCO dataset
No of images- 330K
Type of objects- Common Objects in Context.
such as bicycle, person, car, bike, etc.
Inductive transfer learning
Source domain and task varies from target domain and task
Non-medical domain
Surface crack dataset
No of images- 3k
Type of objects- surface cracks
Transductive transfer learning
Source and target domains are different but the tasks are similar
Medical domain
wrist bone dataset
No of images- 315
Type of objects- wrist bone fractures similar to cracks
generic features- edges blobs, etc.
generic and specific features
Knowledge transfer

Figure 4.2 Knowledge is transferred from datasets in non-medical domains (COCO) to WFD using the transfer learning methodology.

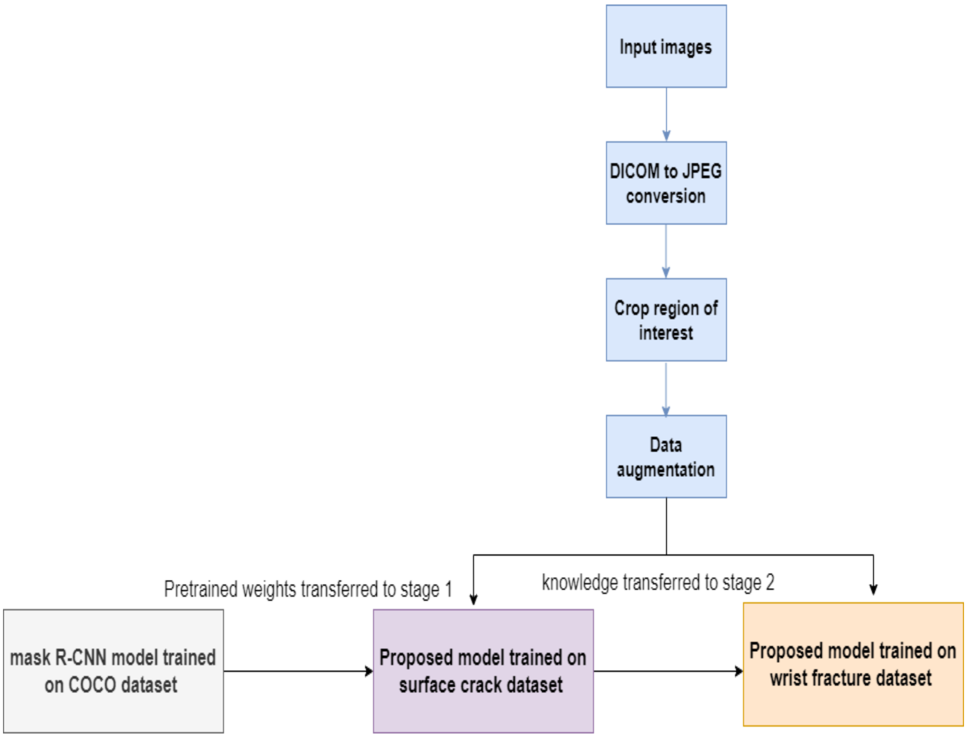

Input images
DICOM to JPEG conversion
Crop region of interest
Data augmentation
Pretrained weights transferred to stage 1
knowledge transferred to stage 2
mask R-CNN model trained on COCO dataset
Proposed model trained on surface crack dataset
Proposed model trained on wrist fracture dataset

Figure 4.3 A schematic diagram representing overall steps involved in fracture detection

4.2.1 Feature extraction for fracture detection in wrist bones

The overall architecture of wrist fracture detection is divided into three sub-architectures: the Backbone network, the Region Proposal Network (RPN), and the Region of Interest Align (RoIAlign), depicted in Figure 4.4. The task of extracting features from the input image takes place in the Backbone architecture, which comprises a top-down and bottom-up pathway, illustrated in Figure 4.5.

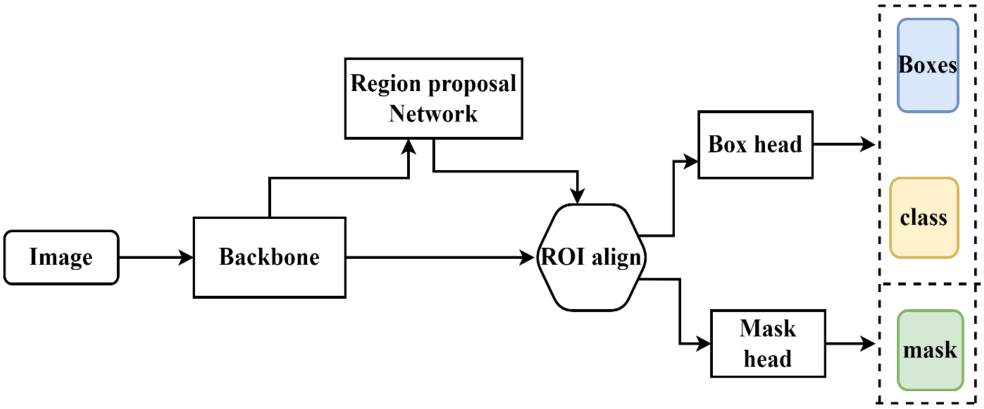


Figure 4.4 Overall architecture for wrist fracture detection is presented.

The in-depth architecture of the top-down and bottom-up pathways is illustrated in Figure 4.6. It consists of an input stem (1), residual layers (2), lateral convolution layers (3), output convolution layers (4), and a modified last-level concatenated layer (5).

A convolutional network intended for feature extraction is deployed in the top-down pathway. The input stem and residual layers of the top-down pathway are inspired by ResNet-50 architecture [63]. To identify large and small objects, we have created a pyramid using the same image at various scales. The bottom-up pathway comprises lateral convolution and output convolutional layers. The bottom-up pathway is inspired by the Feature Pyramid Network (FPN) [116], which combines low-resolution, semantically powerful features with high-resolution, semantically weak features. The spatial resolution of the image reduces as we advance deeper into the CNN architecture, whereas the semantic value for each layer increases as more high-level structures is detected. The output of this layer are the features generated at different scales: (1/4) scale-> P2, (1/8) scale-> P3, (1/16) scale-> P4, (1/32) scale-> P5, (1/64) scale-> P6.

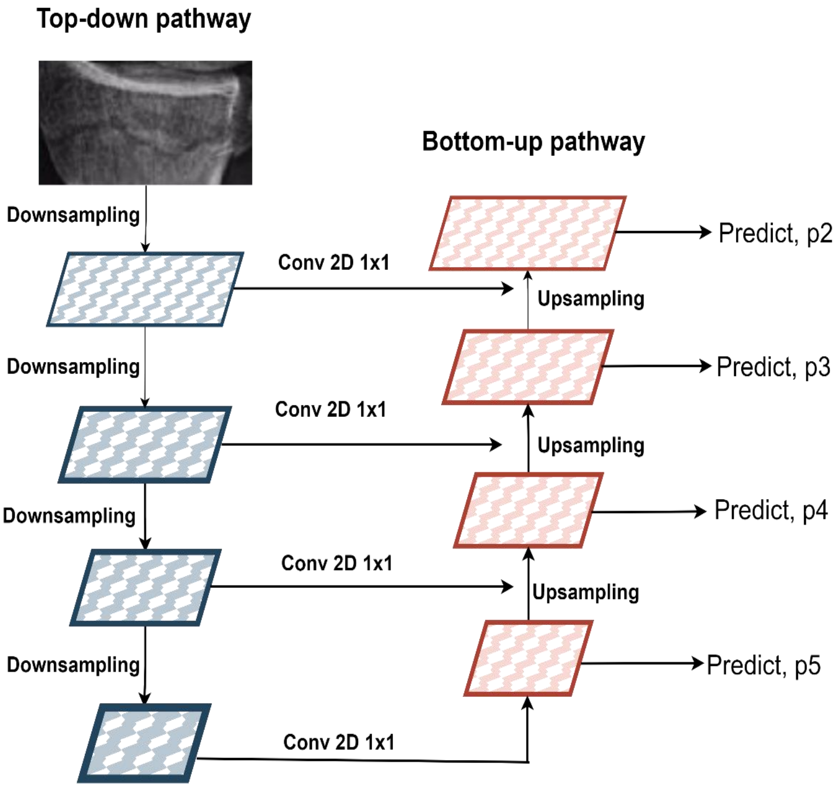


Figure 4.5 A layout of the top-down and bottom-up pathway is illustrated.

The input stem uses a Conv layer with kernel size 7 and stride equal to 2 to downsample the input image. To achieve high computational speed, the number of parameters must be reduced. As a result, following the convolution operation, the input stem reduces the size of the input image by using the maxpool layer with kernel size of 3 and stride value of 2. The input stem returns a feature map tensor with the dimensions (B, 64, H / 4, W / 4), where B represents batch size and H and W represent Height and Width, respectively.

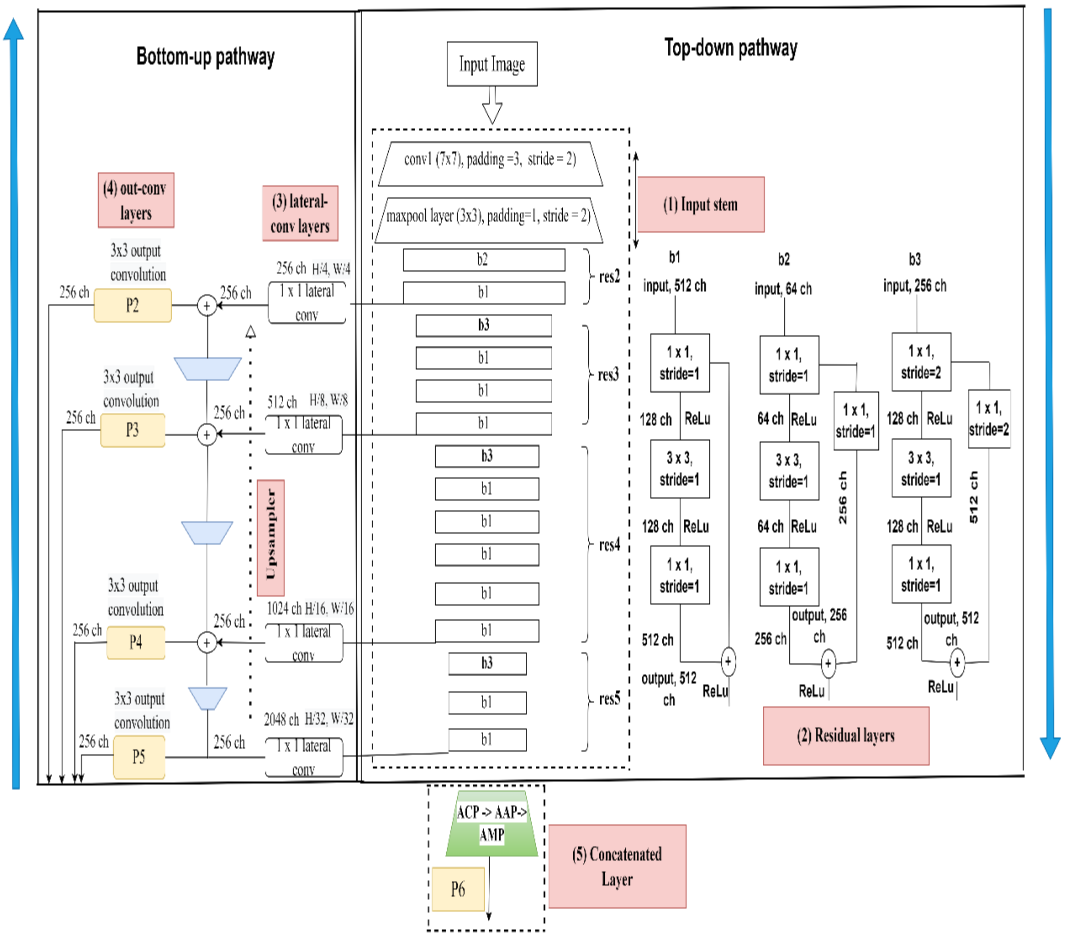


Figure 4.6 The indepth architecture of the top down and bottom up pathway is illustrated. It consists of an input stem (1), residual layers (2), lateral convolution layers (3), output convolution layers (4), and a modified last-level concatenated layer (5).

The residual layers consist of b1, b2, and b3 blocks, where each block consists of three convolution layers, the details of which are mentioned in Table 4.1 Architecture level details used in the proposed methodology. The ResNet blocks (2-5) consist of a combination of residual layer blocks (b1, b2, b3). The first block (b3) of stages (3-5) in the ResNet architecture downsamples the feature map. A shortcut convolution layer is added in b2 and b3 blocks to match the input and output channels at the first block of the ResNet stage (2-5). The identity shortcuts add the input and output features in b2 and b3. At block 3, a shortcut connection is added with stride=2 to match the input and output channels.

The feedforward feature maps are calculated from the ResNet-50 architecture through a top-down layout. The output feature maps generated from ResNet blocks have the format-

```
res5->  [1, 2048, H/32, W/32]        # stride = 32
res4->  [1, 1024, H/16, W/16]        # stride = 16
res3->  [1, 512, H/8, W/8]           # stride = 8
res2 -> [1, 256, H/4, W/4]           # stride = 4
```

The res(2-5) block has [256, 512, 1024, 2048] output channels. ResNet blocks (2–5) extract features, which are then sent to 1x1 lateral convolutional layers to produce feature maps with 256 channels. To match the dimensions of the feature maps created by the previous layer, the ResNet block feature map is upsampled by a factor of 2. The output of ResNet stage 4 includes the 256-channel feature map from the lateral Conv layer of ResNet stage 5, which was added by the nearest neighbor upsampler. The resulting feature map is then subjected to the 3 x 3 output convolution, yielding P4. The same operation was carried out in an upward direction three times to create the P3 and P2 feature maps.

The Adaptive Concat Pooling (ACP) layer is then employed to concatenate the Adaptive Average Pooling (AAP) and Adaptive Max Pooling (AMP) layers to produce the P6 output. The activation from ResNet-50's last convolution layer is max-pooled to the subsequent fully connected layer in the original ResNet-50 architecture. The proposed architecture preserves the maximum and average activations of the last convolution layer to enable the neural network to select the optimal approach without the requirement for individual experimentation. It has been found that the last layer's H x W feature map's maximum value performs better than the average and vice versa. The AAP and AMP layers are combined with the ACP layer in the revised model. Three different pooling layers are used as a transitional phase to connect the convolution layer to the fully connected layers. Our research preserves the maximum and average activations from the preceding convolution, offering the model knowledge of both the approaches and enhancing performance.

4.2.2 Fracture localization and segmentation

The basic idea of this sub-architecture is to identify regions or areas with a possibility of fracture presence in the image. Once we identify the areas, we label them as foreground, background, and ignored class which is given to a CNN for classification and regression. The ground truth box annotations indicating the location and size are mapped with the feature maps generated in the backbone architecture. We have utilized a multi-scale network proposed in the backbone architecture to detect small and large cracks in the image. The smaller objects are detected by p2 and p3, while p4, p5, and p6 detect larger objects.

The sub-architecture in this stage comprises of two stages- RPN and RoIAlign layer [101]. The detailed architecture of RPN is depicted in Figure 4.7.

