University of Kerbala
College of Computer Science & Information Technology
Computer Science Department

Bone Age Assessment Using a New Deep Neural Network Architecture

A Thesis

Submitted to the Council of the College of Computer Science & Information Technology / University of Kerbala in Partial Fulfillment of the Requirements for the Master Degree in Computer Science

Written by :

Alaa Jamal Jabbar Kadhim

Supervised by:

Assist. Prof. Dr. Ashwan Anwer Abdulmunem

2023 A.D. 1444 A.H.

بسم الله الرحمن الرحيم

﴿ يَرْفَعِ اللَّهُ الَّذِينَ آمَنُوا مِنْكُمْ وَالَّذِينَ أُوتُوا الْعِلْمَ دَرَجَاتٍ
وَاللَّهُ بِمَا تَعْمَلُونَ خَبِير ﴾

صدق الله العلي العظيم

سورة المجادلة (11)

Supervisor Certification

I certify that the thesis entitled (Bone **Age Assessment Using a New Deep Neural Network Architecture**) was prepared under my supervision at the department of Computer Science/College of Computer Science & Information Technology/ University of Kerala as partial fulfillment of the requirements of the degree of Master in Computer Science.

Signature:

Supervisor Name: **Assist. Prof. Dr. Ashwan Anwer Abdulmunem**

Date: / /2023

The Head of the Department Certification

In view of the available recommendations, I forward the thesis entitled "Bone Age Assessment using Anew Deep Neural Network Architecture" for debate by the examination committee.

Signature

Assist. Prof. Dr. Muhannad Kamil Abdulhameed

Head of Computer Science Department

Date: / /2023

Certification of the Examination Committee

We hereby certify that we have studied the dissertation entitled (**Bone Age Assessment Using a New Deep Neural Network Architecture)** presented by the student (**Alaa Jamal Jabbar Kadhim**) and examined her in its content and what is related to it, and that, in our opinion, it is adequate with (**Very Good**) standing as a thesis for the degree of Master in Computer Science.

Signature:
Name: Suhad Ahmed Ali
Title: Prof.Dr
Date: / / 2023
(**Chairman**)

Signature:
Name: Noor Dhia Al-Shakarchy
Title: Assist.Prof.Dr.
Date: / / 2023
(**Member**)

Signature:
Name: Elham Muhammed Thabit

Title: Assist.Prof.Dr.
Date: / / 2023
(**Member**)

Signature:
Name: Ashwan Anwer Abdulmunem

Title: Assist.Prof.Dr.
Date: / / 2023
(**Member and Supervisor**)

Approved by the Dean of the College of Computer Science & Information Technology, University of Kerbala.

Signature :
Title: Assist.Prof.Dr. Name: Ahmed Abdulhadi Ahmed
Date: / / 2023
(**Dean of Collage of Computer Science & Information Technology**)

Dedication

First, I dedicate my effort to holy Karbala city, Imam Al-Hussein, peace be upon him, and then to **My mother**, **My father**, **My husband,** and **My children** who stood with me and supported me in my study journey and all My friends who always supported me and finally to the rest of My family, especially my sister Zaman Jamal.

Alaa Jamal

2023 A.D

Acknowledgement

First and foremost, praises and thanks to Allah, the Almighty, for his showers of blessings throughout my research work to complete the research successfully.

I would like to express my deep and sincere gratitude to my research supervisor, **Assistant Professor Dr. Ashwan Anwer Abdulmunem**, for her efforts, scientific instructions and good advice, and her keenness to master every small and large, which had a great impact on enriching the message. And show it at this scientific level. Working and studying under her supervision was a great privilege and honor. I am very grateful for what you have given me, may Allah reward her.

I extend my sincere thanks and gratitude to the Dean of the College of Computer Science and Information Technology, and all members of the teaching staff, and also the head of the computer department, and all graduate studies staff and my colleagues are graduate students in the College of Computer Science and Information Technology / the University of Karbala.

Alaa Jamal Jabbar

2023 A.D

Abstract

Rapid advances in technology and artificial intelligence have driven automated systems in a variety of fields, including medicine. One application is the automated assessment of bone age that used in Forensic medicine. Bone age assessment used to identify endocrine disorders and many bone diseases or bone age based on x-ray image for left hand. A children's bone age can be determined by tracking changes in left hand bone age patterns, at this point a large gap may appear between bone age and chronological age and this indicates a problem in health of the child.

In this work a model was built to assess the bone age group of people based on deep learning techniques, and to achieve this aim, an architecture commensurate with the components of the computer device was built. The proposed system included pre-processing the data, extracting feature, and then classifying the children bone age into five age groups as outputs of the model.

The left hand image of children from one to the age of 18 years old, based on x-rays images (RSNA) containing 12611 image, which are used as a dataset in this work divided into 5 classes. The accuracy that obtained for each class was (97%, 95%, 96%, 98%, 95 %) respectively, while the total accuracy was 97.22% for testing, validation, and 98% for training.

Declaration Associated with this Thesis

1- A. J. Jabbar and A. A. Abdulmunem, "Bone age assessment using deep learning architecture: A Survey," *2022 International Conference on Intelligent Systems and Computer Vision (ISCV)*, 2022, pp. 1-6, doi: 10.1109/ISCV54655.2022.9806110.

2- A. J. Jabbar, and Ashwan A. Abdulmunem. "Bone age assessment based on deep learning architecture." *International Journal of Electrical and Computer Engineering* 13.2 (2023): 2078.

Table of Contents

List of Tables

Tabel Title **Page**

List of Figures

Figure Title **Page**

List of Algorithms

Title *Page*

List of Abbreviations

Abbreviation	Description
ID	One-Dimensional
AI	Artificial Intelligent
ANN	Artificial Neural Network
Adam	Adaptive Moment Estimation
AAM	Active Appearance Model
AdaGrad	Adaptive Gradient Algorithm
AdaDelta	Adaptive Learning Rate
BAA	Bone Age Assessment
BN	Batch Normalization
CNN	Convolutional Neural Network
CSV	Comma Separated Values
CT	Computerized Tomography
CV	Computer Vision
DHA	Digital Hand Atlas
DL	Deep Learning
DNN	Deep Neural Network
DRU	Distal Radius and Ulna
FC	Fully Connected
FP	False Positive
FN	False Negative

GP	Greulich and Pyle
HOG	Histogram of Oriented Gradients
LBP	Local Binary Pattern
MBGD	Mini-Batch Gradient Descent
NTS-Net	Navigator Agent. Teacher Agent. Scrutinizer Agent
RSNA	The Radiological Society of North America
RMSProp	Root Mean Square Propagation
ROI	Region of Interest
ReLU	Rectified Linear Unit
SGD	Stochastic Gradient Descent
SIFT	Scale-Invariant Feature Transform
SVM	Support Vector Machine
TW2	Tanner Whitehouse 2
TP	True Positive
TN	True Negative
TS	Turner Syndrome

CHAPTER ONE

INTRODUCTION

1.1 Overview

The human age can be estimated using different types of radiograph images such as X-ray, MRI, CT and DICOM [1]. Age estimation for live human is critical in many fields and due several reasons. In addition, accurate age estimation can play a great role in the medical diagnosis process[2]. Human hand bones have many significant properties that can be employed effectively in human age assessment. Many methods have been proposed for age estimation based on hand x-ray images. However, these methods involve intensive image processing in order to prepare the images for the consequent stages[3]. In addition, in order to compute the hand lengths, these methods usually perform a segmentation process which is a very challenging process[4]. Moreover, the success of the future extraction step highly depends on the success of the segmentation step. The deep learning can be used effectively to perform the feature extraction step without the need for the segmentation process by taking the hand image as input and returning the hand features that can be used in the age estimation process as output[4]. Early research efforts have been performed in order to achieve accurate bone age assessment[5]. Several manual methods that depended on left hand bones have been proposed including Greulich and Pyle Method (GP) and Tanner Whitehouse Method (TW). In both circumstances, the bone examination technique takes a long time [6]. An x-ray of the left hand, including the fingers and wrist, took first followed by image analysis. The x-ray image's bones are compared to radiographs in a defined atlas of bone development. These are based on a large number of rays taken from children of the same gender and age[7], [8]. The Active

Appearance Model is used by BoneXpert (AAM). The algorithm calculates the total bone age based on the form, texture, and intensity of the bones using the GP or TW approaches. However, it is susceptible to picture quality and does not use the carpal bones, despite their relevance in assessing skeletal development in newborns and toddlers[9].

Recently, deep learning-based approaches demonstrated performance improvements over traditional machine learning methods for many problems in biomedicine[10]. Convolutional neural networks (CNN) have been utilized successfully in medical imaging [11].

The processing of one image using a deep learning technique often takes few time[12], and the accuracy of these approaches often exceeds that of conventional methods. Deep neural network-based techniques for the assessment of bone age from hand radiographs have already been proposed [13]. However, this work did not conduct a numerical evaluation of the performance of their models utilizing different hand bones[12]–[14]. Furthermore, discover that better preprocessing and training networks on radiographs from scratch, rather than fine-tuning from natural picture domain, can increase the performance of deep learning models for bone age estimation[12]–[14].

1.2 Problem Statement

Left hand bone age assessment is a common clinical practice to analyze and assess the biological maturity of pediatric patients, The current manual process is very time-consuming and has high probabilities of misjudgment in classifying the left hand bone age. However, recent developments in the field of neural networks provide an opportunity to automate this process.

1.3 Related Work

In The past years, there are plentiful published works on the development of automated bone age assessment systems which can be broadly categorized into common machine learning and deep learning approaches. Generally, the TW approach has been more widely adopted compared to the GP approach due to its modular structure that allows features to be extracted from selected regions of interest. Hence, the conventional machine learning approach explores various handcrafted features from the specific ROIs to train the classifier or regressor to assess the bone age [9]. These approaches, however, were time demanding, and the output results varied due to the user's subjectivity [15]. As a result, number of researchers aimed to develop fully automated approaches for assessing bone age. Related work classifies into two approaches: Machine learning and Deep learning [15].