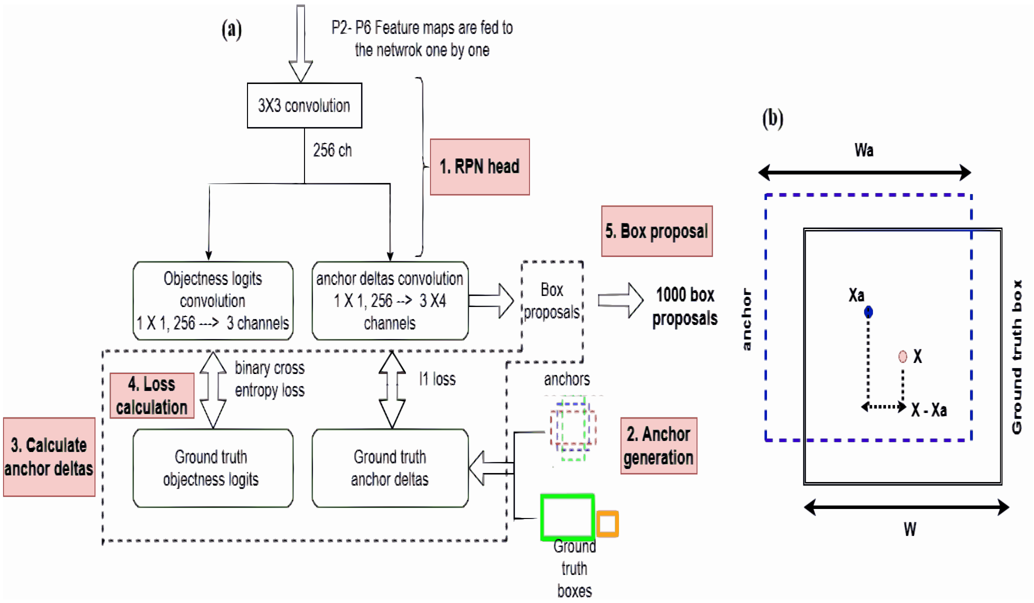

(a)
P2- P6 Feature maps are fed to the netwrok one by one
3X3 convolution
256 ch
1. RPN head
Objectness logits convolution 1 X 1, 256 ---> 3 channels
anchor deltas convolution 1 X 1, 256 --> 3 X4 channels
Box proposals
5. Box proposal
1000 box proposals
4. Loss calculation
binary cross entropy loss
l1 loss
anchors
3. Calculate anchor deltas
Ground truth objectness logits
Ground truth anchor deltas
2. Anchor generation
Ground truth boxes
(b)
Wa
anchor
Xa
X
X - Xa
Ground truth box
W

Figure 4.7 The detailed architecture of RPN is depicted.

RPN has two major components; the RPN head with neural network functionalities and the remaining layers with non-neural network functionalities. RPN applies a sliding window on the predicted feature maps to identify the objectness score (has an object or not) and the bounding box of the object. RPN head comprises three convolutional layers conv1, conv2, and conv3. The backbone architecture's feature maps (p2 – p6) are provided as input to the RPN head, generating the objectness logits and anchor deltas. The objectness logits represent the probability of object existence, while the anchor delta represents the relative box shape to anchors.

pred_obj = [B, 3 ch, H_i, W_i]

pred_anchor = [B, 3×4 ch, H_i, W_i]

where B is batch size, H_i, and W_i correspond to the feature map sizes of P2 to P6.

Anchor generation is the next step toward object detection, which helps connect the pred_obj and pred_anchor to the ground truth boxes. A series of bounding boxes with a predetermined height and breadth are called anchors. There are nine anchors in the faster-RCNN's default configuration at a point of an image. However, we have utilized three anchors of various sizes to identify fracture cracks. Anchor sizes of (32, 64, 128, 256, and 512) are used for (p2-p6) feature maps generated from the backbone architecture. The aspect ratio of 2:1, 1:1, and 1:2 is set for defining the shape of the anchors. For example, the p2 feature map has a 32x32 dimension with 1:2, 1:1, and 2:1 aspect ratios resulting in three anchors displayed in Figure 4.8.

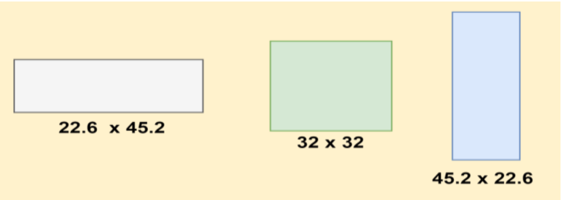


Figure 4.8 Cell anchors for the P2 feature map with the aspect ratios of 1:2, 1:1, and 2:1.

Similarly, l2 anchors are generated for p3, p4, p5, and p6 feature maps. The anchors generated for p2-p6 feature maps are placed on the corresponding predicted feature maps (p2-p6). The size and stride of the anticipated feature map, p2, are (150x200) and 4, respectively. Next, each grid cell is attached with the three anchors, creating 150x200x4 anchors. The anchor generation process is repeated for the remaining feature maps, yielding a total of 1,20,015 anchors. The Intersection over Union (IoU) is computed using the generated anchors and the ground truth boxes. The objective is to find anchors similar to the ground truth boxes out of 1,20,015 anchors using the concept of IoU. We have defined a threshold of 0.7 to label the anchor as background, foreground, or ignored. Foreground anchors possess higher than 70% overlaps with ground truth boxes. The generated anchors and the ground truth box are allocated as background if the IoU is less than a specified threshold (30%). It is considered ignored if the percentage is between 30% and 70%.

Ground truth boxes and foreground anchor labels have similar shapes, and the network has been taught to properly detect the precise location and shape of the ground truth boxes. The ground truth box annotations are used in this layer to calculate the loss. The annotation data consists of a class label that identifies the presence or absence of a fracture and a box parameter indicating the location and size of the bounding box containing a fracture. The ground truth box labels are required for detecting the fractures in this layer. The four regression parameters mentioned in the equations 3-6 are defined, which are required to identify the exact location of the ground truth boxes.

$$\Delta x = \frac{(X - Xa)}{Wa} \qquad 4.3$$

$$\Delta w = log\left({}^{W}/_{Wa}\right) \qquad 4.4$$

4.4

$$\Delta y = \frac{(Y - Ya)}{Ha} \qquad 4.5$$

$$\varDelta h = log\left({}^{H}/_{Ha}\right) \qquad 4.6$$

The next step is to resample the boxes for calculating the loss. The resampling is essential as the number of anchors generated per image is 1,20,015, where most anchors are background. In our experiments, less than 100 anchors are the foreground, less than 1000 anchors are ignored, and the remaining are background. Next, the loss is calculated on the predicted objectness maps and the ground truth labels. Localization loss or l1 loss is computed by ignoring all background labels. It is applied to the grid points of the predicted objectness maps where the ground truth score is 1. Objectness score or binary cross entropy loss is applied to the grid points of the predicted objectness maps where the ground truth score is 1 and 0. Finally, 1000 region proposal boxes are selected by applying the predicted anchor deltas to the corresponding anchors and sorting them based on the objectness score individually for (p2-p6) feature maps. Next, Non-Max Suppression (NMS) is applied to selected top-scored 1000 boxes.

4.2.2.1 Region of Interest Align (RoIAlign)

The architecture comprises two parts: RoI head and RoI pool. The detailed architecture of RoI head is depicted in Figure 4.9.

RoI head accept as inputs ground truth boxes, 1000 region proposal boxes, and feature maps from the backbone architecture (p2-p5). To accelerate the training process in RoI head, the ground truth boxes are included in the proposal boxes. Afterward, these are categorized as foreground or background based on the IoU calculation threshold. The foreground and background samples are resampled to encounter the imbalance dataset problem. In RoI pooling, the proposal boxes-specified feature maps are cropped into rectangular regions of interest. The RoI is cropped from the feature maps by allocating proposal boxes to the relevant feature maps using the equation 7.

$$Feature\ level\ assigned = floor\left(\frac{4 + log2\ \sqrt[2]{proposal\_box\_area}}{224}\right) \qquad 4.7$$

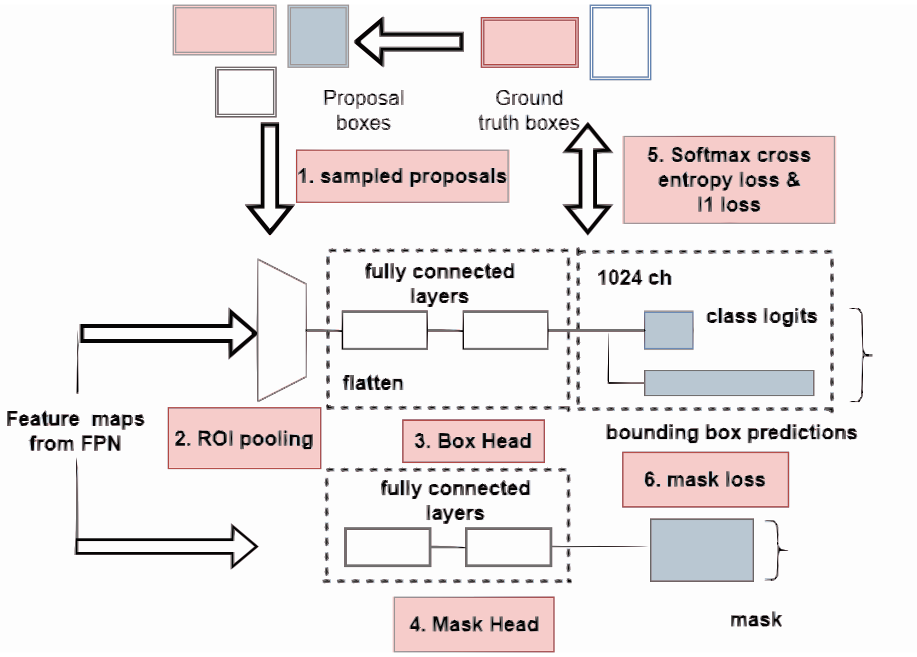


Figure 4.9 Detailed architecture of the RPN is depicted.

The region of interest (RoIs) is accurately cropped by the proposal boxes consisting of floating point coordinates. The detectron2 [125] package modifies the RoIAlign technique from the mask-RCNN architecture to crop the region of interest precisely. The RoIAlignv2 is a modified version of the RoIAlign that computes the neighboring pixel value by deducting the half-pixel offset from RoI coordinates. This method has overcome the disadvantages of choosing a slightly off-aligned pixel value while using bilinear interpolation. The resulting tensor after cropping the RoIs from corresponding feature maps (p2- p5) has the size of [N × B, 256, 7, 7], where N x B is number of RoIs across the batch, 256 is the number of channels, seven corresponds to the height and width of the RoI.

4.2.2.2 Box head and mask head

The box head consisting of two fully connected layers receives an input of a flattened tensor of [B, 256,7, 7 = B, 256x7x7=12,544] channels. The output from this layer is the classification score and bounding box predictions. Next, the classification and localization loss is calculated during training.

A parallel layer is added with the existing object detection framework to generate the mask around the cracks. The output obtained from the box head consists of a class label and the bounding box parameters. The RPN stage is adopted here to perform the pixel-to-pixel alignment, followed by extracting features using RoI pool and RoIAlignv2 from each bounding box. A multi-task loss is calculated for each region of interest obtained, which is calculated using equation 4.8.

$$L = L_{obj} + L_{loc} + L_{mask} \tag{4.8}$$

The sub-architecture in this layer consists of a convolutional layer to extract features, generating pixel-to-pixel mapping. Next, another convolutional layer is added, followed by RoIPool and RoIAlign layers to obtain the bounding box for the classification and regression task.

Table 4.1 Architecture level details used in the proposed methodology

Backbone architecture- Feature Extraction				
Layers	**Input**			**Output**
(1) input stem	Conv1 (kernel=7x7, stride= 2),	batchnorm layer, ReLU, Maxpool layer (kernel=3x3, stride= 2)		tensor with dimensions- (B, 64, H / 4, W / 4).
(2) residual layers	res2 stage, 1/4 scale block b2 (stride=1, with shortcut conv) block b1 (stride=1, w/o shortcut conv) × 2	b1- stride=1, No shortcut conv	[Conv1 (kernel=1x1), Conv2 (kernel=3x3), Conv3 (kernel=1x1)], stride=1	res2 = [1, 256, H/4, W/4], stride = 4
	res3 stage, 1/8 scale			

	block b3 (stride=2, with shortcut conv) block b1 (stride=1, w/o shortcut conv) × 3	b2- stride=1, identity shortcuts	[Conv1 (kernel=1x1), Conv2 (kernel=3x3), Conv3 (kernel=1x1)], stride=1	res3 = [1, 512, H/8, W/8, stride = 8
	res4 stage, 1/16 scale block b3 (stride=2, with shortcut conv) block b1 (stride=1, w/o shortcut conv) × 5	b3- stride=2, identity shortcuts	Conv1(kernel=1x 1), stride=2 [Conv2 (kernel=3x3), Conv3 (kernel=1x1)], stride=1	res4 = [1, 1024, H/16, W/16], stride = 16
	res5 stage, 1/32 scale block b3 (stride=2, with shortcut conv) block b1) (stride=1, w/o shortcut conv) × 2			res5 = [1, 2048, H/32, W/32], stride = 32

(3-5) lateral, out, & concatenated layers	res2 = [1, 256, H/4, W/4], stride = 4
	res3 = [1, 512, H/8, W/8, stride = 8
	res4 = [1, 1024, H/16, W/16], stride = 16
	res5 = [1, 2048, H/32, W/32], stride = 32
		Lateral conv layers: (res2-res5)→(1x1 conv layer) x4,
		Upsampler: (F.interpolate with nearest neighbor),
			Concatenated Layer:
			ACP = AAP + AMP
				p2 = [1, 256, 150, 200], stride = 4
				p3 = [1, 256, 75, 100], stride = 8
				p4 = [1, 256, 38, 50], stride = 16
				p5 = [1, 256, 19, 25], stride = 32
				p6 = [1, 256, 10, 13], stride = 64

Region Proposal Network (RPN)				

(1) RPN Head	p2 = [1, 256, 150, 200],
	p3 = [1, 256, 75, 100],
	p4 = [1, 256, 38, 50],
	p5 = [1, 256, 19, 25],
	p6 = [1, 256, 10, 13]
		Conv1 (kernel=3×3, 256 -> 256 ch), pred_obj_Conv2 (kernel=1×1, 256 -> 3 ch), pred_anchor_Conv3(kernel=1×1, 256 -> 3×4 ch).		for (p2-p6) feature maps-
				pred_obj = [B, 3 ch, Hi, Wi]
				pred_anchor = [B, 3×4 ch, Hi, Wi]

(2) anchor generation	Anchor sizes for (p2-p6) feature maps-
	p2 =32
	p3= 64
	p4= 128
	p5= 256
	p6= 512
	Aspect ratio= [0.5, 1.0, 2.0]
		Anchors generated for (p2-p6) feature maps-
		p2=150x200x3=90,000
		p3=75x100x3=22,500
		p4=19x25x3=5,700
		p5=19x25x3=1,425
		p6=10x13x3=390
				total anchors generated= p2+p3+p4+p5+ p6= 1,20,015

(3) calculate anchor deltas	Total anchors generated= p2+p3+p4+p5+p6= 1,20,015	Label the generated anchor as-
		if IoU>=70%, foreground
		elif IoU<=30%, Background
		elif IoU>30% and IoU<70%, Ignored
			Calculate anchor deltas for foreground labels –
			(Δx, Δy, Δw, Δh)
			Calculate -
			l_1 loss
			objectness loss
				proposal_boxes: 1,000
				objectness_logits: 1,000

RoI calculation				
box head and pool	1000 proposal boxes + (p2-p5) feature maps + ground truth boxes	Proposal boxes as labeled based on the IOU value, Foreground and background boxes are resampled	RoIAlignv2 is used to accurately crop the Region of interest	tensor size of the cropped RoIs from (p2-p5) feature maps- [N × B, 256, 7, 7]
box head and mask head				
(1) box head	cropped RoIs from (p2-p5) feature maps- [N × B, 256, 7, 7]	FC_Layer1(in_features=12,544, out_features=1024, bias=True) FC_Layer2(in_features=1024, out_features=1024, bias=True		class label , bounding-box offset
(2) mask head	cropped RoIs from (p2-p5) feature maps- [N × B, 256, 7, 7]	Warped feature vectors for each RoI is passed to the conv layers	RoIAlignv2- accurately crop the RoI having floating-point values.	class label , bounding-box offset, object mask

4.3 Conclusion

This study focuses on constructing the segmentation mask around the class labels to detect wrist fractures using the integration of faster-RCNN and semantic segmentation. Transfer learning is utilized in the proposed methodology to transfer the knowledge from the non-medical (COCO) dataset to the wrist-fracture (medical-domain) dataset via surface crack dataset. The first phase of the proposed methodology uses the COCO dataset for genertaing the pre-trained weights, which are then used in phase II where surface crack dataset is used to generate high level features like cracks in the images. Finally, the model is fine-tuned on wrist fracture dataset by utlizing the knowledge acquired from phase II.