Machine learning (ML) applications in healthcare have seen significant advancements. Taking advantage of these unprecedented advancements in AI, the medical community has developed AI applications that make the most of medical pictures and various aspects of clinical practice offer assistance for clinical choices [16]. Medical experts put in a lot of time and effort to keep up with the tremendous expansion of medical image works that is subjective, prone to human error, and can vary from one expert to the next. Therefore, it has significant interest for researchers [11].

Most of them used (ML) for bone age assessment in this section such as Dehghani, at el. [16] who used computer vision methods such as histogram of oriented gradients (HOG), local binary pattern (LBP), and scale invariant feature transform (SIFT) to automatically estimate the bone age of children aged newborn to 18 years old. In this case, 442 left-hand radiographs from the University of Southern California (USC) hand atlas used for the first time, HOG–LBP–dense SIFT features with background removal used to estimate the subject group's bone age in this investigation. Features from the carpal and epiphyseal areas of interest are retrieved for this purpose (ROIs). For classification, SVM and 5-fold cross-validation utilized. Female radiographs had an accuracy rate of 73.88 percent, whereas male radiographs had an accuracy rate of 68.63 percent. The average absolute error is 0.5 for both sex x-ray.

Tajmir, et al. [17] assess bone age based on AI. Bone radiographs of 280 ages for children and young people aged 5 to 18 years old they used to test the suggested technique. According to the findings collected, the suggested system had an overall accuracy rate of 68%.

Deep learning in assess bone age was a very important topic worldwide. The researches in this area were persistent. Due to "Deep Neural Networks" (DNN) having been utilized for various objectives and producing very excellent results, it's frequently compared to DL approaches. DNN has been used for multiple purposes and generally produces promising results. The researchers worked done in bone age process and classification using deep learning techniques is explained in this section. Most of them using pertained method. For example, Larson at el. [12] have introduced a bone age assessment system based on the deep learning techniques. The classifier Resnet-50 produced probability distribution for skeletal age spanning from 1 to 18 years in one-month increments, which was used to assess bone age, and achieved an accuracy of 63%.

Hyunkwang, at el. [14] proposed fully automated deep learning pipeline that can conduct BAA, segment an area of interest, standardize and preprocess input radiographs. On these held-out test photos, the models, which are powered by an ImageNet pre-trained and fine-tuned convolutional neural network (CNN), achieved accuracies of 57.32 and 61.40 percent, respectively, for the female and male cohorts.

N. Mualla at el. [18] developed an automated bone age assessment system using the transfer learning approach. Using deep neural networks that have been trained and fine-tuned, the proposed method preprocesses x-ray images of hands and extracts distinguishing properties (AlexNet and ResNet-101). A variety of classification methods are used to determine the age group of the hand X-ray image, including the decision tree and the k-nearest neighbor algorithm. The proposed approach used1684 x-ray image from the RSNA bone age dataset. Features based on ResNet-101 surpass those based on AlexNet, according to the findings. Furthermore, the decision tree classifier surpasses the competition with a classification accuracy of up to 100%.

Bone age estimated by using the Tanner-Whitehouse (TW3) technique and deep convolution networks T. D. Bui, at el. [19] based on extracted regions of interest (ROI)-detection and classification using Faster-RCNN and Inception-v4 networks, respectively, in this work. To improve the accuracy of bone age assessment, the proposed technique utilizes expert knowledge from TW3 and engineering from deep convolution networks. The results of the experiments indicated that the suggested approach had the greatest performance among current state-of-the-art methods, with a mean absolute error of roughly 0.59. DHA dataset is used to assess the model's performance for maturity stage classification, and the accuracy is computed by averaging the five-fold cross-validation results for each maturity stage. Inception-v4 and Resnet-101 evaluated for their performance used the ImageNet pre-trained model. The suggested technique had an average accuracy of 81.5 percent with Inception-v4 and 78 percent with Resnet-101. Consequently, settle on Inception-v4 for maturity stage classification.

Pengyi Hao, at el. [20] discrete bone age labels (for example, 0-228 month) are thought of as a sequence that is used to generate a series of subsequences. The network then uses the overlapping information between adjacent subsequences to output multiple bone age ranges for one case at the same time. The overlapping portion of these age ranges is used to calculate the final predicted bone age. When compared to state-of-the-art methods on a public dataset, the work with no preprocessing achieves a much smaller mean absolute error. ResNet-101and Inception -v3 methods used in this work, the suggested technique obtains 76 percent and 80 percent, respectively.

Rani, et al. [21] automated methods have been developed to remove non-uniform illumination-induced noise from radiograph acquisitions. Following this, a computational method for extracting the wrist area from an image utilizing operations on specified bit planes have been presented.

Extracting wrist areas and classifying them according to gender requires the use of a deep convolutional neural network

architecture called the GeNet. A total of 12442 pictures from the Radiological Society of North America (RSNA) databases were used in the experiments. Extracted wrist areas yielded an accuracy of 82.18 percent for GeNet's performance.

Other researchers used (CNN) as Chen, Matthew C.[22] researchers investigated the application of convolutional neural network approaches to train a model to predict a patient's developing bone age based on x-ray pictures. They used the digital hand atlas dataset (DHA), which was made up of scans that have been annotated independently by two radiologists. Previously used solutions for this purpose often entailed a pipeline of segmentation and handcrafted feature extraction. Given recent breakthroughs in the usefulness of convolutional neural networks for image categorization, researchers want to abandon this technique. On the validation set, it reached to the top these accuracies of (46 % and 70 %), respectively, with (RMS) error of 1.1, using (CNN) technique, found that supplementing the dataset with random distortions resulted in the greatest increase in accuracy. This appears to imply that the performance is mostly reliant on the amount of training instances and that it would likely improve more with additional data.

Lee, at el. [23] proposed a completely automated deep learning pipeline for segmenting a region of interest, standardizing and preprocessing input radiographs, and performing BAA. These models employed an ImageNet pre-trained, fine-tuned convolutional neural network (CNN) to obtain 57.32 and 61.40 % accuracies on held-out test pictures for the female and male cohorts, respectively. B Sowmya, at el. [24] A faster R-CNN takes the left-hand radiograph as input and returns the detected DRU region from the left-hand radiograph. This output is fed into a properly trained CNN model. The experiment section contains information about the experiments performed on 1101 radiographs of the left hand and wrist datasets, as well as the accuracy of the model when different optimization algorithms and training sample amounts were used. Finally, after parameter optimization, this system achieves 92% (radius) and 90% (ulna) classification accuracy.

S. Mutasa, at el. [25] designed a neural network method that is specifically calibrated to the assessment of bone age using a rather big institutional database. This work aims to demonstrate that advanced architectures may be effectively trained from scratch in the medical imaging domain and provide results that surpass any current suggested technique 8909 photos from researcher hospitals Image archiving and communication systems and 1383 photographs from the public digital hand atlas database were used to create the training data. All children aged 8 or older and under 10 years old were divided into four cohorts, with one each for boys and girls. 1-year cohort radiographs were used, with half of the subjects being male and half female, ranging in age from 0 to 1 year to 14–15 years. Left-hand radiographs were taken for bone age assessment, and no noteworthy results were found in the trauma examination. Only 0.654 and 0.561 were accurate for validation and test sets for young females respectively. There were 0.66 and 0.49 test and validation accuracies for older females. validation and test accuracy were 0.64 and 0.58, respectively, for young male subjects. The accuracy of the test sets for older males was 0.58. The sets of training were tow: one for females and the other for males.

Wu, at el. [26] the designers presented an ensemble-based deep learning pipeline to automatically assess the maturity of the distal radius and ulna (DRU) using left hand radiographs. The experimental dataset contains 1189 left-hand X-ray scans of children and teenagers. This method achieves 85.27% and 91.68% for radius and ulna classification respectively left-hand x-ray scans of children and teens in the experimental dataset.

K. Li, et al. [27] used the hand bone imaging dataset from the Radiological Society of North America (RSNA) to develop an advanced convolutional neural network (CNN) model based on fine-grained image categorization. For bone age classification, the extracted local features are integrated with global features from the whole picture in this model, which can detect relevant areas and extract local features automatically in the process of hand bone image identification. Bone age may be accurately and quickly determined using this approach, which does not need any image annotation information (besides bone age tags). Experimental findings reveal that using the RSNA dataset, the proposed technique accurately recognizes males and females in proportions of 66.38% and 68.53%, respectively. The summary of related work is illustrated in table 1.1.

Table 1.1: Summary of related Work.

Author	Year	Dataset	Proposed model	Accuracy
Chen, Matthew C.[15]	2016	DHA	VGGNet GoogleNet	46% 70%

Hyunkwang, at el. [14]	2017	RSNA	GoogleNet	Female=57.32
				Male=61.40 %

Lee, at el. [16]	2017	RSNA	CNN	Male=57.32 Female 61.40 %
Larson at el.[12]	2018	RSNA	Resnet-101	63%
S. Mutasa, at el.[17]	2018	8909 photos from the researcher hospital's and 1383 public digital atlas	DCNN	Older 10 year Male 58.85= Female =49% Under 10 year Male 58.1%= female=56%

B Sowmya, at el.[18]	2019	RSNA	F-RCNN	Ulna=92%
				Radius=90%

T. D. Bui, J. J. Lee, and J. Shin [19]	2019	DHA	Inception-v4 Resnet-101	81.5% 78%
Tajmir, et al. [20]	2019	Left hand x-ray from 5 to 18 year	Ml	68%

Wu, at el. [21]	2019	1189 left-hand X-ray	ensemble-based deep learning	Radius = 85.27%
				Ulna =91.68%

N. Mualla at el. [18]	2020	1684 left hand x-ray from RSNA	AlexNet and ResNet-101for extract feature and DT,SVM clasiffier	100%
Dehghani, at el. [22]	2020	442left-hand radiographs	SVM and 5-fold cross-validation	73.88%female 68.63%male
Rani, et al.[23]	2020	Left-Hand x-ray radiographs From 1-17 year	GeNet	82.18%
Pengyi Hao, at el. [24]	2021	RSNA	ResNet-101 Inception -v3	76%. 80%
K. Li, et al. [25]	2021	RSNA	NTS-NET network	66% male 68% female

1.4 The Aim of the Thesis

Used deep neural network architecture for bone age assessment. The proposed system aims to classify the age group of children using their left-hand x-ray images.

1.5 Contribution

In this thesis a model with a new Neural Network Architecture was proposed to classify left hand x-ray images for children from one to eighteen years old.

1.6 Thesis Organization

After chapter which presents an introduction to the entire thesis, and presents literature review. Chapter two contains the theoretical foundations of the working model as an explanation of the structure of the convolutional neural network models employed in this thesis, as well as the method of the convolving filters around the picture pixel. Stacked layers and pooling techniques are explained and the metrics are used to evaluate system. The proposed method for automated classification of x-ray images for detection of bone age maturity infection is detailed in chapter three**.** The experimental result of x-ray image classification models for bone age assessment diagnosis are reported in chapter four. The thesis conclusion and future works will be discussed in chapter five.

1.7 Summary

This chapter contains an overview on bone age assessment, problem statement, related work, the aim of this thesis, and organization of this thesis.

CHAPTER TWO

THEORETICAL BACKGROUN

2.1 Overview

This chapter describes the bone age assessment system. It also explains the concept of deep learning technique as represented by the convolutional neural network model and the convolution mechanism, as well as network training issues. Finally, will be presented the design model approach then followed by software testing and metrics.

2.2 Bon Age Assessment

Left hand bone age evaluation is a widespread clinical procedure in children for determining age group later than chronological age. One of the most widely used methods for determining hormone growth in children is bone age estimation. It enables them to diagnose children and comprehend the various disorders that accelerate or slow physical growth. The bone age study is used to estimate children's final height, determine puberty age, and predict the children's age group. Bone age analysis also assists doctors in treating children who exhibit the following symptoms: [Growth hormone shortage, hypothyroidism, premature puberty, and adrenal gland abnormalities are all examples of diseases that alter growth hormone levels. Turner syndrome and other growth problems are examples of genetic growth disorders (TS). Orthopedic or orthodontic problems in which the timing and type of treatment (surgery, bracing etc.) must be guided by the child's predicted growth[26]. The child's age gap does not always mean that they are ill. Even though medical science has made enormous advances and discoveries, the conventional method of determining bone age has not improved much. The Greulich-Pyle (GP) and Tanner Whitehouse 2 (TW2) methods are the most often used methods for determining bone age that represent subjective methods[27]. The two methods are illustrating below:

- GP technique is a well-established method for estimating bone age. First, an x-ray of the child's left hand is done, and the results are then compared to a set of standard images. Due to the number of bones in the hands, radiographs of the hands are simpler to take than those of other body parts. However, the inter and intra-observer variability in the bone age evaluation is significant, making it difficult to correctly estimate the bone age[28].
- TW2 is a rating system for estimating bone age. Children’s radiography from the United Kingdom was used in the development of this approach (1950 – 1960). Similarly, to the GP approach this technique likewise relies on radiographs of the left hand and wrist to gather data. A total of almost 20 bones make up the left hand, 13 of which are known as long or short bones, while the other seven are known as carpal bones. Bones are classified into one of nine groups, a through it, based on them. In order to get the overall score for the hand, replace each bone group with a given number and subtract that number from the previous total score. they can estimate a child's bone age based on their overall score[29].

2.3 Normalization

Normalization is one of the most essential preprocessing procedures employed. It involves normalizing the dimensions of the sample data (which belongs to both the train, validation, and test datasets) by splitting each dimension by its standard deviation. It will be done mathematically as follows[30]:

$$x^{"} = \frac{x'}{\sqrt{\frac{\sum_{i=1}^{N}(x_i - x^*)^2}{N-1}}} \tag{2.1}$$

$$x' = x - x^* \tag{2. 2}$$

$$\text{And } x^* = \frac{1}{\sum_{i=1}^{N} x_i} \tag{2. 3}$$

N is the size of training dataset, $x'$is the mean. $x^*$is standerd divation.

2.4 Data Augmentation

Applying data augmentation technology during network training is to increase the dataset size by modifying the form of the original data and getting additional invisible data for use in training. This is accomplished by employing a generator that increases the size of each picture each time it is transmitted over the network. It employs a variety of methods, the most frequent of which being translation, scale, flip, rotation, zoom, and Gaussian noise[31]. The aim of utilizing this strategy is to decrease the possibility of overfitting, increase the performance of the created model, and occasionally there is only a small training dataset available for most real-life crucial circumstances (e.g. medical datasets) This is what sets more deep than machine learning algorithms apart from others. Figure 2.1 depicts the data augmentation [32].

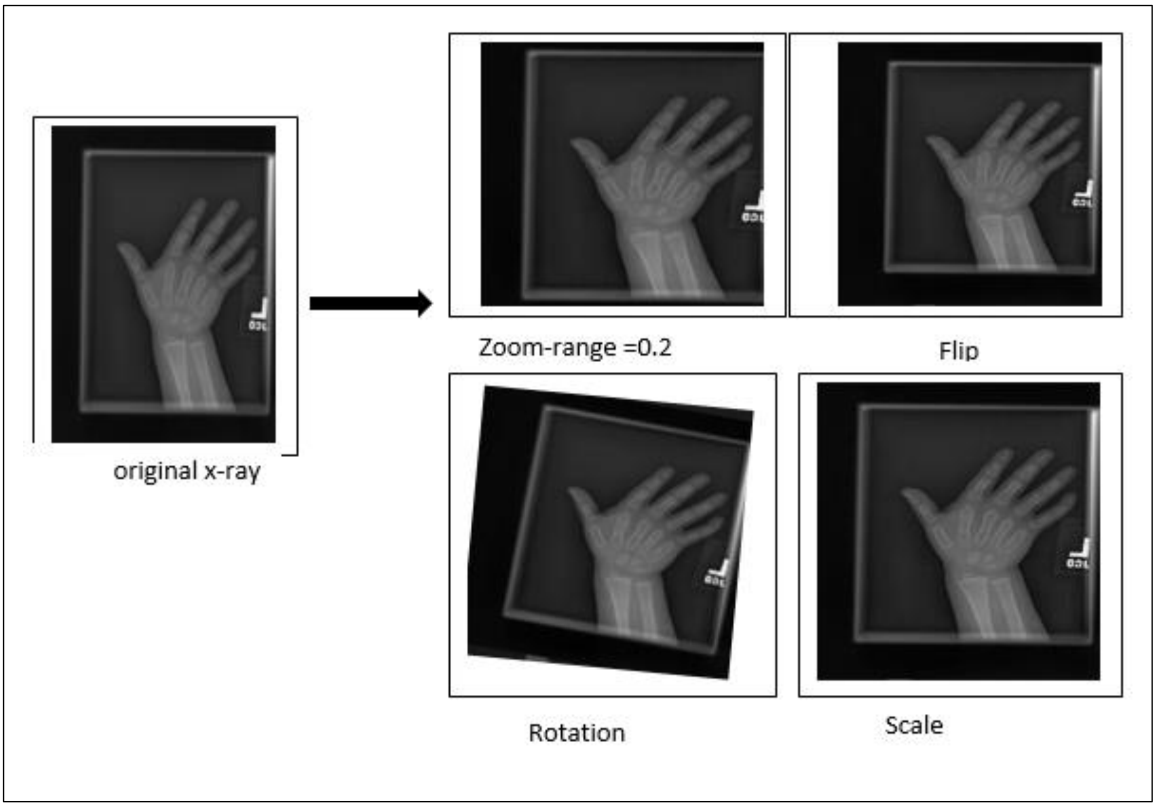


Figure 2.1: Data Augmentation.

2.5 Deep Learning

Deep learning is a subset of machine learning that is concerned with approaches for modelling neurons in the human brain[33]. Deep learning algorithms perform two functions: mental technique simulation and data mining development principles[34]. The machine learning branch is concerned with pattern recognition and data learning in particular[35]. The machine learning branch is primarily concerned with analytical thinking and data learning[38].

Deep learning can automatically extract features and deliver correct results[37]. However, machine learning requires a broader variety of data to be pre-processed since features must be retrieved manually. Deep learning extracts a wide variety of dimensions' characteristics, whether or not they are visible to the naked eye. Deep learning may emulate expert doctors in clinical decision-making by detecting and uncovering characteristics. Deep learning architectures are used in medical x-ray detection as well as other domains such as computer vision and image processing[39]. Deep learning has been developed in the medical industry to better comprehend the best results, the probability of illness aggravation, and creating a valid medical image in the shortest amount of time in disease diagnosis systems[40]**.**

2.5.1 Convolution Neural Network Concept

The phrase "convolutional neural networks" refers to the use of the convolution mathematical process to networks. CNNs are neural networks that appear to have a grid-like structure when processing input. It comprises, for example, digital picture data, which represents a two-dimensional network of pixels, and time series data, which represents a one-dimensional network sampled at regular intervals of time[32]. The CNN's design is affected by the arrangement of the visual cerebral cortex, and it is comparable to the structure of the communication pattern of neurons in the human brain[39]. In classic feedforward neural networks, each input neuron is linked to every output neuron in the layer below, known as the fully connected (FC) layer. Do not use FC layers in CNNs until the last layer(s) of the network[40].

A CNN is used to solve image processing difficulties in which the computer detects the item in the image. CNN modelling could be used for image processing, classification, recognition, and identification. The program transmits data to patterns that simulate the functioning of the human nervous system. CNNs are a sort of deep learning architecture that has been widely employed in a wide range of practical applications such as pattern recognition and image categorization[41]. It is a neural network that uses a specific "convolutional" layer instead of a fully connected layer for at least one layer in the network[42]. CNNs are essentially NNs that utilize convolution instead of generic array multiplication in at least one of their layers, and employ a linear technique known as convolution[43]. Convolution is a specific sort of linear operation. Convolutional networks are rudimentary neural networks with at least one layer that employs convolution rather than conventional array multiplication. Convolutional networks offer the benefit of processing inputs at many spatial scales. These types of input cannot be represented by traditional neural networks based on matrix multiplication[44].