The feature maps are retrieved from the input image using a modified feature pyramid network involving ResNet-50 architecture as convolution neural network. The details of extracting features from the input image is mentioned in the top-down and botton-up pathway of the backbone sub-architecture of the proposed model. The top-down pathway meant for feature extraction comprises of input stem and residual layers, whereas bottom-up pathway is meant for constructing semnatically strong features by merging the feature maps extracted from the top-down pathway. The feature pyramid network integrates low-resolution, semantically robust features with high-resolution, semantically weak features. Next, the task of fracture localization and segmentation is adopted from the mask-RCNN [86] model, consisting of three stages-RPN (Region Proposal Network), RoIAlign layer, and box-mask head.

CHAPTER -5

EXPERIMENTAL RESULTS

5.1 Introduction

The neurons are the processing nodes in a deep neural network (DNN) that operate on the data as it moves through the network. Each node in the DNN has a weight value associated with training that indicates to our model how much of an impact it will have on the prediction outcome. These weights illustrate a parameter in the model [126]. *Hyperparameters* are the parameters that control the training. Configuring a Deep Neural Network (DNN) includes, for example, deciding how many hidden layers of nodes to use between the input and output layers and how many nodes each layer requires. These variables have no direct relationship to the training data but are configuration variables. Typically, hyperparameters remain constant throughout a task, while parameters change during a training job [126].

The process of selecting the best set of hyperparameter values to employ when training a model using the tuned algorithm on any given data set is known as parameter tuning. The model's performance is optimized using a set of hyperparameters, which minimizes a specified loss function, and produces better results with fewer errors. It should be noted that the learning algorithm optimizes the loss based on the input data and seeks the best solution within the constraints. However, hyperparameters precisely define this configuration. Our predicted model parameters will not give optimal results if our hyperparameters are not properly tweaked to minimize the loss function. The accuracy or confusion matrix will be worse in reality [127].

5.2 Hyperparameters Tuning

To achieve the best possible outcomes, hyperparameters must be tuned for surface crack identification and fracture detection. The proposed model is tuned by selecting the appropriate parameters and hyperparameters significant for analysis and experimentation. The parameters, such as the number of filters, filter size, activation function, pooling size, etc., chosen while training the model at sub-architecture levels are discussed in the previous chapter. The learning strategy adopted and weight initialization techniques involved with the motive to minimize the cost function is discussed in this section.

The proposed methodology involving Phase I, II, and III adopted for training the model is explained in depth in section 4.2. The pre-trained weights of the COCO dataset obtained from Phase I of the proposed methodology are transferred to Phase II for surface crack detection. The weight file from the pre-trained mask-RCNN model is incorporated rather than applying random weight initialization techniques. The Backbone sub-architecture meant for extracting features from the input image is inspired by the mask-RCNN model, where three alternatives of the CNN architectures were available: ResNet50, ResNet101, and ResNeXt101 [128].

The trade-off between accuracy and training time motivated us to select ResNet50 as the feature extractor because ResNet50 trains faster than the later models because it comes with several open-source pre-trained weights for large datasets like COCO. ResNet50 architecture has significantly shortened the training period for various instance segmentation approaches. ResNet101 and ResNext101 architectures will take longer to train because they have more layers. However, they will likely perform better if no pre-trained weights are utilized and fundamental variables like the number of epochs and learning rate are correctly adjusted. The best method for real-world item recognition is to start with pre-trained weights, such as COCO with ResNet50, and assess the model's performance. The models pre-trained on the COCO dataset operate faster and more effectively. The ResNet101 and ResNext101 architectures can be investigated if accuracy is crucial and high computing power is available.

We used the notion of freezing and unfreezing certain layers of the proposed architecture. While the model is being trained, a few layers of the backbone architecture for feature extraction are frozen. Specifically, the early layers of the ResNet-50 architecture, which is utilized as a feature extractor, are frozen, so the weights for the model are not modified during backpropagation. The first layer of CNN is intended to detect simple gradients of the line, the second layer to find simple shapes, and the third layer to come across combinations of lines and shapes. These early layers are meant to obtain general characteristics. On the other hand, the latter layers concentrate more on the particular task at hand, such as identifying the image's crack patterns.

It is unlikely to generate better features at the initial layers while updating the gradients at the same learning rate because the features predicted by the initial layers of a CNN architecture will be the same irrespective of the dataset used. The source and target domains in Phase I and Phase II are different, where Phase I utilizes COCO dataset as opposed to Phase II, which uses a surface crack dataset. Therefore, we have not trained the initial layers of the backbone architecture in Phase II. The results are saved, and the model is loaded in Phase III to detect wrist fractures. The source and domain datasets are different, but the task is similar in Phase II and Phase III, where both stages aim to identify the cracks in the input image. Therefore, the model is trained using a surface crack weight file without freezing any layer in Phase III.

The network updates the parameters at a differential learning rate using a learning rate finder curve [129]. This method employs different learning rates for deep neural architecture segments [130]. The learning rate gradually rises from an exponentially low value (10^{-7}) to a high value (1) while training the data in small batches. The training rate fluctuates from a lower learning rate boundary (min $\_lr$) to a higher boundary (max $\_lr$) during the cool-down Phase before returning to the initial low boundary rate. During the anhillation phase, the learning rate value is further decreased to $1/10$ of (min $\_lr$), as shown in Figure 5.1. Following each mini-batch, the learning rate is updated using the following formula:

$$lr_i = init\_lr * \left(\frac{max\_lr}{init\_lr}\right)^{i/n} \qquad 5.1$$

$$max_{lr} = init_{lr} * q \qquad 5.2$$

where n is the number of iterations and q is the factor by which the learning rate is increased after every mini-batch. The summary of the training involving hyperparameter tuning is provided in Table 5.1.

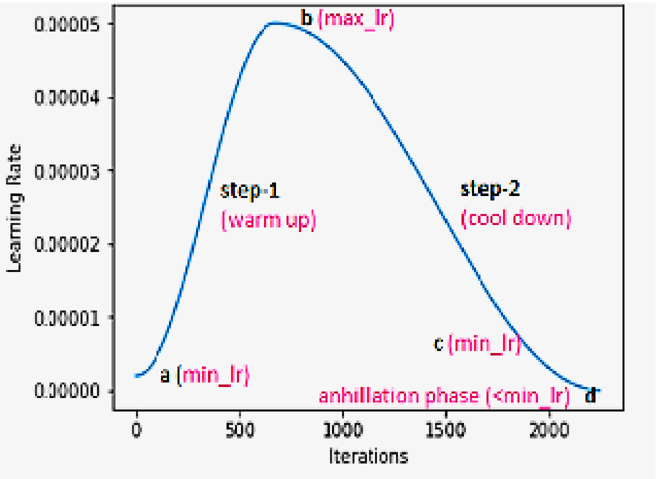


Figure 5.1 Modified one-cycle scheduler

Table 5.1 Summary of training

Training steps	**Dataset used**	**Hyperparameters used**		**Freeze initial layers**	**Freeze later layers**
		Weight initialization	**Learning rate**		
Phase I	COCO	The pre-trained weights of faster-RCNN and mask-RCNN architecture on the COCO dataset is		Not applied	Not applied

		acquired. No training takes place at this stage.			
Phase II	SCD	The model is trained by loading the weights from the Phase I	The parameters are updated at a differential learning rate technique in the network. LR varies from 'init_lr' - (~10-7) to a large value 'max_lr'(~1).	√ The first two stages of the backbone network are freezed	X The model is not freezed at later stages
Phase III	WFD	The model is trained by loading the weights from Phase II		X The model is not freezed at any stage	X The model is not freezed at any stage

5.3 Experimental Set-up and Results

The wrist fracture dataset is analyzed to identify the fracture presence, location, and segmentation mask. The study conducted in the past has involved multiple bones targeting different types of fractures. The researchers have trained and tested their model in their private datasets [23-30]. Additionally, the dataset involving bone fracture images with the annotation files corresponding to the bounding box and segmented mask labels is unavailable in the public domain. Therefore, comparing our proposed methodology to the state-of-the-art dataset is impractical. The experiments are executed in three stages. Stage 1 obtains the weights from the COCO dataset trained on the mask-RCNN model. Stage 2 focuses on training the surface crack dataset using the weight files obtained from stage 1. The last stage is responsible for getting the wrist fracture dataset as input to the proposed architecture and utilizing the weights from stage 2 to detect, localize and segment the fractures accurately.

The model is executed separately for object detection and instance segmentation tasks. The standard COCO metrics utilize Average Precision (AP) at various threshold scales to analyze the results. The concept of Intersection over Union (IoU) is employed to evaluate the performance measure for fracture detection and localization using the AP value. The evaluation criteria, for instance segmentation are similar to those for object detection, with the exception that the IoU of the mask is calculated rather than bounding boxes. By calculating the percentage of overlap between the target mask and the predicted mask, the IoU is determined. The output of the suggested model is contrasted with the radiologists' annotated ground truth label and results from related studies. For fracture detection, an average precision of 92.278% on a scale of 50^0 and 79.003% on a strict scale of 75^0 were reported. For fracture segmentation, an average precision of 77.445% on a scale of 50^0 and 52.156 on a strict scale of 75^0 were reported. The IoU value is calculated using the equation 5.3.

$$IoU = \frac{target\ mask\ \ \cap\ predicted\ mask}{target\ mask\ \cup\ predicted\ mask} \qquad 5.3$$

The bounding box with the largest confidence score is considered True Positive (TP), whereas the remaining predictions are considered False Positive (FP) when multiple bounding box predictions are created for single ground truth, and the IoUs for all of the predictions are larger than the stated threshold. Using equations 5.4 and 5.5, the precision and recall values are computed.

$$Precision = \frac{TP}{TP + FP} \qquad 5.4$$

$$Recall = \frac{TP}{TP + FN\ (all\ ground\ thruths)} \qquad 5.5$$

Finally, using equation 5.6, the Area under the Precision-Recall Curve (AUC) at thresholds 50 and 70 is calculated

$$AP_{\propto} = \int_0^1 p(r)dr \qquad 5.6$$

The l1 loss and binary cross-entropy loss are used to calculate the localization and objectness losses.

$$L = L_{obj} + L_{loc} \qquad 5.7$$

$$L_{obj} = \frac{1}{N_{obj}} \sum_{i} L_{obj}\,(p_i,\, p_i^*) \qquad 5.8$$

$$L_{loc} = \frac{\lambda}{N_{loc}} \sum_{i} p_i^* \cdot k_1^{smooth}\,(\,q_i - q_i^*) \qquad 5.9$$

$$L_{obj}\,(\,p_i\,, p_i^*\,) = -p_i^*\,log\,p_i - (1 - p_i^*\,)log(1 - p_i) \qquad 5.10$$

where q_i is the predicted coordinates of the bounding box and q_i^* is the ground truth coordinates, and p_i^* is the probability of the anchor being an item. The number of anchor locations is set to N_{loc}, the normalization term is set to N_{obj} , and a balancing parameter (~ 10) is used to evenly weight L_{obj} and L_{loc}. Finally, from each feature map level of the image, 1000 region proposals are selected. The predicted bounding boxes are sorted utilizing the objectness score at each level, and the highest-scoring boxes are then obtained via non-max suppression. The segmentation, localization, and object classification masks are combined in the RCNN mask.

$$L = L_{obj} + L_{loc} + L_{mask} \qquad 5.11$$

The mask is constructed with a dimension of $k * n * n$, where k is the total number of classes for each RoI and class. The L_{mask} is calculated as the average binary cross-entropy loss considering only the k^{th} mask into account because the model is constructed to learn a mask for every class regardless of the number of classes.

$$L_{mask} = -\frac{1}{n_2} \sum_{1 \le i,j \le n} y_{ij}\,log\,\hat{y}_{ij}^k + \left(1 - y_{ij}\right) log\left(1 - \hat{y}_{ij}^k\right) \qquad 5.12$$

where i, j are the cell labels and y_{ij} , $\hat{y}_{ij}^i$ are the true and expected masks for the class 'k' region with a size of n x n, respectively.

To produce the results, a differential learning rate strategy is used over 1500 iterations on NVidia K80/T4 for SCD and WFD by keeping the hyperparameters and architecture the same. The computation time for the proposed architecture involving 1497 iterations is 103 minutes and 45 seconds, while the original mask-RCNN architecture took 97 minutes and 31 seconds to complete 1231 iterations. A delay of 6 minutes and 14 seconds is observed because of the modifications done at the backbone architecture.

The results are compared based on two levels: First, the ground truth annotations provided by the expert radiologist were examined to compare the outcomes produced by the proposed model. Table 5.2 lists the findings from various researchers' analyses of bone fracture datasets. The models in the published articles were pre-trained on non-medical datasets first, which were later used to fine-tune the wrist radiographs [23; 26-29]. Instead of using a non-medical dataset to train the model, we developed a surface cracks dataset (SCD), which has crack patterns resembling those of wrist bone fractures. The model was fine-tuned on the wrist fracture dataset after acquiring the pre-trained weights from the SCD.

Second, results obtained due to the modification done at the sub-architecture level are examined. The experiments are conducted and the results are analyzed at three levels (Level-0, level-1 and level-2). Combining the modifications proposed at these level 1 and level 2, we have obtained improved results against the standard mask-RCNN model for the wrist fracture dataset.

At level-0, the experiments are conducted on mask-RCNN architecture using only the wrist fracture dataset, where the surface crack dataset is not used to derive crack-like features. Transfer learning is employed so that the COCO dataset trained on mask-RCNN architecture is used to obtain the pre-trained weights, which are then used as initial parameters before training the model on the wrist dataset. This leads to the average precision of 91.667% and 78.99% for fracture detection and 77.415% and 52.00% for fracture segmentation on 50^0 and 75^0 scales, respectively.

Level-1 modification: The original mask-RCNN architecture is modified at sub-architecture levels by customizing the backbone architecture's last level layer and utilizing the RoIAlign concept, as explained in sections 4.2.1 and 4.2.2, respectively. We also included the surface crack dataset to transfer knowledge from the non-medical COCO dataset to the wrist fracture dataset via surface crack images. Section 4.2 provides a detailed explanation of the knowledge transfer mechanism. On 50^0 and 75^0 scales, fracture detection achieved an average precision of 92.56% and 78.82% while fracture segmentation achieved an average precision of 77.432% and 50.211% respectively.

Level-2 modification: We used the original mask-RCNN architecture, with the initial layers of the backbone architecture frozen, and the experiments were carried out in three stages, as explained in section 5.2. The surface cracks dataset is utilized to transfer knowledge from the non-medical COCO dataset to the wrist fracture dataset. On 50^0 and 75^0 scales, we achieved average precision of 91.553% and 78.08% for fracture detection and 77.421 and 52.10% for fracture segmentation, respectively.

Finally, we integrated the level 1 and level 2 modifications to obtain the improved average precision value. We achieved an average precision of 92.278% and 79.003% for fracture detection and 77.445 and 52.156% for fracture segmentation on 50^0 and 75^0 scales, respectively. The results obtained at various levels are recorded in Table 5.3.

Table 5.2 Analysis of results obtained by the existing articles on wrist fractures for object detection and segmentation.

Ref	Bone type	Architecture type	Input images	FC	FL	FS	Results (%)
[24]	wrist	Extension of unet architecture	34,990	✓	✓	✕	AUC 97.88, 98 on two test sets
[26]	wrist	Inception v3	11,112	✓	✕	✕	AUC- 95.4
[27]	wrist	faster R-CNN	14, 614	✓	✓	✕	AUC (per-study)- 89.5(95% CI: 87.0, 92.0)

[28]	wrist	faster R-CNN	38	✓	✓	✕	mAP – 86.6
[29]	wrist	Inception-ResNet faster-RCNN	7356	✓	✓	✕	Sensitivity- 95.7%, specificity- 82.5%, and AUC- .918
Proposed architecture		Model based on mask-RCNN	315	✓	✓	✓	Fracture detection- $AP_{50\%}$ -92.278 $AP_{75\%}$ -79.003 and Fracture segmentation- $AP_{50\%}$ -77.445 $AP_{75\%}$ - 52.156

Table 5.3 Analysis of results obtained by the proposed architecture on changes made at the architecture levels.

Architecture Type	**Fracture detection (%)**	**Fracture segmentation (%)**
Level-0 modification - Original mask-RCNN architecture	$AP_{50\%}$ -91.667 $AP_{75\%}$ -78.99	$AP_{50\%}$ -77.415 $AP_{75\%}$ - 52.00
Level-1 modification - Proposed model using modified lastlevel layer at backbone architecture and using RoIAlign	$AP_{50\%}$ -92.56 $AP_{75\%}$ -78.82	$AP_{50\%}$ -77.432 $AP_{75\%}$ - 50.211
Level-2 modification - Proposed model by freezing initial layers of the backbone architecture	$AP_{50\%}$ -91.553 $AP_{75\%}$ -78.08	$AP_{50\%}$ -77.421 $AP_{75\%}$ - 52.10
Proposed model combining modifications done at Level 1 and Level 2.	$AP_{50\%}$ -92.278 $AP_{75\%}$ -79.003	$AP_{50\%}$ -77.445 $AP_{75\%}$ - 52.156

Each radiograph is given a ground truth label to test the model's correctness. In Figure 5.2, the outcome of the proposed model is shown in comparison to the radiologists' annotated ground truth label. In order to comprehend the false-negative findings, the misclassified examples shown in Figure 5.3 were examined. Most false-negative instances could not detect fractures due to a lack of training examples and unusually looking anomalies. The deep learning models are data-hungry, and in order to produce the best results, they need a large number of datasets with various types or forms of fractures.

Figure 5.4 displays true-positive examples of surface cracks detected and localized by the network for which the confidence score is provided as a percentage count. Figure 5.5 determines and visualizes the segmentation loss, objectness loss, and localization loss for wrist fracture detection.

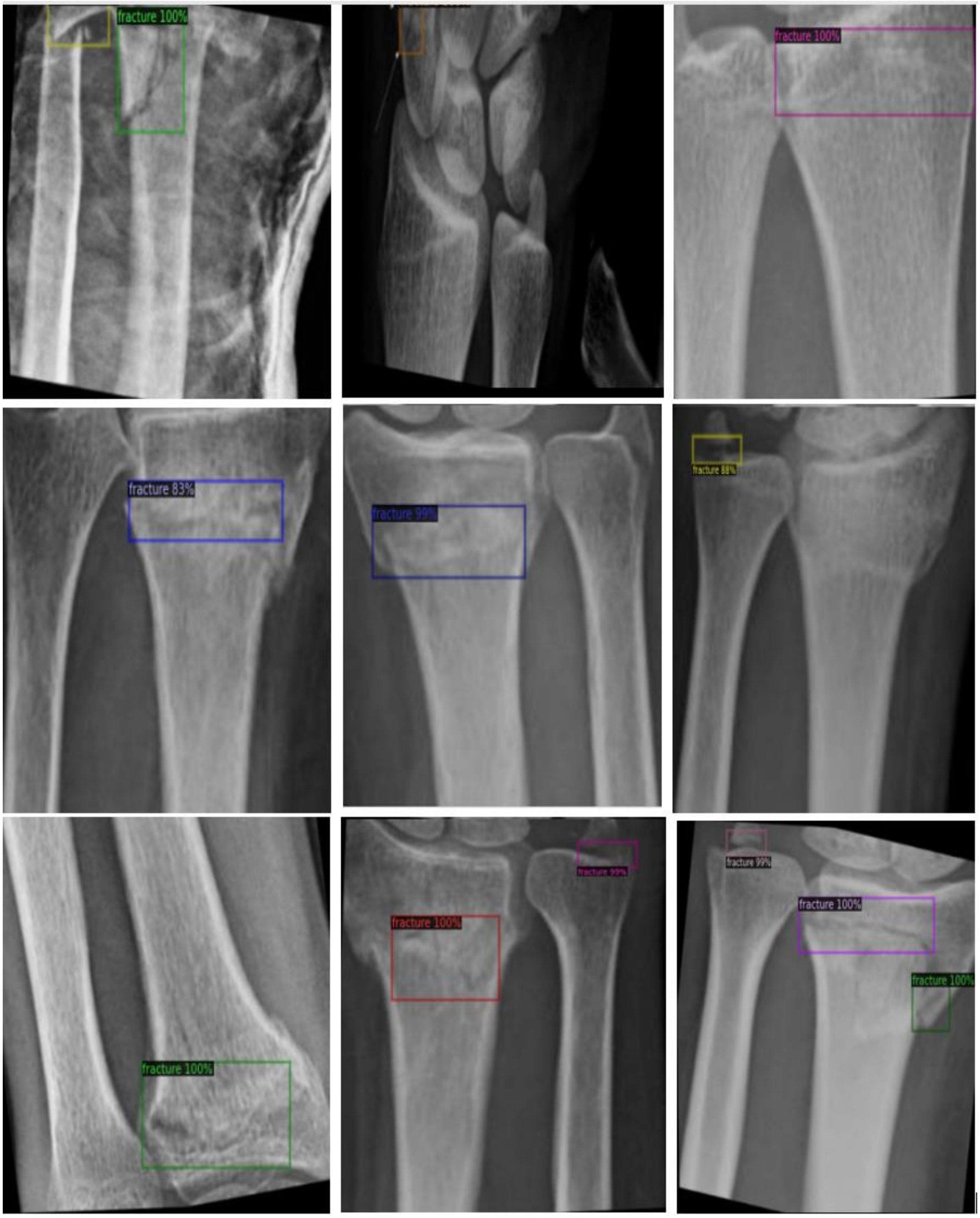


Figure 5.2 (Row 1-3) Radiographs display fractures of the radius and ulna as true-positive cases. The suggested network detects and locates fractures for which a percentage-based confidence score is given.

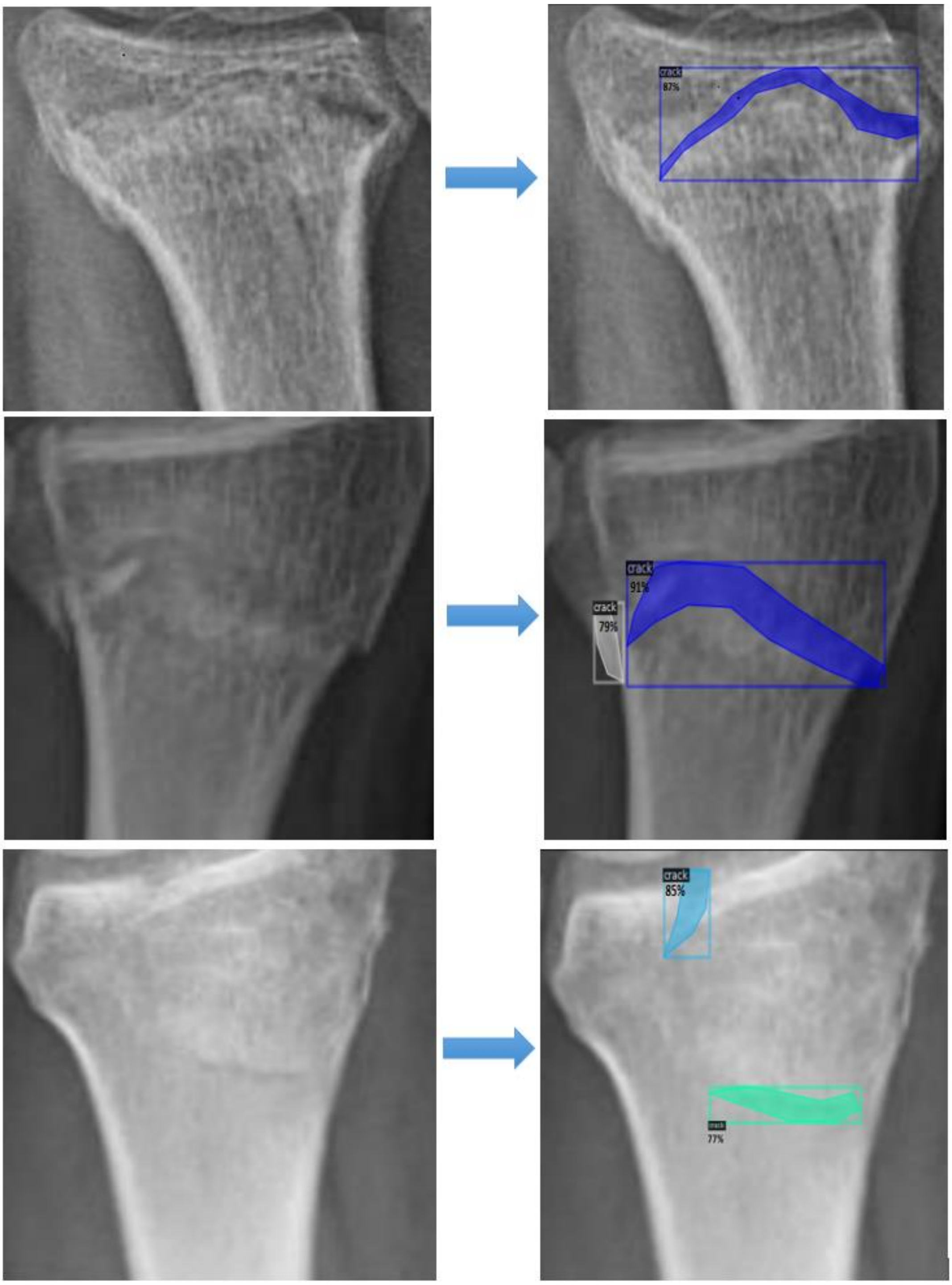


Figure 5.3 (contd.) (Row 1-3) Radiographs display fractures of the radius and ulna as true-positive cases. The suggested network detects, locates and segment fractures for which a percentage-based confidence score is given.

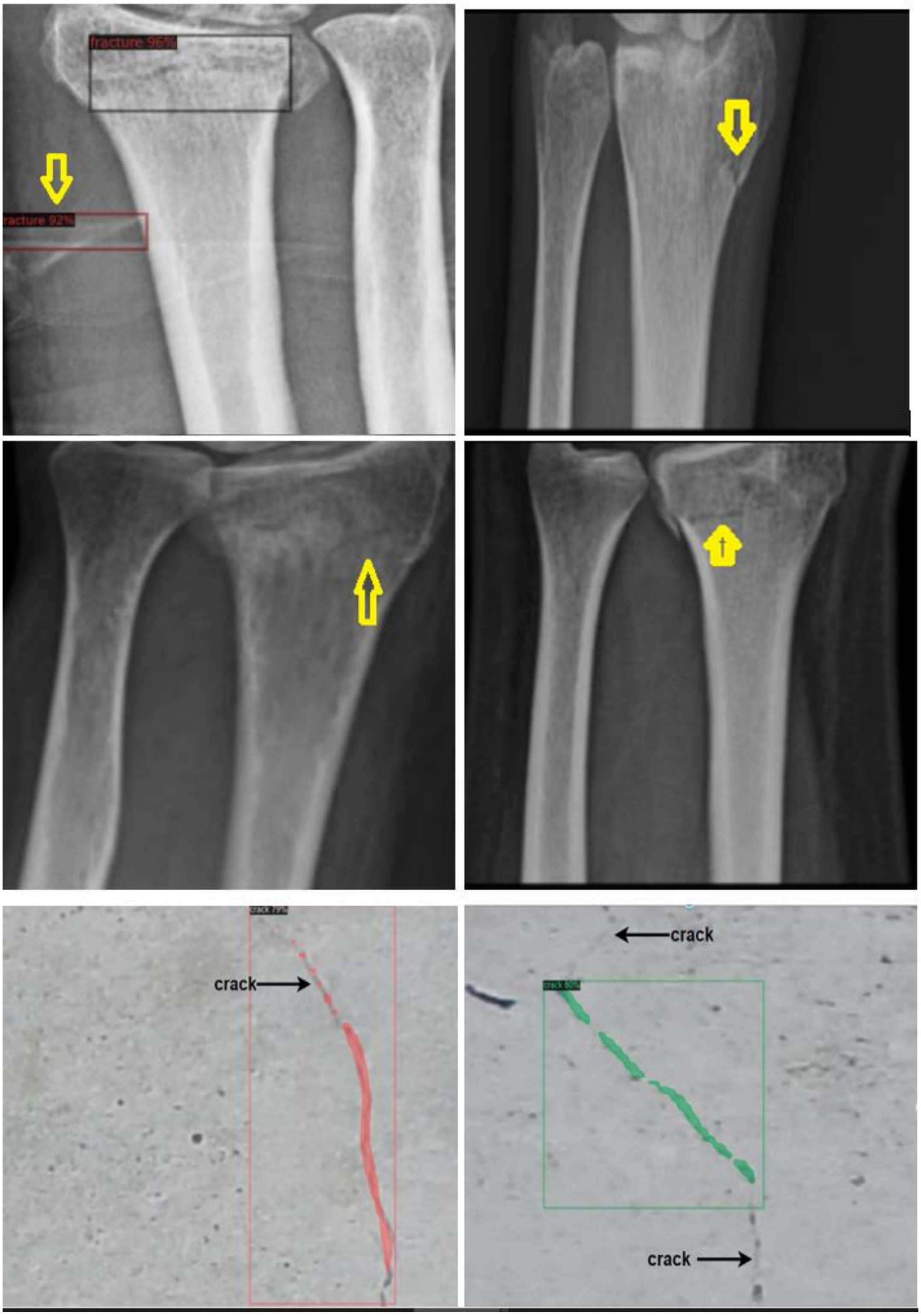


Figure 5.4 (Row 1-2) Radiographs display false negative fracture examples where an arrow indicates the presence of the fracture.
(Row 3) The examples from the surface crack dataset display false negative examples where an arrow indicates the presence of the crack.