CNN is a deep learning method that performs well in computer vision. Furthermore, CNN has the convolutional layer and pooling layer applied, and the activation function is employed as a ReLU "Rectified Linear Unit"[45]. It is collected by three types of layers: convolution, pooling, and fully connected. The first two layers conduct feature extraction by convolution and pooling, while the third, a fully connected layer, transfers the retrieved features to final outputs such as classification[46]. CNN designs use the explicit premise that network inputs are pictures, allowing us to add certain qualities within the architecture. Convolutional neural networks make use of three key concepts that might aid enhance machine learning systems: sparse interactions, parameter sharing, and equal representations[47], this will make the front-end function easier to build and will greatly minimize the number of parameters in the total network. Convolution also allows for the handling of variable-size inputs. Each output unit interacts with each input unit in standard neural network layers[48], see figure 2.2.

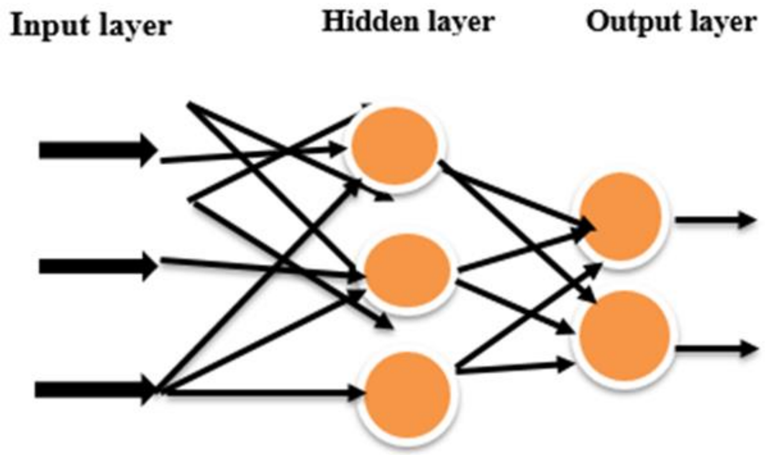
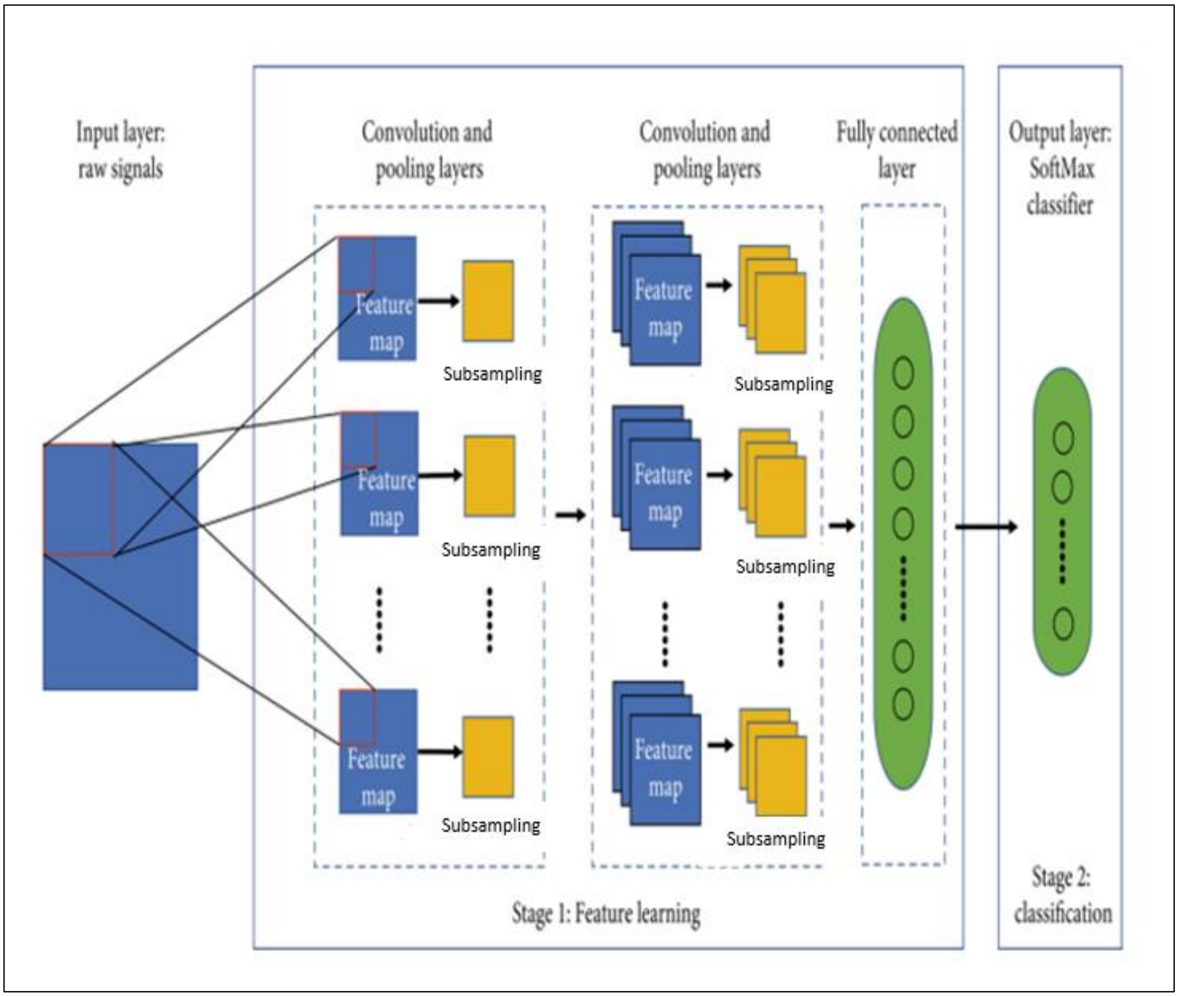


Figure 2.2: Example of Fully Connected Network in Traditional

A neural network transforms an input image or feature bus (one input node per input) through a number of hidden layers using nonlinear activation functions. Each hidden layer also comprises a set of neurons, each of which is completely linked to all of the neurons in the layer before it. The last layer of a neural network (the output layer) is likewise completely linked and shows the network's final output classifications[49]. Increasing the picture size, the number of inputs, weights, and hidden layers more than one layer, and altering the number of nodes in each layer, makes it difficult for the neural network to scale properly[50]. Figure 2.3 illustrates the overall structure of a CNN with forwarding propagation lately proven successful in a range of computer vision [41],[52].



2.3: General CNN with Forwarding Propagation[52].

The convolutional neural network(CNN) is built from a set of layers and an organization, which are as follows[52]:

2.5.1.1 Convolutional Layers.

The input is convolved in the first layer of the convolutional layer. Performing feature extraction is a necessary step in the CNN creation process. Then it's a mix of linear (the convolution method) and non-linear (the activation function) activities. The convolutional layer computes the convolutional operation of the input image by utilizing kernel filters (a tiny array of numbers) that are applied to the input, which is a tensor, an array of numbers, to extract essential characteristics. The kernel filters have the same dimensions as the input pictures but smaller constant parameters (tensors), allowing each filter to extract various sorts of characteristics [53], [54]**.** The feature map obtained as output values at each point of the tensor is computed using product-wise calculation between each kernel element and the input tensor element and added to generate the output values in the corresponding location in the output tensor. Convolution is the process reduces the size of an image while keeping the relationship between pixels by filtering it with a smaller pixel filter. It starts with the convolution window in the upper left corner of the input tensor, then scrolls over all locations both down and to the right. Several filters extract different features and create several of feature maps that representing multiple input tensor features. As a result, expressing the product as extracts is an additional functionality. The size and number of kernels are the two fundamental hyper parameters that determine the convolution procedure. The former is normally 3 x 3 as shown in figure 2.4 [50].

Sometimes there are important and necessary features in the external component of the image (the edges), which are represented in the input tensor that the center of the kernel cannot reach during the convolution operation, in addition to the difference in the size of the input tensor from the corresponding dimensions of width and height of the output represented by a map features. The padding process is used to treat this case by adding columns and rows of a specific number on either side of the input tensor, usually zero padding [55], [56].

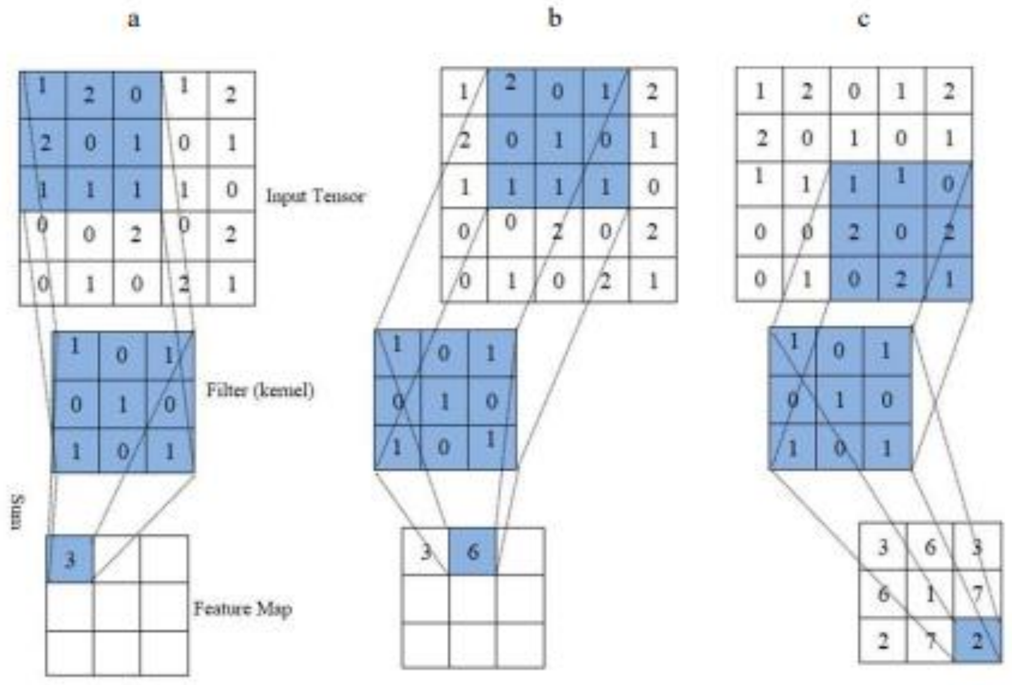


Figure 2.4, *Convolution Operation* [49].