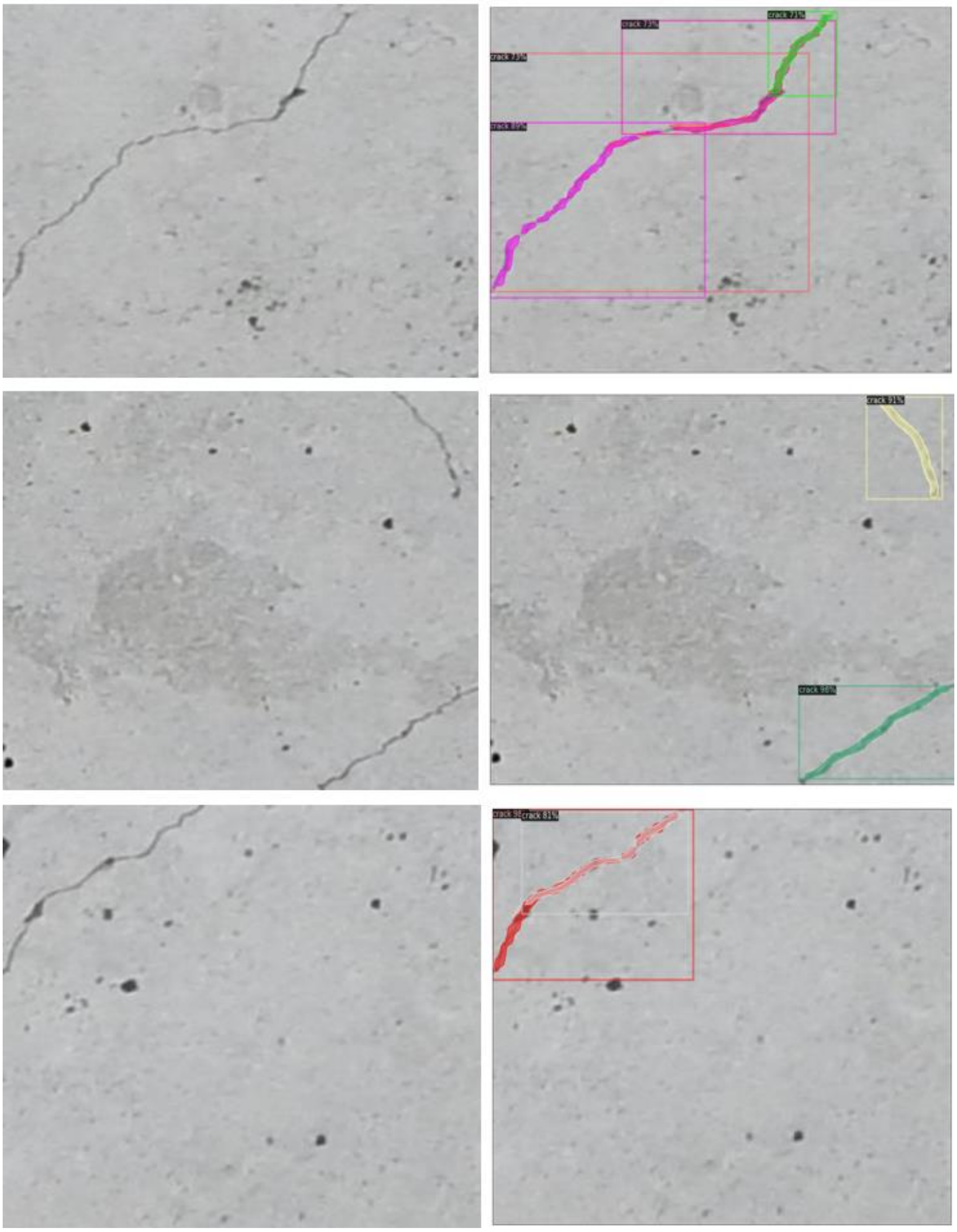


Figure 5.5 Images show true positive examples of surface cracks. The proposed network detects, localizes, and segments the cracks for which the confidence score is provided in the form percentage count.

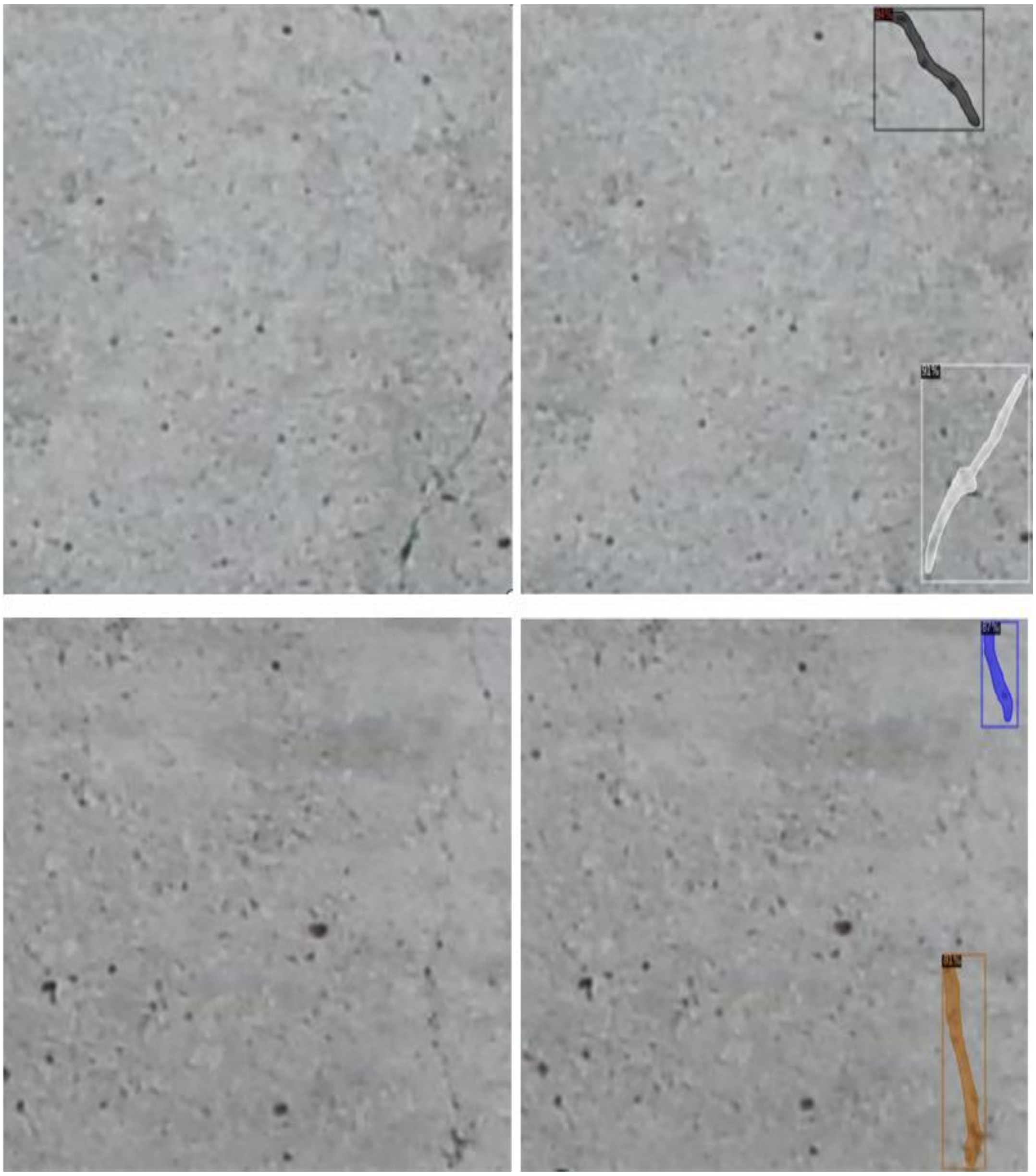


Figure 5.6 (contd.) Images show true positive examples of surface cracks. The proposed network detects, localizes, and segments the cracks for which the confidence score is provided in the form percentage count.

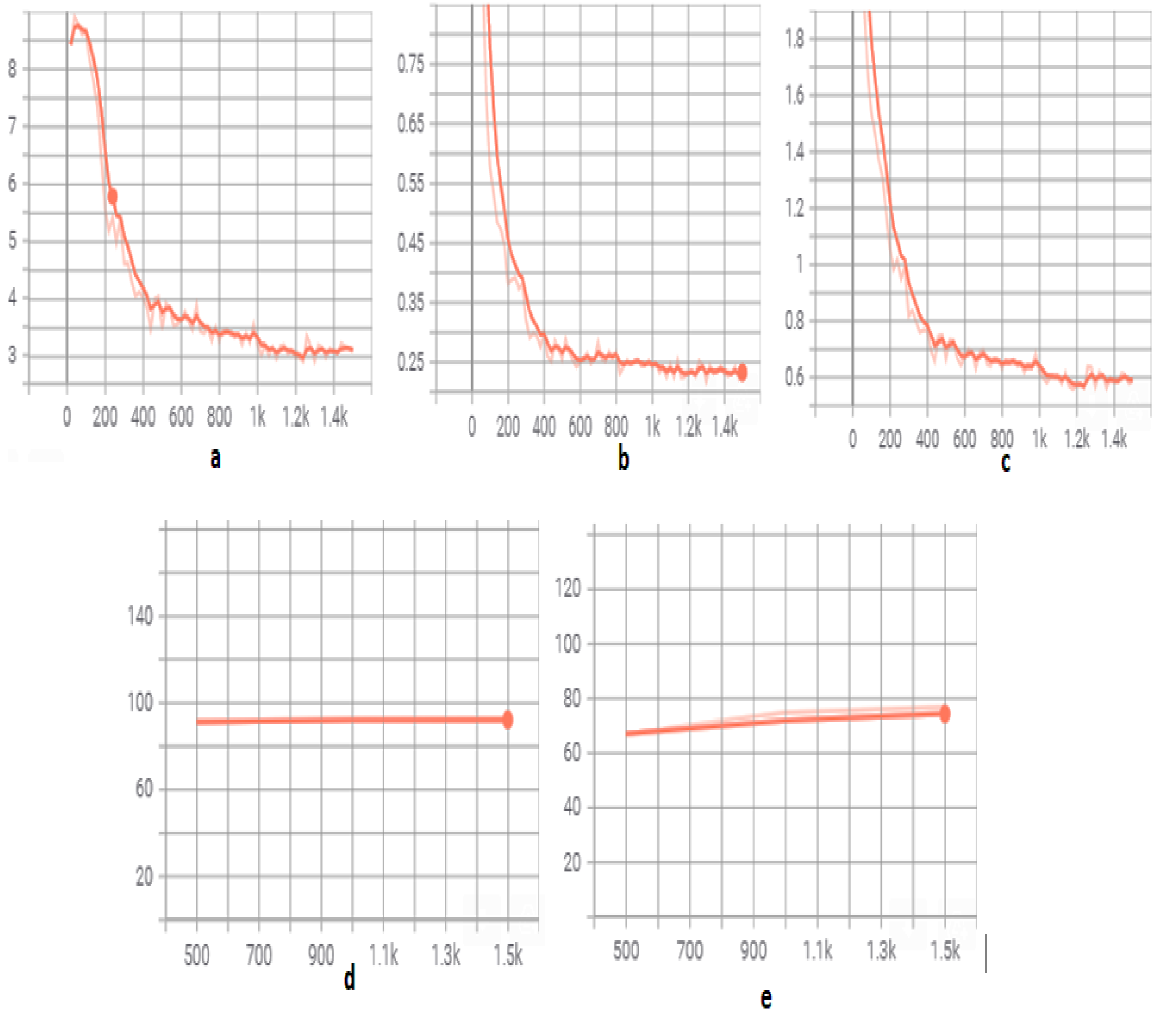


Figure 5.7 (a) Loss box regression, L_{loc} (b) classification loss, L_{obj} (c) Total loss for fracture detection model (d) AP_{50}
Fracture localization (e) AP_{50} Fracture segmentation

5.4 Conclusion

The proposed model is tuned by selecting the appropriate parameters and hyperparameters significant for analysis and experimentation. The parameters selected during model training at sub-architecture levels, such as the number of filters, filter size, activation function, pooling size, etc., are discussed. The open-source pre-trained weight file of the COCO dataset trained on the mask-RCNN model with Resnet-50 as the feature extractor is utilized to fine-tune the model on the surface crack dataset.

The model is frozen at the initial layers of the training, which means the parameters are not updated during the backpropagation of the model for the freezed layers. This is done because the initial layers of the CNN architecture are meant to obtain generic features. The first layer of CNN is intended to identify simple gradients of the line, the second layer finds simple shapes, and the third layer finds the combinations of lines and shapes. On the other hand, the final layers concentrate more on the particular task at hand, like identifying the image's crack patterns.

The pre-trained COCO dataset is used to fine-tune the proposed architecture on the surface crack dataset. The same step is repeated for fine-tuning the wrist fracture dataset using the weight file obtained from the SCD. Because the features predicted by the initial layers of a CNN architecture will be the same regardless of the dataset used, it is unlikely that better features will be generated at the initial layers while updating the gradients at the same learning rate. Considering the same reason, the initial layers were frozen while training the model. The learning rate is kept different for different architecture sections, computed using a differential learning rate strategy. Firstly, the data is trained batch-wise, where the learning rate gradually rises from an exponentially low value (10^{-7}) to a high value (1). Next, the learning rate is selected using a learning rate finder curve to update the network parameters.