The convolutional layer parameters computed according to equation (2.4)[57].

Convolutional= ((kernel size) x image Chanel+ bias) x no. of filters) (2.4)

The following is the formula for determining the output feature map size after convolution[57]:

$$h' = [\frac{h-f+p}{s} + 1] \quad (2.5)$$

$$w' = [\frac{w-f+p}{s} + 1] \quad (2.6)$$

Where h' denotes the height of the output feature map, w' denotes the width of the output feature map, h denotes the height of the input image, w denotes the width of the input image, f is the filter size, p denotes the padding of convolution operation and s denotes the stride of convolution operation.

2.5.1.2 Non-Linear Layer

The activation function is a property of an activated neuron that may be retained and mapped by a nonlinear function utilized to solve nonlinear problems[58]. An activation function's essential job in neural network models is to map inputs to outputs, with the input value obtained by computing the weighted sum of the neurons' inputs. The activation function is used by the neural network model to improve the model's expressiveness and to add importance to the neural network's artificial intelligence. Nonlinear activation layers are added after each learnable layer in the CNN design (layers with weights, i.e. convolutional layers and FC layers). The activation function is crucial to the design of a deep neural network[59]. The nonlinear activation function is applied to the output of the linear process represented by the convolution. Smooth nonlinear functions, which are mathematical representations of biological neuron activity, were previously used. Famous activation functions, as seen in figure 2.5, include the sigmoid, ReLU, tanh, and softmax[37].

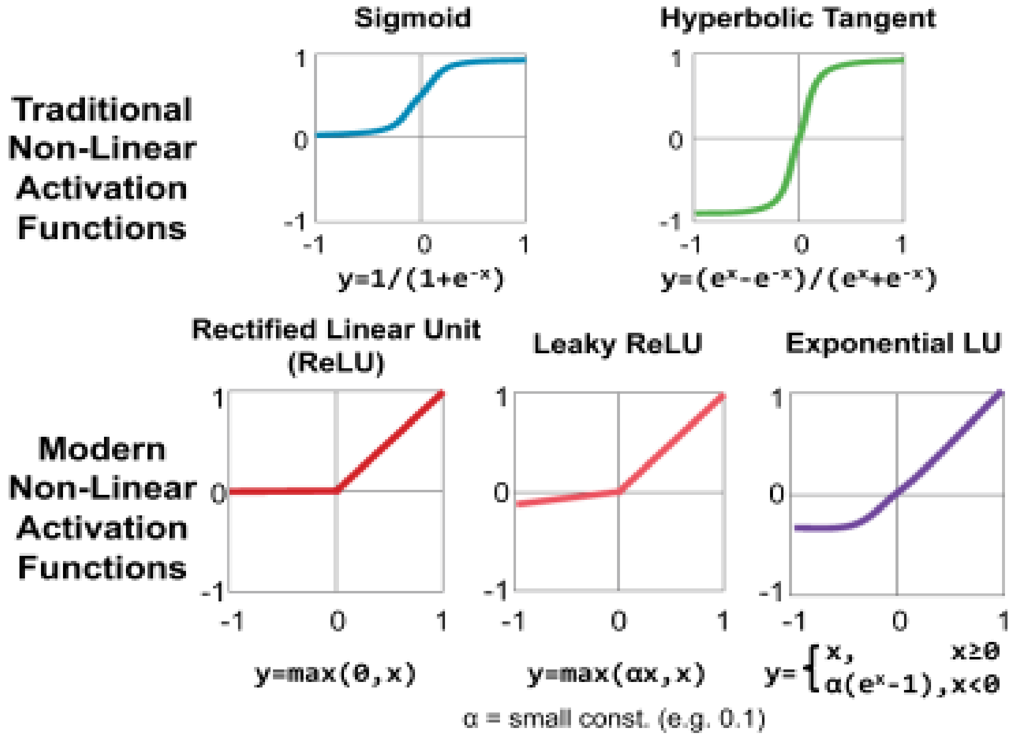


Figure 2.5: Forms of Activation Function[37].

The Rectified Linear Unit, or ReLU, is the most common form of activation function used in network that design today. It has the potential to increase the performance of CNNs. It does not stimulate all of the neurons at once, and it is linear in the positive dimension but zero in the negative dimension, implying that only a few neurons are affected at once, resulting in an efficient and scattered network. The twist in the function causes the non-linearity. Because of the gradient for half of the real line is zero, positive linearity effectively prevents gradient non-saturation (in contrast to sigmoid activations). However, there may be some significant drawbacks when using the ReLU activation function. Consider the scenario when a larger gradient is expected during the error back-propagation procedure; if this larger gradient is fed via a ReLU function, the weights may be altered in such a manner that the neuron is never triggered again. This circumstance is known as the Dying ReLU problem [55]. This function is defined by the equation (2.7) below [55]:

$$f(x)_{ReLU=\max(0,x)} \tag{2.7}$$

The SoftMax function, used for multiclass classification, is a more generalized variant of the sigmoid activation function. CNN utilize SoftMax after the final fully connection layer. The curve of this function is formed like a 'S'. The sigmoid is mathematically expressed as[37]:

$$s(x_i\ )=\frac{e^{x_i}}{\sum_{j=1}^{N} x_j} \tag{2.8}$$

where $s(xi\ )$ is the SoftMax function, and $e\ x$ is the standard exponential function of the input value x, and N is the number of output.

The Tanh activation function is used to limit the range of input values (real numbers) from -1 to 1. Tanh is mathematically expressed as[37]:

$$f(x)_{tanh=\frac{e^x - e^{-x}}{e^x + e^{-x}}} \tag{2.9}$$

Where $e\ x$ is the standard exponential function of the input value x, e^{-x} is the standard exponential function of the input value x.

2.5.1.3 Pooling Layer

The pooling layer is a necessary layer that conducts down sampling on the preceding layer's feature maps, resulting in new feature maps with a lower resolution[55]. This layer reduces the spatial dimension of the input greatly. The main advantage of this layer is to minimize the number of parameters or weights, thus lowering the computational cost. The second purpose is to protect the network from becoming overfit. There are two main methods for pooling (Average Pooling and Max-Pooling)[60].

$$h' = \left\lceil \frac{h-f}{s} \right\rceil \quad (2.10)$$

$$w' = \left\lceil \frac{w-f}{s} \right\rceil \quad (2.11)$$

Where h ′ signifies the output feature map's height, w′ denotes the output feature map's width, h denotes the height of the input feature map, w denotes the width of the input feature map, f is the pooling area size, and s is the pooling operation's stride.

Average pooling: To pool and extract features, use an average or mean idea. In a deep neural network, this is the first convolutional layer. The average pooling layer is used for down sampling. The input is divided into rectangular pooling portions, and the average values of each are calculated see figure 2.6 [58].

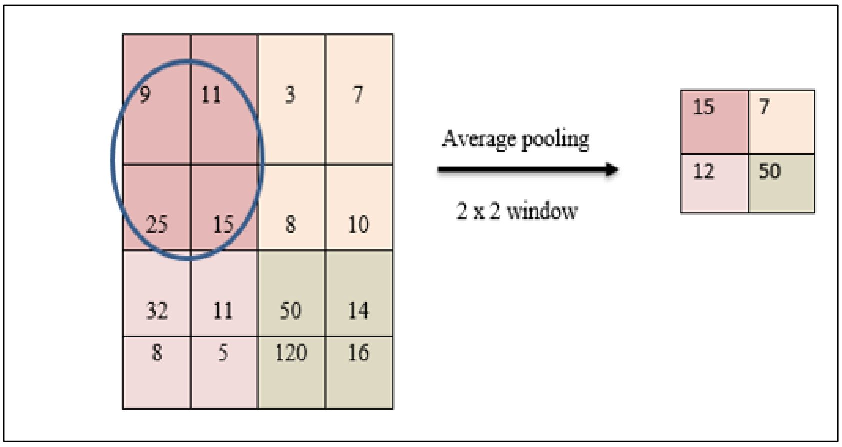


Figure 2.6: Example of Average Pooling Operation.

Max pooling: The most common pooling approach is max pooling. This approach works by providing feature maps in batches, selecting the greatest value for each patch, and disregarding the remainder. The max-pooling operation can be used to down sample the convolutional output bands. Max-pooling improves generalization efficiency by increasing convergence rates and reducing variability by selecting superior invariant features. Unlike the height and width dimensions, the depth dimension of feature maps does not vary. The max-pooling operator passes along the largest value inside a group of activations. The max-pooled band is made up of connected filters. The pooling layer decreases the dimensionality of the output from convolutional bands to pooled bands. Figure 2.7 depicts the average and max-pooling [59].

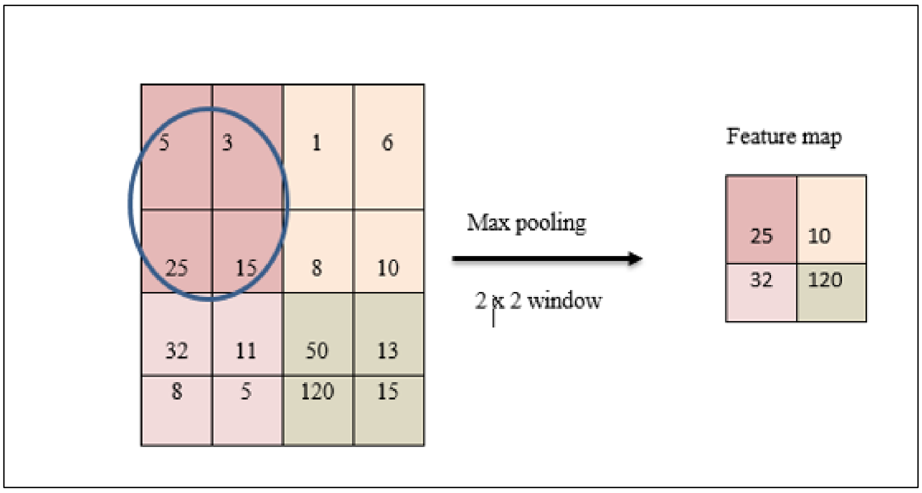


Figure 2.7: Example of Max Pooling Operation.