The model is executed separately for object detection and instance segmentation tasks of the self-collected Surface crack and wrist bone datasets. The standard COCO metric Average Precision (AP) is employed at threshold 50^0 and 75^0 to analyze the results using Intersection over Union (IoU). If the IoU of the predicted crack label is greater than 50^0, then only the crack is said to be detected else not. Similarly, a strict scale of 75^0 is used to identify, localize and segment the cracks in SCD and WFD.

The researchers have trained and tested their model in their private datasets. Therefore, evaluating the proposed model performance on the state-of-the-art dataset was not feasible. The results are compared based on two levels: First, the ground truth annotations provided by the expert radiologist were examined to compare the outcomes produced by the proposed model. Second, results obtained due to the modification done at the sub-architecture level are examined. For fracture detection, an average precision of 92.278% on a scale of 50^0 and 79.003% on a strict scale of 75^0 was reported. For fracture segmentation, an average precision of 77.445% on a scale of 50^0 and 52.156% on a strict scale of 75^0 was reported.

CHAPTER -6

CONCLUSIONS AND FUTURE SCOPE

In radiology, computer-assisted detection (CAD) has historically failed to improve diagnostic precision, lowering clinician sensitivity and leading to unnecessary additional diagnostic tests. The ability to identify fractures in radiographs is crucial for clinical purposes. In 2012, there was a predicted backlog of 12,000 cross-sectional studies and 200,000 plain radiographs [9]. These numbers call for urgent workflow management improvements and reporting efficiency to reduce the harm that delayed or missed diagnoses cause to patients. Radiologists could benefit greatly from automatic detection-based or localization techniques in their fight against fatigue.

According to the type of abnormality detected by the automated system, radiologists can further prioritize diagnosis and treatment. Non-orthopaedic surgeons or novice medical professionals who are untrained in fracture detection are often the first sources of contact for any patient in the event of a fracture. Therefore, it is quite common for fractures to be incorrectly identified during X-ray images interpretation.

The primary driving force behind this research was to propose a fracture detection architecture based on deep learning methods with superior accuracy and minimal complexity. This chapter highlights our major contributions to bone fracture detection using machine learning and computer vision models and discusses possible future research directions.

6.1 Summary of the Thesis

In this thesis, we presented a deep learning model to be applied to wrist bone X-rays to detect and segment radius and ulna fractures. We provided two datasets of wrist bone X-rays and surface cracks, together with the pixel-level labels correlating to them. These datasets were utilized for training our deep learning models. For other researchers working in this field, we have made a portion of the dataset available in public domain [118]. The main research objectives mentioned in Chapter 1, 1.3 have been addressed in this thesis in the following order:

First, we propose two novel datasets: The wrist fracture dataset (WFD) and the Surface crack dataset (SCD). For studies carried out between February 2019 and March 2020, the Doon Hospital, Dehradun, India, provided anonymized wrist radiographs. The dataset was obtained without revealing the participant's identity or demographic information under the Ethical Conduct in Human Research and Related Activities Regulations. Instead of pretraining the model on a non-medical dataset, we have incorporated a surface crack dataset with similar crack patterns to wrist bone fractures. SCD consists of pictures taken from walls, pavements, and roads, created using a mobile camera. Both SCD and WFD are preprocessed before feeding them to the network. The WFD is cropped to exclude the finger bone regions from the hand X-rays, followed by removing the "red spot" annotations from the image. Next, we converted the DICOM image to the 24-bit lossless JPEG format while ensuring the best windowing was chosen under the doctor's supervision. Afterward, the images are undergone a labeling process followed by augmentation. All the images in our datasets are manually labeled, which was time-consuming but less error-prone than the automatic annotation software. The radiologists have utilized LabelMe software to annotate the wrist bone images. The skilled radiologist labels the wrist fractures by tracing a box around the fracture. Drawing a bounding box may not accurately depict the shape of the fractures, as it involves non-essential bone areas when the model is being trained. In order to further label each image, a mask is made by drawing a more intricate shape, such as a polygon, around the crack. We manually annotate the SCD for the two distinct tasks of instance segmentation and object detection. A portion of the dataset is made publicly available for research to circumvent data collection challenges and wrist fracture labeling [118].

Second, we demonstrate a novel fracture localization and segmentation model comprised of three sub-architectures: the Backbone network, the Region Proposal Network (RPN), and RoIAlign. The fractures are localized and segmented based on the instance segmentation technique, which integrates faster-RCNN and semantic segmentation. Instance segmentation combines classification, localization, and segmentation tasks into a single output, a polygon mask encircling the defined target. We have used Instance segmentation which detects segments and classifies every fracture in the image by assigning a label to an individual image pixel. According to our knowledge, this study is the first one to concentrate on creating a segmentation mask around the class labels to detect wrist fractures. The intuition behind involving segmentation along with the localization of wrist fractures is better visualization of the shape of the fractures. It has been observed that the fracture shape is extended in a vertical and horizontal direction in most of the X-ray samples collected. The automatic localization of the fractures is improved by creating a segmented mask along with the bounding box to locate the fractures and their shape.

The backbone network consisting of a top-down and bottom-up pathway is responsible for extracting semantically powerful features (p2-p6) from the input image. The top-down pathway inspired by ResNet-50 architecture is intended for feature extraction. The bottom-up pathway inspired by the feature pyramid network (FPN) combines low-resolution, semantically powerful features with high-resolution, semantically weak features. To identify large and small objects, we have created a pyramid using the same image at various scales. We modified the backbone architecture's last-level max-pool layer by replacing it with a linear combination of ACP (AdaptiveConcatPool), AMP (AdaptiveMaxPool), and AAP (AdaptiveAvgPool) layers. The pooling layers (AAP, ACP, AMP) are used as a transitional phase to connect the convolution layer to the fully connected layers in the modified architecture. In our investigations, the maximum and average activations from the previous convolution are preserved, offering the model knowledge of both the approaches and enhancing performance.

The smaller and larger objects are detected next using the fracture localization and segmentation techniques. This stage's sub-architecture comprises two networks- RPN (Region Proposal Network) and the RoIAlign layer. RPN applies a sliding window on the predicted feature maps (p2-p6) to identify the objectness logits and anchor deltas of the object. Anchor generation is another step toward object detection, which helps connect the pred_obj and pred_anchor to the ground truth boxes. Anchor sizes (32, 64, 128, 256, and 512) are used for (p2-p6) feature maps generated from the backbone architecture. The aspect ratio of 2:1, 1:1, and 1:2 is set for defining the shape of the anchors. The intersection over union (IoU) is computed using the generated anchors and the ground truth boxes. The objective is to find anchors similar to the ground truth boxes out of 1,20,015 anchors using the concept of IoU. The resampling is done as the number of anchors generated per image is 1,20,015, where most anchors are background. In our experiments, less than 100 anchors are the foreground, less than 1000 anchors are ignored, and the remaining are the background. Localization loss or l1 loss is computed by ignoring all background labels. Finally, 1000 region proposal boxes are selected by applying the predicted anchor deltas to the corresponding anchors and sorting them based on the objectness score individually for (p2-p6) feature maps. Next, NMS (Non-max suppression) is applied to selected top-scored 1000 boxes.

In RoI pooling, the proposal boxes-specified feature maps are cropped into rectangular regions of interest (RoIs). The RoI is cropped from the feature maps by allocating proposal boxes to the relevant feature maps. A new technique (RoIAlignv2) is adopted for cropping the region of interest precisely using a modified version of ROIAlign. The neighboring indices are accurately computed by subtracting the half-pixel offset (0.5) from ROI coordinates. This method has overcome the disadvantages of choosing a slightly off-aligned pixel value while using bilinear interpolation. Next, a parallel layer is added with the existing object detection framework to generate the mask around the cracks. The sub-architecture in this layer consists of a convolutional layer to extract features, generating pixel-to-pixel mapping. Next, another convolutional layer is added, followed by RoIPool and RoIAlign layers to obtain the bounding box for the classification and regression task.

Third, transfer learning is utilized in the proposed methodology to transfer the knowledge from the non-medical (COCO) dataset to the wrist-fracture (medical-domain) dataset via the surface crack dataset. The proposed approach does not directly use transfer learning on the wrist fracture dataset. The wrist dataset is fine-tuned using a surface crack dataset with similar crack patterns. Before that, the surface crack dataset was fine-tuned using the state-of-the-art COCO dataset.

Fourth, the proposed model is tuned by selecting the appropriate parameters and hyperparameters significant for analysis and experimentation. The pre-trained weights of the COCO dataset obtained from Phase I of the proposed methodology are transferred to Phase II for surface crack detection. The weight file from the pre-trained mask-RCNN model is incorporated rather than applying random weight initialization techniques in phase I. We used the notion of freezing and unfreezing certain layers of the proposed architecture. During the second phase of training, the proposed model's initial layers are frozen (not trained). It is unlikely to generate better features at the initial layers while updating the gradients at the same learning rate because the features predicted by the initial layers of a CNN architecture will be the same irrespective of the dataset used. The entire architecture is then unfrozen and trained in the third phase by updating the learned parameters. Using a learning rate finder curve, the network updates the parameters at a differential learning rate. This method employs different learning rates for deep neural architecture segments.

6.2 Future Work

The proposed deep learning model for fracture detection in bone X-rays provides better detection accuracy than the baseline model due to the effectiveness of deep learning-based models in obtaining better accuracy in computer vision techniques. However, the research presented here have wider scope with several extensions addressing variety of challenges that require future attention, as is the case with many other academic articles in the same field. In the part that follows, we go through some of these issues and suggest upcoming directions that, in our opinion, will have a significant influence.

Utilizing a trained model to forecast the fracture takes a few milliseconds on a modern computer. Though, we have presented two datasets in this work but it is a laborious operation in medical imaging to gather hundreds of thousands of radiographs, give accurate labels to these X-ray images, and feed adequate training data to the models. A promising solution to the unavailability of the large dataset is to improve a CNN that has already been trained on a different network. These pre-trained models enable researchers to gain the very sophisticated and potent features required for the topic of interest. The model can be trained on numerous images rather than millions of non-radiology images. The model can be trained on bone X-ray images, such as those of the ankle, knee, neck, hip, and other bones, rather than being pre-trained on millions of non-radiology images. By doing so, we could more effectively set up the model parameters that will be utilized to train the required X-ray images later on. With the small exception that we pre-train the model on various types of bone X-ray images rather than non-medical images, this method is comparable to transfer learning. When given enough training data, a machine like CNN can perform consistently and even outperform humans in terms of the ability to interpret a variety of complex X-ray structures. Furthermore, people tend to predict the correct outcome in shapes with which they are familiar rather than those with which they are unfamiliar with the fracture configuration. Therefore, a CNN could potentially be trained with enormous amounts of training data that includes all possible cases, from those with simpler fracture structures to those with the most complex ones, more than any orthopedics will ever encounter in their lives.

This proposed work focuses on constructing the segmentation mask around wrist fractures using the integration of faster-RCNN and semantic segmentation. Till now, most published work on fracture detection and classification emphasizes on a single anatomical region or a single type of fracture in many anatomical regions [23-30] [106-115]. A model capable of identifying diverse fracture types in various anatomical regions would be ideal. In order to detect fractures, more than one body region, such as the wrist, may be inspected. At the moment, wrist fracture diagnosis's only capabilities are localizing and providing a segmentation mask [23-30]. However, depending on its location, it would be preferable if we could determine the type of wrist fracture, which could be a lunate, scaphoid, radius, or ulna. Furthermore, depending on the pattern, the fracture can be transverse, undisplaced, comminuted, and so on. Our research is limited to identifying, localizing and segmenting fractures, which can be extended to categorize fracture types based on their location and pattern in the image.

The proposed architecture's performance is compared on two levels: First, the expert radiologist's ground truth annotations were compared to the outcomes generated by the proposed model. Second, the results obtained from the sub-architecture modification are examined. When a classifier is compared to actual or known data provided by the radiologist, the radiologist's performance is recognized as 100% accurate. This indicates that the radiologist interpreting hundreds of images finds and categorizes fractures in all samples with a zero mistake rate. Cohen's kappa statistic was utilized in 2017 to develop a *Gold standard* for abnormality detection in X-ray samples from various anatomical locations [103]. Three experts are chosen randomly from a pool of six board-certified radiologists to constitute the gold standard, and a majority vote determines the label. Similarly, a globally recognized gold standard for a labeled fracture dataset might be produced for fracture detection and classification. This would allow academics to compare the performance of the proposed model to industry best practices.

REFERENCES

[1] WHO Study Group on Assessment of Fracture Risk, & its Application to Screening for Postmenopausal Osteoporosis. (1994). *Assessment of fracture risk and its application to screening for postmenopausal osteoporosis* (No. 843-849). World Health Organization.

[2] *how to identify fracture types – All Things AAFS!* (2013, August 7). All Things AAFS! https://allthingsaafs.com/tag/how-to-identify-fracture-types.

[3] *X-rays, CT Scans and MRIs - OrthoInfo - AAOS*. (2017, June 1). X-Rays, CT Scans and MRIs - OrthoInfo - AAOS. https://orthoinfo.aaos.org/en/treatment/x-rays-ct-scans-and-mris.

[4] Guly, H. R. (2001). Diagnostic errors in an accident and emergency department. *Emergency Medicine Journal*, *18*(4), 263-269.

[5] Whang, J. S., Baker, S. R., Patel, R., Luk, L., & Castro III, A. (2013). The causes of medical malpractice suits against radiologists in the United States. *Radiology*, *266*(2), 548-554. https://doi.org/10.1148/radiol.12111119.

[6] Krupinski, E. A., Berbaum, K. S., Caldwell, R. T., Schartz, K. M., & Kim, J. (2010). Long Radiology Workdays Reduce Detection and Accommodation Accuracy. *Journal of the American College of Radiology*, 7(9), 698–704. https://doi.org/10.1016/j.jacr.2010.03.004.

[7] Waite, S., Scott, J., Gale, B., Fuchs, T., Kolla, S., & Reede, D. (2017). Interpretive Error in Radiology. *American Journal of Roentgenology*, 208(4), 739–749. https://doi.org/10.2214/ajr.16.16963.

[8] Stec, N., Arje, D., Moody, A. R., Krupinski, E. A., & Tyrrell, P. N. (2018). A Systematic Review of Fatigue in Radiology: Is It a Problem? *American Journal of Roentgenology*, 210(4), 799–806. https://doi.org/10.2214/ajr.17.1861

[9] Joshi, D., & Singh, T. P. (2020). A survey of fracture detection techniques in bone X-ray images. *Artificial Intelligence Review*. https://doi.org/10.1007/s10462-019-09799-0

[10] Joshi, D., Anwarul, S., & Mishra, V. (2020). Deep Leaning Using Keras. In Machine Learning and Deep Learning in Real-Time Applications (pp. 33-60). IGI Global.

[11] Donnelley, M., Knowles, G., & Hearn, T. (2008, July). A CAD system for long-bone segmentation and fracture detection. *In International Conference on Image and Signal Processing* (pp. 153-162). Springer, Berlin, Heidelberg.