2.5.1.4 Fully Connected Layer

After multiple convolutions and pooling layers, the flatten layer added and computed according to equation (2.12).

Flatten_1= (image size) x filters (2.12)

The CNN generally ends up with several completely linked layers. Several neural network layers are added, with each input to each output coupled to a learnable weight, which is referred to as dense layers[61].

Using a subset of the fully connected layers, features retrieved by convolution layers and down sampled by pooling layers are transferred to the network's final outcomes, such as probabilities for each category in classification tasks. As illustrated in figure 2.8, the number of output nodes in the fully linked layer is generally equal to the number of classes. Various nonlinear functions, including Euclidean loss, sigmoid cross-entropy, and softmax cross-entropy, can be used to examine how network training handles variations between predicted and real labels[62].

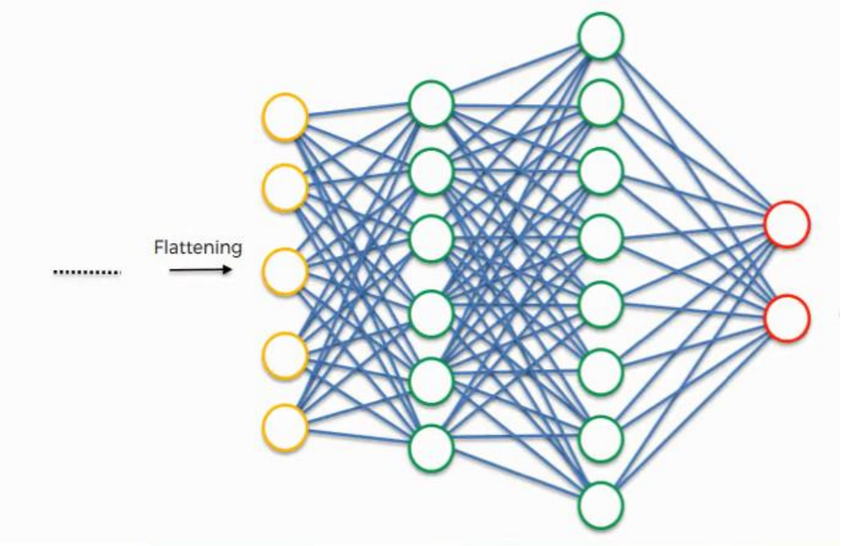


Figure 2. 8: The Architecture of Fully Connected Layers[62].

The activation function in this layer frequently differs from the other functions [63]. Each job necessitates the selection of a suitable activation function. In the multiclass classification problem, the softmax function is utilized, where each value squashes in ranges between 0 and 1. It adapts the real values of the final completely connected layer's outputs to the likelihood of the target class. Table 2.1 summarizes the most often used fully linked layer activation functions for various reasons[41].

Table 2.1: List of Commonly Used Fully Connected Layer Activation Functions [41].

Last layer Activation Function	Task
Linear	Continuous-value regression
Sigmoid	Classification with multiple labels, one node, pair node, and binary classification
Softmax	Classification into single and multiclass

2.6 Model parameter and Hyperparameters

A convolutional neural network is made up of both normal and hyper parameters. Model parameters are features that the model employs to adjust to data, such as biases and weights. Hyper-parameters are features or characteristics that must be established in advance in order to order the entire training process (predefined)[64]. Because hyper- parameters not be learnt directly from the training process, they must be pre-determined. The optimal setting of the hyper-parameters results in improved efficiency and outcomes. The kind of activation functions, the learning rate, the number of hidden nodes, the number of hidden layers, the number of periods, the batch size, kernels size, kernel numbers, stride, and padding are among the hyperparameters that the user may configure, as shown in table 2.2[64].

Table 2.2: Parameters and Hyper Parameters[64].

Layers	Parameter	Hyper parameter
Convolutions	Kernels	Size of kernels, numbers of kernels, stride, activation functions and padding
Pooling	-	padding, stride, and filter size
Fully Connected	Weights	Number of weights and the activation function
Other		Model structure, loss function, optimizer, learning rate, epochs, dataset splitting, mini batch size, weight initialization, and regularization.

objectives[65]. A loss function, also known as a cost function, calculates the difference between the network's output predictions and assigns ground truth labels via forwarding propagation [66]**.** The cross-entropy function or the soft-max loss function, the Euclidean loss function, and the hinge loss function are among the most prominent loss functions used to measure the performance of the convolutional neural network model. However, the most commonly used is the cross-entropy loss function as an alternative to the square error loss function in multi-class classification problems, also known as the log loss function, whose product is probability p 0,1. It employs softmax activations in the output layer to create output with a probability distribution, i.e. p, y RN, where p signifies the probability of each output class and y the intended outcome, and the probability of each output class may be computed as follows [67]**:**

$$p = \frac{e^{a_i}}{\sum_{k=1}^{N} e^{a_k}} \qquad (2.13)$$

Where N is the number of neurons in the output layer and e^{a_i} signifies each normalized output from the network's preceding layer. Finally, cross-entropy loss is defined as:

$$H(p, y) = -\sum y_i \ log(p_i) \qquad (2.14)$$

Where i € [1, N]

despite the fact that mean squared error (MSE) is commonly employed for regression to continuous values[67].

2.8 Network Training and Main Challenges

One of the network training strategies is a convolution of kernels in convolutional layers and weight found in fully connected layers. The backpropagation algorithm, a popular method for training neural networks, lowers the distance between the ground truth labels in the training dataset and the corresponding labels in the output predictions. The loss function and the gradient descent optimization technique play the most crucial roles[68]. In a training data set with defined kernel parameters and weights, the model output is computed using a loss function and forward propagation. Optimization methods such as gradient descent and backpropagation are used to alter learnable parameters such as kernels and weights based on the value of the loss. The purpose of a convolutional neural network model's training or learning process is to enhance the model's accuracy. Underfitting happens when a model fails to have a loss as low as the needed degree in the training set, implying that the model fails to learn the important patterns in the training data[69]. Overfitting, on the other hand, happens when the network learns the training data well and performs well, but does not perform as well on the new dataset, such as the validation data. As a result, the network is incapable of generalization. Overfitting happens when the discrepancy between the training and validation errors is too large. Figure 2.9 depicts the connection between underfitting and overfitting in regard to model potential. [70]:

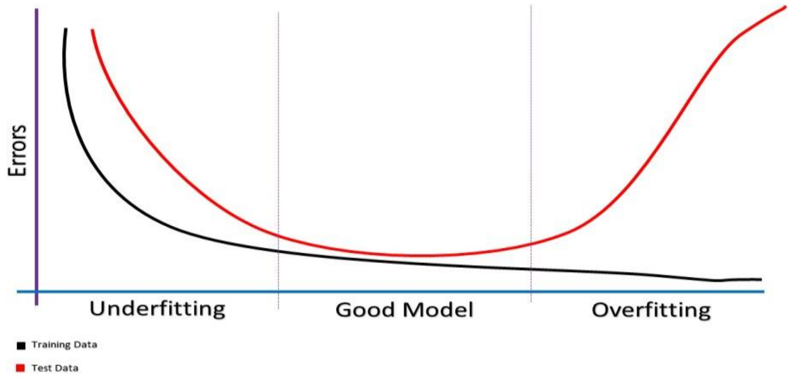


Figure 2.9: Overfiting and Underfiting Problem[70].

The challenges in identifying any infection are mostly due to the wide variation in shape, volume, location, regularity, and The emergence of non-homogeneous (for example, the severity, variance, and tissue variety of diverse diseases). Another problem is gathering sufficient data from a single source and dividing it into training, validation, and testing sets. The quality of the data used to train the model has a significant impact on the accuracy of the findings and, as a result, the model's dependability for usage in a critical sector such as medicine. Because it contains many parameters that are meant to learn, the training set must be large enough to update and specify parameters for deep neural networks, and the restricted number of training data causes the problem of overfitting. In order to make neural networks more generalizable, it is anticipated that the form, size, appearance, and location of the affected (such as intensity, variance, and tissue diversity of various infections) should be as varied as possible in the training data. To overcome most of the challenges and build a credible model, neural networks must employ a large amount of varied training data[71].

The purpose of utilizing a validation set is to fine-tune some hyper parameters and assess how successfully the neural network has been trained. If the test set is larger than the training data set, there will be less training and validation data, which is undesirable since training data is insufficient for neural networks to create a correct model. By using enhanced data, the number of training and validation data sets is quadrupled. This is accomplished via zooming, vertical and horizontal flipping, cropping, and rotating photos. As a result, the goal of training a machine learning model is to[68]:

1- Reduce training loss as much as feasible.
2- As a result, the time between training and testing is kept to a minimum.

The "optimal capacity" of the network can be surpassed by increasing the capacity of the network model (owing to deeper networks, more neurons, no regularization, etc.). At the stage of expanding the capacity of the model, the error (loss)/accuracy of training and validation begin to converge, and a perceptible gap appears. The goal is to close this gap while maintaining the model's generalizability. As a result, if this space is not filled, the "overfitting component" will enter[68].

The training loss will either remain constant or decrease at this stage, while the validation loss will either remain constant or grow. The increase in validation failures over a long period of time is a clear signal of overfitting. There are two broad approaches to dealing with the problem of overfitting:

1- Choose a shallow network with fewer layers and neurons to reduce the model's complexity.

2- Make use of regularization techniques.