[12] Peterson, L. E. (2009). K-nearest neighbor. *Scholarpedia*, 4(2), 1883.

[13] Noble, W. S. (2006). What is a support vector machine?. *Nature biotechnology*, 24(12), 1565-1567.

[14] Khatik, I. (2017). A study of various bone fracture detection techniques. *Int J Eng Comput Sci*, *6*(5), 21418-21423. doi:10.18535/ijecs/v6i5.38.

[15] Zhuang, F., Qi, Z., Duan, K., Xi, D., Zhu, Y., Zhu, H., ... & He, Q. (2020). A comprehensive survey on transfer learning. *Proceedings of the IEEE*, *109*(1), 43-76.

[16] Yap, D. W. H., Chen, Y., Leow, W. K., Howe, T. S., & Png, M. A. (2004, August). Detecting femur fractures by texture analysis of trabeculae. *In Proceedings of the 17th International Conference on Pattern Recognition*, 2004. ICPR 2004. (Vol. 3, pp. 730-733). IEEE. doi:10.1109/ICPR.2004.1334632.

[17] Lim, S. E., Xing, Y., Chen, Y., Leow, W. K., Howe, T. S., & Png, M. A. (2004, September). Detection of femur and radius fractures in x-ray images. In *Proc. 2nd Int. Conf. on Advances in Medical Signal and Info. Proc* (Vol. 65).

[18] Lum, V. L. F., Leow, W. K., Chen, Y., Howe, T. S., & Png, M. A. (2005, September). Combining classifiers for bone fracture detection in X-ray images. In *IEEE International Conference on Image Processing 2005* (Vol. 1, pp. I-1149). IEEE.). doi: 10.1109/ICIP.2005.1529959.

[19] He, J. C., Leow, W. K., & Howe, T. S. (2007, August). Hierarchical classifiers for detection of fractures in X-ray images. In *International*

conference on computer analysis of images and patterns (pp. 962-969). Springer, Berlin, Heidelberg. doi:10.1007/978-3-540-74272-2_119.

[20] Tian, T. P., Chen, Y., Leow, W. K., Hsu, W., Howe, T. S., & Png, M. A. (2003, August). Computing neck-shaft angle of femur for x-ray fracture detection. In *International Conference on Computer Analysis of Images and Patterns* (pp. 82-89). Springer, Berlin, Heidelberg.

[21] Tian, D. Z., & Ha, M. H. (2004, August). Applications of wavelet transform in medical image processing. In *Proceedings of 2004 international conference on machine learning and cybernetics (IEEE Cat. No. 04EX826)* (Vol. 3, pp. 1816-1821). IEEE.

[22] Haralick, R. M., Shanmugam, K., & Dinstein, I. H. (1973). Textural features for image classification. *IEEE Transactions on systems, man, and cybernetics*, (6), 610-621.

[23] Gale, W., Oakden-Rayner, L., Carneiro, G., Bradley, A. P., & Palmer, L. J. (2017). Detecting hip fractures with radiologist-level performance using deep neural networks. *arXiv preprint arXiv:1711.06504*. http://arxiv.org/abs/1711.06504.

[24] Lindsey, R., Daluiski, A., Chopra, S., Lachapelle, A., Mozer, M., Sicular, S., ... & Potter, H. (2018). Deep neural network improves fracture detection by clinicians. Proceedings of the National Academy of Sciences, 115(45), 11591-11596. doi:10.1073/pnas.1806905115

[25] Chung, S. W., Han, S. S., Lee, J. W., Oh, K. S., Kim, N. R., Yoon, J. P., ... & Kim, Y. (2018). Automated detection and classification of the proximal humerus fracture by using deep learning algorithm. *Acta orthopaedica*, *89*(4), 468-473. doi:10.1080/17453674.2018.1453714.

[26] Kim, D. H., & MacKinnon, T. (2018). Artificial intelligence in fracture detection: transfer learning from deep convolutional neural networks. *Clinical radiology*, *73*(5), 439-445. doi: 10.1016/j.crad.2017.11.015.

[27] Olczak, J., Fahlberg, N., Maki, A., Razavian, A. S., Jilert, A., Stark, A., ... & Gordon, M. (2017). Artificial intelligence for analyzing orthopedic

trauma radiographs: deep learning algorithms—are they on par with humans for diagnosing fractures?. *Acta orthopaedica*, *88*(6), 581-586.

[28] Yahalomi, E., Chernofsky, M., & Werman, M. (2019, July). Detection of distal radius fractures trained by a small set of X-ray images and Faster R-CNN. In *Intelligent Computing-Proceedings of the Computing Conference* (pp. 971-981). Springer, Cham.

[29] Thian, Y. L., Li, Y., Jagmohan, P., Sia, D., Chan, V. E. Y., & Tan, R. T. (2019). Convolutional neural networks for automated fracture detection and localization on wrist radiographs. Radiology. Artificial intelligence, 1(1).

[30] Raisuddin, A. M., Vaattovaara, E., Nevalainen, M., Nikki, M., Järvenpää, E., Makkonen, K., ... & Tiulpin, A. (2021). Critical evaluation of deep neural networks for wrist fracture detection. *Scientific reports*, *11*(1), 1-11.

[31] Mahendran, S. K., & Baboo, S. S. (2011). An enhanced tibia fracture detection tool using image processing and classification fusion techniques in X-ray images. *Global Journal of Computer Science and Technology*, *11*(14), 22-28.

[32] Dimililer, K. (2017). IBFDS: intelligent bone fracture detection system. *Procedia computer science*, *120*, 260-267. https://doi.org/10.1016/j.procs.2017.11.237.

[33] Chai, H. Y., Wee, L. K., Swee, T. T., & Hussain, S. (2011). Gray-level co-occurrence matrix bone fracture detection. *WSEAS Transactions on Systems*, *10*(1), 7-16. doi:10.3844/ajassp.2011.26.32.

[34] Umadevi, N., & Geethalakshmi, S. N. (2012, July). Multiple classification system for fracture detection in human bone x-ray images. In 2012 Third International Conference on Computing, Communication and Networking Technologies (ICCCNT'12) (pp. 1-8). IEEE. doi: 10.1109/ICCCNT.2012.6395889

[35] Al-Ayyoub, M., Hmeidi, I., & Rababah, H. (2013). Detecting Hand Bone Fractures in X-Ray Images. *J. Multim. Process. Technol.*, *4*(3), 155-168. doi: 10.13140/RG.2.1.2645.8327.

[36] Al-Ayyoub, M., & Al-Zghool, D. (2013). Determining the type of long

bone fractures in x-ray images. *WSEAS Transactions on Information Science and Applications*, *10*(8), 261-270.

[37] Myint, W. W., Tun, K. S., & Tun, H. M. (2018). Analysis on leg bone fracture detection and classification using X-ray images. *Machine Learning Research*, *3*(3), 49-59. doi:10.11648/j.mlr.20180303.11.

[38] Myint, W. W., Tun, K. S., & Tun, H. M. (2018). Analysis on leg bone fracture detection and classification using X-ray images. *Machine Learning Research*, *3*(3), 49-59. doi:10.1007/BF03178082.

[39] Rajan, J., & Kaimal, M. R. (2006). Image denoising using wavelet embedded anisotropic diffusion (WEAD). doi: 10.1049/cp:20060597.

[40] Hari, C. V., Jojish, J. V., Gopi, S., Felix, V. P., & Amudha, J. (2009, December). Mid-point Hough transform: A fast line detection method. In *2009 Annual IEEE India Conference* (pp. 1-4). IEEE. Doi: 10.1109/INDCON.2009.540945.

[41] Teixeira, L., Celes, W., & Gattass, M. (2009). Accelerated corner-detector algorithms. doi: 10.5244/C.22.62.

[42] Khashman, A., & Dimililer, K. (2008). Image compression using neural networks and Haar wavelet. *WSEAS Transactions on Signal Processing*, *4*(5), 330-339.

[43] Antipov, G., Berrani, S. A., & Dugelay, J. L. (2016). Minimalistic CNN-based ensemble model for gender prediction from face images. *Pattern recognition letters*, *70*, 59-65. doi: https://doi.org/10.1016/j.patrec.2015.11.011.

[44] Minetto, R., Segundo, M. P., & Sarkar, S. (2019). Hydra: An ensemble of convolutional neural networks for geospatial land classification. *IEEE Transactions on Geoscience and Remote Sensing*, *57*(9), 6530-6541.

[45] Ding, C., & Tao, D. (2017). Trunk-branch ensemble convolutional neural networks for video-based face recognition. *IEEE transactions on pattern analysis and machine intelligence*, *40*(4), 1002-1014. doi:10.1109/TPAMI.2017.2700390.

[46] Kumar, A., Kim, J., Lyndon, D., Fulham, M., & Feng, D. (2016). An

ensemble of fine-tuned convolutional neural networks for medical image classification. *IEEE journal of biomedical and health informatics*, *21*(1), 31-40. doi:10.1109/JBHI.2016.2635663.

[47] Wang, H. Z., Li, G. Q., Wang, G. B., Peng, J. C., Jiang, H., & Liu, Y. T. (2017). Deep learning based ensemble approach for probabilistic wind power forecasting. *Applied energy*, *188*, 56-70. doi: https://doi.org/10.1016/j.apenergy.2016.11.111.

[48] Töscher, A., Jahrer, M., & Bell, R. M. (2009). The bigchaos solution to the netflix grand prize. *Netflix prize documentation*, 1-52.

[49] Polikar, R. Ensemble learning. Scholarpedia 4 (1), 2776 (2009). doi:10.4249/scholarpedia.2776.

[50] Quinlan, J. R. (1996, August). Bagging, boosting, and C4. 5. In *Aaai/Iaai, vol. 1* (pp. 725-730).

[51] Kuncheva, L. I., Skurichina, M., & Duin, R. P. (2002). An experimental study on diversity for bagging and boosting with linear classifiers. *Information fusion*, *3*(4), 245-258.

[52] Maclin, R., & Opitz, D. (1997). An empirical evaluation of bagging and boosting. *AAAI/IAAI*, *1997*, 546-551.

[53] Schapire, R. E. (1999, July). A brief introduction to boosting. In *Ijcai* (Vol. 99, pp. 1401-1406).

[54] Schapire, R. E. (2003). The boosting approach to machine learning: An overview. *Nonlinear estimation and classification*, 149-171.

[55] Schapire, R. E., & Freund, Y. (2013). Boosting: Foundations and algorithms. *Kybernetes*.

[56] Wolpert, D. H. (1992). Stacked generalization. *Neural networks*, *5*(2), 241-259. doi:10.1016/S0893- 6080(05)80023-1.

[57] Van der Laan, M. J., Polley, E. C., & Hubbard, A. E. (2007). Super learner. *Statistical applications in genetics and molecular biology*, *6*(1)

[58] Cao, Y., Wang, H., Moradi, M., Prasanna, P., & Syeda-Mahmood, T. F. (2015, April). Fracture detection in x-ray images through stacked random forests feature fusion. In *2015 IEEE 12th international symposium on*

biomedical imaging (ISBI) (pp. 801-805). IEEE. doi:10.1109/ISBI.2015.7163993.

[59] Lampert, C. H., Blaschko, M. B., & Hofmann, T. (2008, June). Beyond sliding windows: Object localization by efficient subwindow search. In *2008 IEEE conference on computer vision and pattern recognition* (pp. 1-8). IEEE. doi: 10.1109/CVPR.2008.4587586.

[60] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, *60*(6), 84-90. doi:10.1145/3065386.

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

[62] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., ... & Rabinovich, A. (2015). Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1-9).

[63] 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).

[64] Lin, M., Chen, Q., & Yan, S. (2013). Network in network. *arXiv preprint arXiv:1312.4400*.

[65] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., ... & Fei-Fei, L. (2015). Imagenet large scale visual recognition challenge. *International journal of computer vision*, *115*(3), 211-252.

[66] LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, *86*(11), 2278-2324. doi: 10.1109/5.72679.

[67] Zeiler, M. D., & Fergus, R. (2014, September). Visualizing and understanding convolutional networks. In *European conference on computer vision* (pp. 818-833). Springer, Cham.

[68] Quinn, J., McEachen, J., Fullan, M., Gardner, M., & Drummy, M. (2019). *Dive into deep learning: Tools for engagement*. Corwin Press.

[69] He, K., & Sun, J. (2015). Convolutional neural networks at constrained

time cost. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 5353-5360).

[70] He, K., & Sun, J. (2015). Convolutional neural networks at constrained time cost. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 5353-5360).

[71] Taigman, Y., Yang, M., Ranzato, M., & Wolf, L. Closing the gap to human-level performance in face verification. deepface. In *Proceedings of the IEEE Computer Vision and Pattern Recognition (CVPR)* (Vol. 5, p. 6).

[72] Ouyang, W., Wang, X., Zeng, X., Qiu, S., Luo, P., Tian, Y., ... & Tang, X. (2015). Deepid-net: Deformable deep convolutional neural networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2403-2412).

[73] Dai, J., Qi, H., Xiong, Y., Li, Y., Zhang, G., Hu, H., & Wei, Y. (2017). Deformable convolutional networks. In *Proceedings of the IEEE international conference on computer vision* (pp. 764-773).

[74] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., & Berg, A. C. (2016, October). Ssd: Single shot multibox detector. In *European conference on computer vision* (pp. 21-37). Springer, Cham

[75] Wang, L., Ouyang, W., Wang, X., & Lu, H. (2015). Visual tracking with fully convolutional networks. In *Proceedings of the IEEE international conference on computer vision* (pp. 3119-3127).

[76] Le, T. N., & Sugimoto, A. (2017, September). Deeply Supervised 3D Recurrent FCN for Salient Object Detection in Videos. In *BMVC* (Vol. 1, p. 3).

[77] Donahue, J., Jia, Y., Vinyals, O., Hoffman, J., Zhang, N., Tzeng, E., & Darrell, T. (2014, January). Decaf: A deep convolutional activation feature for generic visual recognition. In *International conference on machine learning* (pp. 647-655). PMLR.

[78] Azizpour, H., Sharif Razavian, A., Sullivan, J., Maki, A., & Carlsson, S. (2015). From generic to specific deep representations for visual recognition. In *Proceedings of the IEEE conference on computer vision and pattern*

recognition workshops (pp. 36-45).

[79] Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q. (2017). Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4700-4708).

[80] Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 580-587).

[81] Chen, L. C., Papandreou, G., Kokkinos, I., Murphy, K., & Yuille, A. L. (2017). Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence, 40*(4), 834-848.

[82] Pohlen, T., Hermans, A., Mathias, M., & Leibe, B. (2017). Full-resolution residual networks for semantic segmentation in street scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4151-4160).

[83] Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., ... & Adam, H. (2017). Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*.

[84] Denton, E. L., Zaremba, W., Bruna, J., LeCun, Y., & Fergus, R. (2014). Exploiting linear structure within convolutional networks for efficient evaluation. *Advances in neural information processing systems, 27*.

[85] Girshick, R. (2015). Fast R-CNN. Proceedings of the IEEE International Conference on Computer Vision (ICCV).

[86] Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems, 28*.

[87] Adriana, R., Nicolas, B., Ebrahimi, K. S., Antoine, C., Carlo, G., & Yoshua, B. (2015). Fitnets: Hints for thin deep nets. *Proc. ICLR, 2*.

[88] Han, S., Pool, J., Tran, J., & Dally, W. (2015). Learning both weights and connections for efficient neural network. *Advances in neural information*

processing systems, *28*.

[89] Luo, J. H., Wu, J., & Lin, W. (2017). Thinet: A filter level pruning method for deep neural network compression. In *Proceedings of the IEEE international conference on computer vision* (pp. 5058-5066).

[90] Iandola, F. N., Han, S., Moskewicz, M. W., Ashraf, K., Dally, W. J., & Keutzer, K. (2016). SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and< 0.5 MB model size. *arXiv preprint arXiv:1602.07360*.

[91] Yang, Z., Moczulski, M., Denil, M., De Freitas, N., Smola, A., Song, L., & Wang, Z. (2015). Deep fried convnets. In *Proceedings of the IEEE international conference on computer vision* (pp. 1476-1483).

[92] Kohli, M., Prevedello, L. M., Filice, R. W., & Geis, J. R. (2017). Implementing machine learning in radiology practice and research. *American journal of roentgenology*, *208*(4), 754-760.

[93] Dormehl, L. (2019). What is an artificial neural network? Here's everything you need to know. *Digital Trends*. (accessed 13 February 2019).

[94] Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, *60*(2), 91-110.

[95] Meyer, P., Noblet, V., Mazzara, C., & Lallement, A. (2018). Survey on deep learning for radiotherapy. *Computers in biology and medicine*, *98*, 126-146.

[96] Kim, K. G. (2016). Book review: Deep learning. *Healthcare informatics research*, *22*(4), 351-354. doi:10.4258/hir.2016.22.4.351.

[97] Shen, D., Wu, G., & Suk, H. I. (2017). Deep learning in medical image analysis. *Annual review of biomedical engineering*, *19*, 221. https://doi.org/10.1146/annurev-bioeng-071516-044442.

[98] Waite, S., Scott, J., Gale, B., Fuchs, T., Kolla, S., & Reede, D. (2017). Interpretive error in radiology. *American Journal of Roentgenology*, *208*(4), 739-749. doi:10.2214/AJR.16.16963.

[99] Tajbakhsh, N., Shin, J. Y., Gurudu, S. R., Hurst, R. T., Kendall, C. B., Gotway, M. B., & Liang, J. (2016). Convolutional neural networks for medical image analysis: Full training or fine tuning?. *IEEE transactions on*

medical imaging, *35*(5), 1299-1312.

[100] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016). Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2818-2826). doi:10.1109/CVPR.2016.308.

[101] He, K., Gkioxari, G., Dollár, P., & Girshick, R. (2017). Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision* (pp. 2961-2969).

[102] Joshi, D., Mishra, V., Srivastav, H., & Goel, D. (2021). Progressive transfer learning approach for identifying the leaf type by optimizing network parameters. *Neural Processing Letters*, *53*(5), 3653-3676.

[103] Rajpurkar, P., Irvin, J., Bagul, A., Ding, D., Duan, T., Mehta, H., ... & Ng, A. Y. (2017). Mura: Large dataset for abnormality detection in musculoskeletal radiographs. *arXiv preprint arXiv:1712.06957*.

[104] Ronneberger, O., Fischer, P., & Brox, T. (2015, October). U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention* (pp. 234-241). Springer, Cham.

[105] Pan, S. J., & Yang, Q. (2009). A survey on transfer learning. IEEE Transactions on knowledge and data engineering. *22 (10)*, *1345*.

[106] Raghavendra, U., Bhat, N. S., Gudigar, A., & Acharya, U. R. (2018). Automated system for the detection of thoracolumbar fractures using a CNN architecture. *Future Generation Computer Systems*, *85*, 184-189.

[107] Tomita, N., Cheung, Y. Y., & Hassanpour, S. (2018). Deep neural networks for automatic detection of osteoporotic vertebral fractures on CT scans. *Computers in biology and medicine*, *98*, 8-15.

[108] Ebsim, R., Naqvi, J., & Cootes, T. F. (2018, September). Automatic detection of wrist fractures from posteroanterior and lateral radiographs: a deep learning-based approach. In *International Workshop on Computational Methods and Clinical Applications in Musculoskeletal Imaging* (pp. 114-125). Springer, Cham.

[109] England, J. R., Gross, J. S., White, E. A., Patel, D. B., England, J. T., & Cheng, P. M. (2018). Detection of traumatic pediatric elbow joint effusion using a deep convolutional neural network. *American Journal of Roentgenology, 211*(6), 1361-1368.

[110] Urakawa, T., Tanaka, Y., Goto, S., Matsuzawa, H., Watanabe, K., & Endo, N. (2019). Detecting intertrochanteric hip fractures with orthopedist-level accuracy using a deep convolutional neural network. *Skeletal radiology*, *48*(2), 239-244.

[111] Badgeley, M. A., Zech, J. R., Oakden-Rayner, L., Glicksberg, B. S., Liu, M., Gale, W., ... & Dudley, J. T. (2019). Deep learning predicts hip fracture using confounding patient and healthcare variables. *NPJ digital medicine*, *2*(1), 1-10.

[112] Adams, M., Chen, W., Holcdorf, D., McCusker, M. W., Howe, P. D., & Gaillard, F. (2019). Computer vs human: Deep learning versus perceptual training for the detection of neck of femur fractures. *Journal of medical imaging and radiation oncology*, *63*(1), 27-32.

[113] Pranata, Y. D., Wang, K. C., Wang, J. C., Idram, I., Lai, J. Y., Liu, J. W., & Hsieh, I. H. (2019). Deep learning and SURF for automated classification and detection of calcaneus fractures in CT images. *Computer methods and programs in biomedicine*, *171*, 27-37.

[114] Gan, K., Xu, D., Lin, Y., Shen, Y., Zhang, T., Hu, K., ... & Liu, Y. (2019). Artificial intelligence detection of distal radius fractures: a comparison between the convolutional neural network and professional assessments. *Acta orthopaedica*, *90*(4), 394-400.

[115] Guan, B., Yao, J., Zhang, G., & Wang, X. (2019). Thigh fracture detection using deep learning method based on new dilated convolutional feature pyramid network. *Pattern Recognition Letters*, *125*, 521-526.

[116] Lin, T. Y., Dollár, P., Girshick, R., He, K., Hariharan, B., & Belongie, S. (2017). Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2117-2125).

[117] Cai, Z., & Vasconcelos, N. (2018). Cascade r-cnn: Delving into high quality object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 6154-6162).

[118] D. Joshi, Wrist-Fracture-Images (2022), GitHub repository, https://github.com/djoshi712/Wrist-Fracture-Images.

[119] Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., ... & Zitnick, C. L. Microsoft coco: Common objects in context. InEuropean conference on computer vision 2014 Sep 6 (pp. 740-755).

[120] Everingham, M., Eslami, S. M., Van Gool, L., Williams, C. K., Winn, J., & Zisserman, A. (2015). The pascal visual object classes challenge: A retrospective. *International journal of computer vision*, *111*(1), 98-136.

[121] *How to Label Data for Machine Learning: Process and Tools | AltexSoft*. (2021, November 26). AltexSoft. https://www.altexsoft.com/blog/datascience/how-to-organize-data-labeling-for-machine-learning.

[122] Torralba, A., Russell, B. C., & Yuen, J. (2010). Labelme: Online image annotation and applications. *Proceedings of the IEEE*, *98*(8), 1467-1484.

[123] Kim, H. E., Cosa-Linan, A., Santhanam, N., Jannesari, M., Maros, M. E., & Ganslandt, T. (2022). Transfer learning for medical image classification: a literature review. *BMC medical imaging*, *22*(1), 1-13.

[124] Abdulla, W. (2017). Mask R-CNN for object detection and instance segmentation on Keras and TensorFlow.

[125] Wu, Y., Kirillov, A., Massa, F., Lo, W. Y., & Girshick, R. (2019). Detectron2.

[126] *Overview of hyperparameter tuning | Vertex AI | Google Cloud*. (n.d.). Google Cloud. Retrieved September 27, 2020, from https://cloud.google.com/vertex-ai/docs/training/hyperparameter-tuning-overview

[127] *Anyscale - What is hyperparameter tuning?* (n.d.). Anyscale. Retrieved September 2020, from https://www.anyscale.com/blog/what-is-hyperparameter-tuning

[128] Xie, S., Girshick, R., Dollár, P., Tu, Z., & He, K. (2017). Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1492-1500).

[129] Howard, J., & Ruder, S. (2018). Universal language model fine-tuning for text classification. *arXiv preprint arXiv:1801.06146*.

[130] Smith, L. N. (2017, March). Cyclical learning rates for training neural networks. In *2017 IEEE winter conference on applications of computer vision (WACV)* (pp. 464-472). IEEE.

[131] Smith, L. N. (2018). A disciplined approach to neural network hyper-parameters: Part 1--learning rate, batch size, momentum, and weight decay. *arXiv preprint arXiv:1803.09820*.

[132] Gao, H. (2017, August 31). *Object Localization in Overfeat. The task of object localization is to… | by Hao Gao | Towards Data Science*. Medium. https://towardsdatascience.com/object-localization-in-overfeat-5bb2f7328b62

APPENDIX

8.1 List of Publications

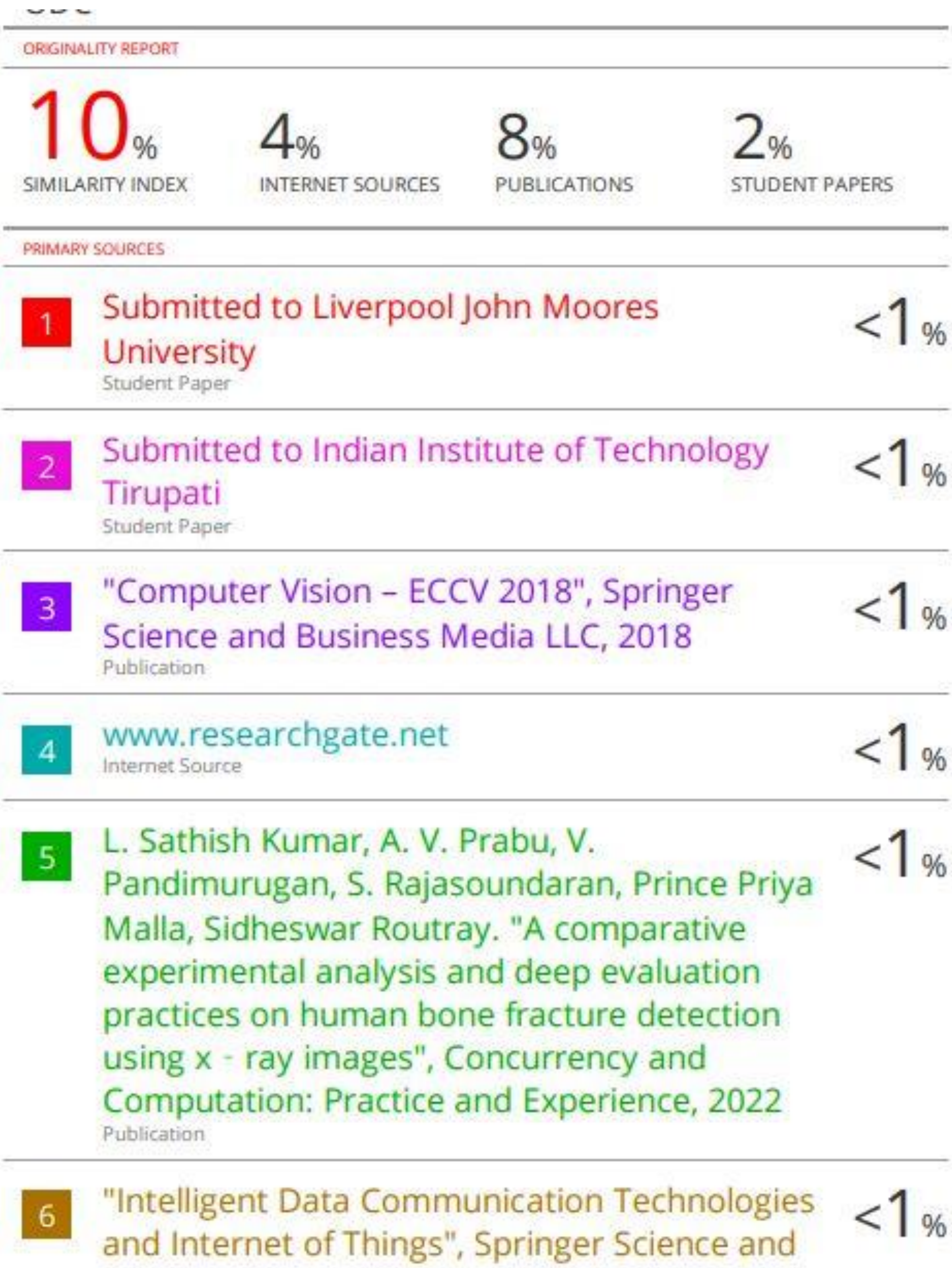
1. Deepa Joshi, Thipendra P Singh. "*A survey of fracture detection techniques in bone X-ray images*". Artificial Intelligence Review, vol. 53 no. 6, pp. 4475-4517, 2020. https://doi.org/10.1007/s10462-019-09799-0.

2. Deepa Joshi, Thipendra P Singh, Anil Kumar Joshi. "*Deep Learning-Based Localization and Segmentation of Wrist Fractures on X-ray Radiographs*". Neural Computing and Applications, 2022. https://doi.org/10.1007/s00521-022-07510-z.

3. Deepa Joshi, Thipendra P Singh, Gargeya Sharma. "*Automatic surface crack detection using segmentation-based deep-learning approach*". Engineering Fracture Mechanics, 2022. https://doi.org/10.1016/j.engfracmech.2022.108467.

4. Deepa Joshi, Thipendra P Singh. "*Deep learning based wrist fracture detection and Segmentation*". International Conference on Computational Intelligence and Smart Communication, 2022 (Presented and accepted).

5. Deepa Joshi, Thipendra P Singh. "*Novel use of a deep convolution architecture pre-trained on surface crack dataset to localize and segment wrist bone fractures*". IEEE International Conference (SMART-2022) (Presented and accepted).



ORIGINALITY REPORT

10%	4%	8%	2%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Liverpool John Moores University Student Paper	<1%
2	Submitted to Indian Institute of Technology Tirupati Student Paper	<1%
3	"Computer Vision – ECCV 2018", Springer Science and Business Media LLC, 2018 Publication	<1%
4	www.researchgate.net Internet Source	<1%
5	L. Sathish Kumar, A. V. Prabu, V. Pandimurugan, S. Rajasoundaran, Prince Priya Malla, Sidheswar Routray. "A comparative experimental analysis and deep evaluation practices on human bone fracture detection using x‐ray images", Concurrency and Computation: Practice and Experience, 2022 Publication	<1%
6	"Intelligent Data Communication Technologies and Internet of Things", Springer Science and	<1%