Fewer neural networks can operate with smaller datasets, but they are not always the best option. Regularization procedures such as weight decay, dropout, data augmentation, and so on should be employed instead. It is usually safer to employ regularization techniques to check overfitting rather than a huge network size to monitor overfitting. The vanishing gradient problem happens during backpropagation through a deep convolutional neural network (CNN with many layers, say 1500). It needs computing loss (error) gradients with respect to the associated weights in the neurons of each layer in order to update those weights. When a result, as the network moves backwards, lowering gradients, the early layer neurons will get extremely small gradients close to or equal to zero, resulting in a very modest refresh in the weights of the early layer and learning these layers very slowly and inefficiently. To assist with avoiding this issue, ReLU/ leaky In the CNN architecture, ReLU can be utilized as the activation function instead of the others (sigmoid, tanh). This issue can also be resolved by employing batch normalization layers[70].

When backpropagation accumulates huge error gradients, resulting in very large changes to network weights, the model becomes unstable and cannot be trained or extended effectively. This is known as an explosion issue, which is the inverse of a vanishing gradient problem. This issue may be addressed by employing various weight regularization approaches or by changing the topology of the established network model[35]. The training procedure consists mostly of the following steps: Data pre-processing, data enhancement, parameter initialization, CNN regularization, and optimizer selection[70].

2.8.1 Parameter Initialization

Millions or billions of parameters make up a deep CNN. This necessitates that it be properly initialized at the start of the training process, since it is critical to the speed and accuracy of the CNN model. Random initialization is one of the approaches used in parameter initialization in CNN. Millions or billions of parameters make up a deep CNN. This necessitates that it be properly initialized at the start of the training process, since it is critical to the speed and accuracy of the CNN model[72]. Random initialization, which comprises of numerous approaches such as Gaussian random initialization, uniform random initialization, and orthogonal random initialization, is one of the strategies used in CNN parameter initialization. The biggest downside of random initialization is that it might lead to disappearing or exploding gradients[72].

2.8.2 Optimizer Selection

Optimizers in CNNs monitor the convergence process and increase classification accuracy. As optimizers, researchers now employ stochastic gradient descent (SGD), adaptive moment estimation (Adam), and RMSProp**.** Gradient descent is a prominent optimization approach whose purpose is to decrease loss (the difference between the expected and actual output) by often changing the network's learnable parameters, such as kernels and weights[73]. Backpropagation of the gradient modifies the learnable parameters in the inverse direction of the gradient, using a hyper parameter termed the learning rate, which is defined by the random step size figure 2.10. The following is an example of a single parameter change:

$$w \coloneqq w - \alpha * \frac{\partial L}{\partial w} \tag{2.15}$$

Where w denotes learnable parameters, L denotes the loss function, and α indicates the learning rate. It's worth mentioning that the learning rate is one of the most important hyper parameters that must be selected before beginning training[74].

In practice, the parameterized loss function gradients are generated using the mini-batch, which is a subset of the training dataset, and memory limitations are applied to the parameter changes[75]. The phrase (mini-batch gradient descent) MBGD process, also known as (stochastic gradient descent) SGD, refers to one of the hyper parameters. During each training period, the model's parameters are continually updated, and the model repeatedly seeks for the best solution locally in each training epoch. Stochastic Gradient Descent (SGD), Batch Gradient Descent, Mini Batch Gradient Descent, Momentum, AdaGrad, AdaDelta, RMSProp, and Adaptive Moment Estimation are the most prominent learning algorithms or optimization methods. RMSProp, Adam, and SGD with momentum have all been widely employed [76].

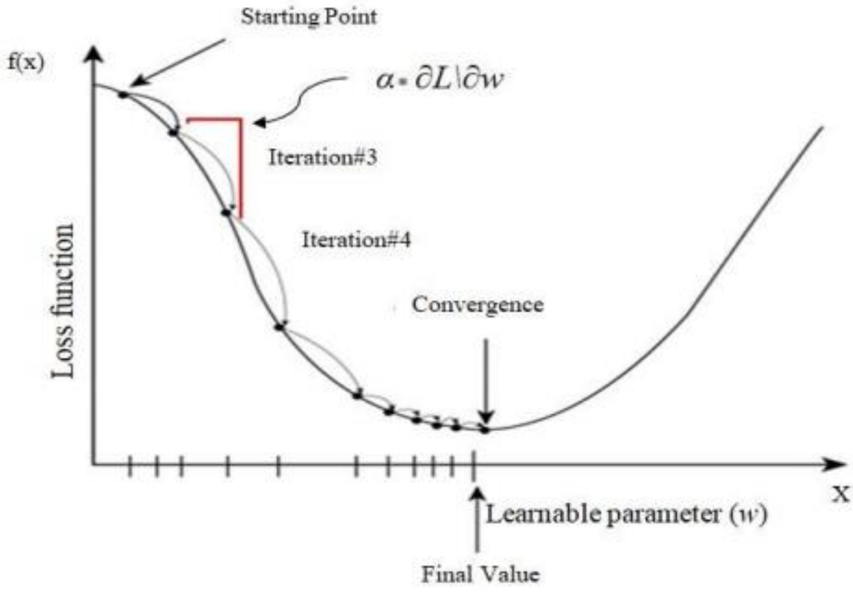


Figure 2.10: Gradient Descent[76].

2.8.3 Regularization to CNN

Regularization uses various intuitive notions to help avoid overfitting, including dropout, data augmentation, early stopping, and batch normalization. Some of which are addressed in the following subsections [77]:

2.8.3.1 Batch Normalization

The distribution of each input layer changes as the parameters of the previous layers change. This leads to the complexity of training deep neural networks. This slows down the training process by necessitating lower learning rates and accurate parameterization, and makes training models very difficult. Internal covariate shift is the term use to describe this phenomenon, and fix it by normalizing layer inputs When build the normalization into the model architecture then give it strength especially for each training mini-batch. Batch normalization allows for much higher learning rates to be used with less care in initialization, and it also eliminates the need to dropout in certain cases, outperforming the original model by a large margin [78].

2.8.3.2 Dropout

The dropout regularization technique may be used in the fully connected layers to avoid overfitting and the ability to efficiently combine multiple neural network structures. The dropout refers to the dropping of visible and hidden units into a neural network. Dropping a unit randomly means it is temporarily removed from the network with all its outgoing and incoming links [79]

2.9 Evaluation Matrices

An important number of classification measures are used in order to analyze, evaluate and compare the results robustly and effectively, as well as to know how the model performs and implements the objective task accurately in the test dataset, which the network model has never seen before. For example, classification accuracy measures are used. Accuracy and sensitivity are all related to loss[80].

2.9.1 Accuracy

The technique for calculating model accuracy is given by the number of correctly identified samples divided by the total number of samples in the testing dataset[81]. As seen in equation (2.17)

$$\text{Accuracy} = \frac{\sum_{i=1}^{N} TP\ (\text{Ci})}{\sum\sum C\ \text{i,j}} \qquad (2.16)$$

Where TP=true positive, Ci, j= (TN=true negative, FP=false negative, FN= false negative, TP).

2.9.2 Precision

Precision is the number of expected values that were really correct. Then it was computed in equation (2.18) [81].

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}} \qquad (2.17)$$

Where TP=true positive, FP=false negative.

2.9.3 Recall

The number of correct positive predictions (TP) divided by the total number of positive predictions (TP) yields recall (P). Then it was computed in equation (2.19) [65].

$$\text{Recall} = \frac{\text{TP}}{\text{P}}\ ,\ \text{where P} = \text{True positive} + \text{False Nagative} \qquad (2.18)$$

2.9.4 Specificity

The number of true negative predictions (TN) divided by the total number of negatives is used to assess specificity (N). Then it was computed in equation (2.20) [65].

$$Specificity = \frac{TN}{TN+FP} \qquad (2.19)$$

2.9.5 F-Score

The f-measure or f-score is a test accuracy metric. It is computed using precision and reminders[65]. Then it was computed in equation (2.21).

$$F1 - Score = \frac{2 \times P \times R}{P + R} \quad (2.20)$$

Where p is precision, R is recall

2.9.6 Confusion Matrix

A binary classifier's confusion matrix. Actual values are labeled true (1) and false (0), whereas anticipated values are labeled positive (1) and negative (0). Estimates of classification model possibilities are generated using the confusion matrix expressions TP, TN, FP, and FN. Classification having more than two classes is referred to as multiclass classification [80]. See table 2.3 show an example for multi class classification [82].

Table 2.3: Confusion Matrix[82].

		True Class		
		A	B	C
Predicted class	A	TP_A	E_{BA}	E_{CA}
	B	E_{AB}	TP_B	E_{CB}
	C	E_{AC}	E_{BC}	TP_C

2.10 Summary

This chapter summarizes the theoretical foundations of bone age assessment. It also explains deep learning and some techniques, including convolution neural networks. The system's implementation and design will be discussed in the next chapter. It also uses a set of standards, including accuracy, recall, precision, and F1-score, to test the performance of algorithms and the proposed model to measure the rate approach in general.

CHAPTER THREE

PROPOSED MODEL

3.1 Overview

The main goal of this thesis is to design an intelligence system for bone age assessment based on an x-ray image of the left hand. In this chapter, the proposed system presented in is implemented two main modes: training and test modes. Each mode contains several Phases and steps.

3.2 Proposed Model

Build a new deep Neral Network architecture to identifying age group utilizing x-ray images. To accomplish this aim. The proposed system presented a bone age assessment system based on left hand x ray images.

The two phases (modes) of the proposed model execution is a training mode and a testing mode:

- Training phase: The proposed model trains each training existing datasets for learning on samples with their class by feeding sample training samples with appropriate labels. This stage also includes the validation process.
- Testing phase: The proposed trained model is put into practice this stage is to simulate the use of the system by predicting the output class from unseen datase

Stages of working on the model: data preprocessing, model construction, model training, model evaluation and then testing. Figure 3.1 show the digram block of the proposed deep neural network model.

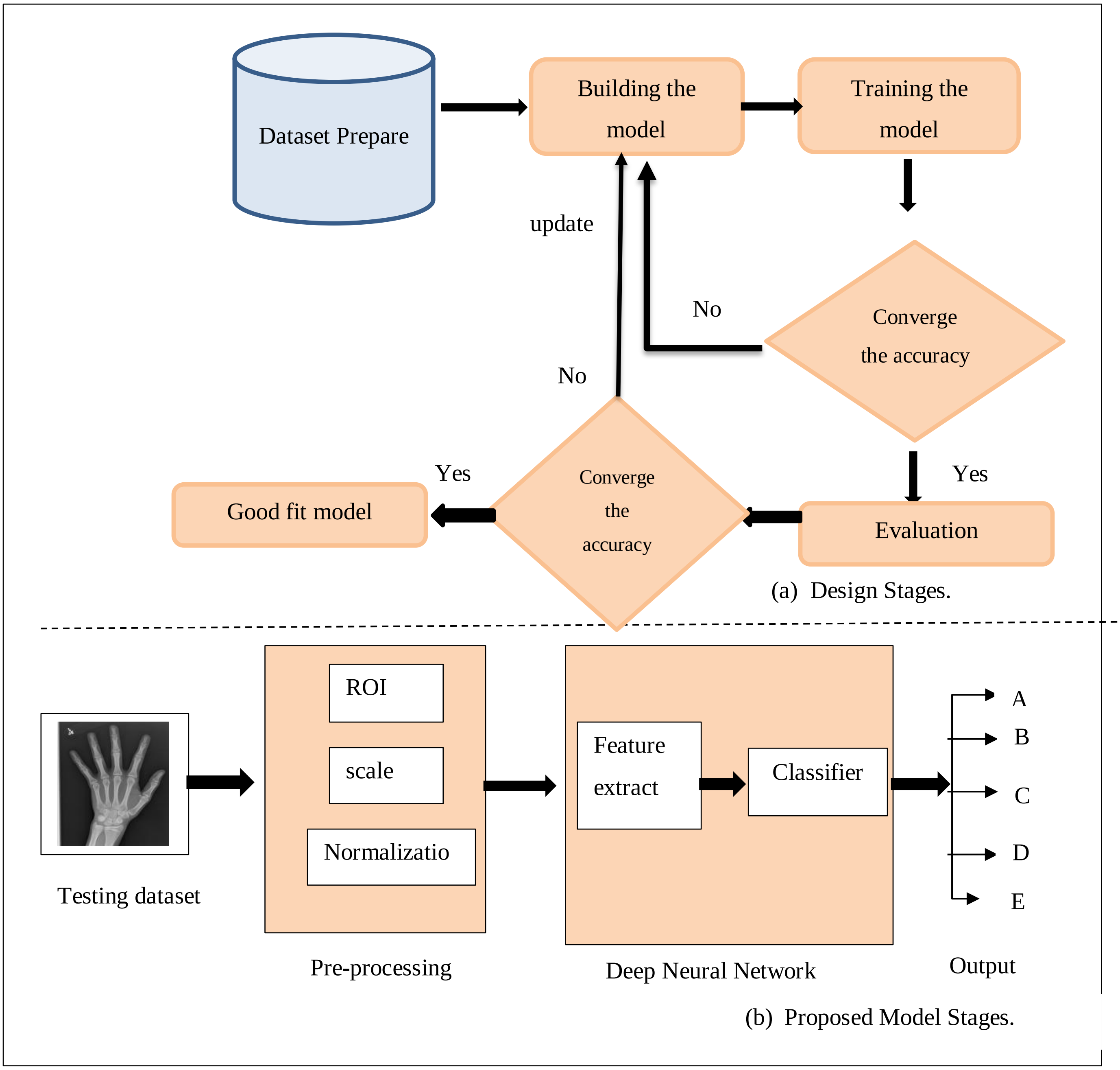


Figure 3.1: Block Diagram of the Proposed System.

3.2.1 Data preprocessing

The preparation of data for neural network modelling depends on the model's input needs. For effective model training, initializing and converting raw data may be an important step in the pre-processing process. Image convert to grayscale grids of pixels, image values rescaling from 0 to 255 to [0,1] interval, and image improvement and noise removal are all included in these pre-processes. Image rescaling is a vital aspect of image processing since it is often necessary to increase or decrease the size of an image. To illustrate this, it reduced the data image size to 224x224 for usage in this work**.** Almost every image in our x-ray datasets contains undesired spaces and areas, leading to poor classification performance. Hence, it is necessary to crop the images to remove unwanted areas and use only useful information from the image. Use the cropping method, which uses extreme point calculation. The step to crop the left hand x- ray images using extreme point calculation is shown in Figure3.2.

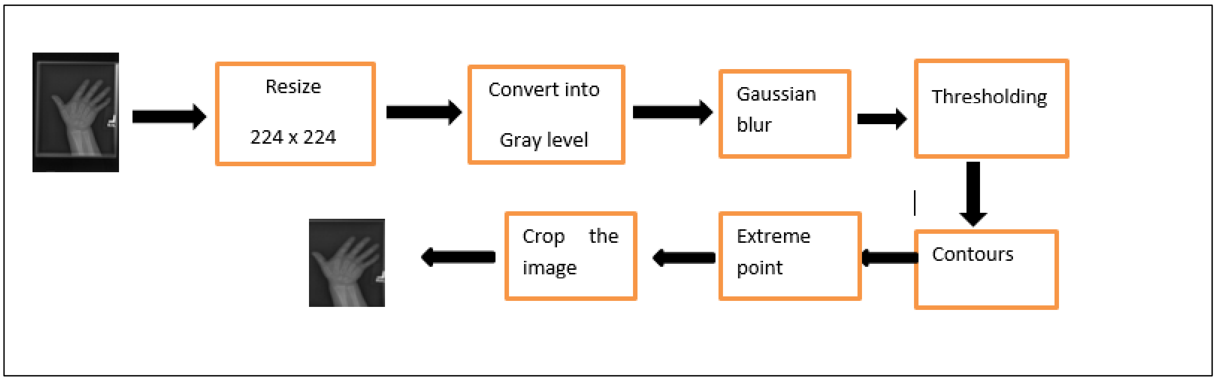


Figure 3.2: Steps of Crop for X-Ray Dataset.

3.2.2 Splitting Dataset

In the data split stage, the percentages of training, validation, and testing data are determined, with 70 % of training data, 15 %validation data, and 15 % testing data are being divided. This is the ratio given the best result after trying several divisions. Figure 3.3 shows the dataset's split ratio.

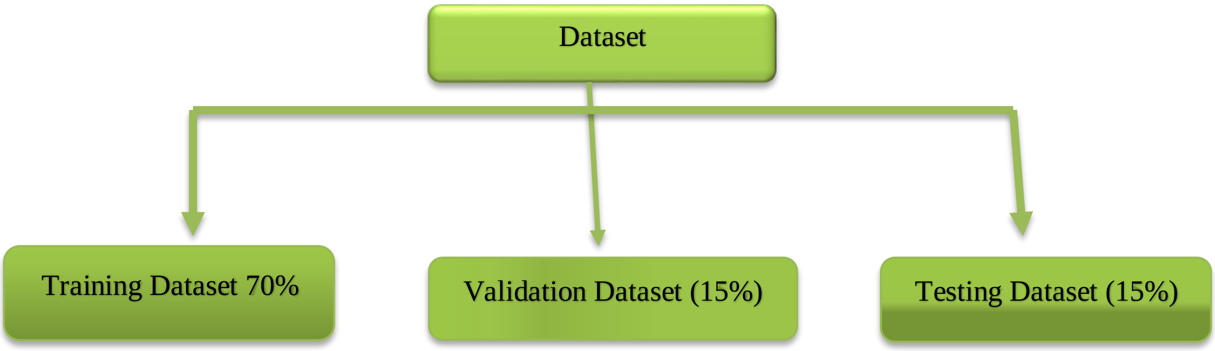


*Figure 3.3: The Data Set Division***.**

3.2.3 Building The Model

Many experiments have been implemented to get the best result with changing some parameters such as the number of epochs, learning rate, and changing the data division ratio. In this section, will discuss the best-proposed model's design, which includes, convolutional layer count, pooling layer count, and other structural elements. Model was built to classify x-ray images for bone age group. Creating a CNN takes numerous steps, each of which must be completed in the correct order. In this stage, need to supply four parameters for use in the first convolutional layer (the size and number of filters, the activation function, and the size of the input image). Finding features in an image using a feature detector (filter) and then storing them in a feature map is the primary goal of convolution processing.

The basic design was utilized, with the revised network consisting of a stack of seven alternating Conv2D layers (with ReLU enabled), Batch Normalization, and MaxPooling2D with filter size 2 x2. In the first layer, 64 filters of sizes 3x3 were employed, followed by 64 filters of the same size in the second block. The first and second blocks made use of the activation function ReLU and the filter size 2 x 2 of maxpooling2D. The third and fourth blocks used the same size with 128 filters, followed by 256 filters in the five and six layers and then 512 filters in the seven conv2D layer. Next, the flat layer flattens the matrix after the seven convolutional layers. There are 256 neurons with ReLU and dropouts (0.2) is used in this work after each dense layer in the final fully-connected layer associated with the classification layer. The dropout layer is used to reduce neural network complexity by randomly deleting some nodes. Uses softmax and dual-multioutput to identify test image in an output layer. In this method, the neural network capacity is increased and the size of the feature maps is reduced so that they are not excessively enormous when the normalization layer is reached. The x-ray image upload are 224 x 224 pixels (random selection), and end up with 7×7 feature maps just before the flattening layer. As the network grows, so does the depth of the feature maps. In the initial convolution, feature maps will have a depth of 64, which will expand to 512 in the seven-layer convolution. It shrinks from 224x224 to size 7 x 7. There are 4608 nodes in a flat layer, followed by 1024 nodes in a dense layer. Softmax is enabled as a result of the categorical task being utilized, and the network now has many units (dense layer size = 5). It's seen in figure 3.4. Table 3.1 illustrate the proposed model.

Table 3.1: Summary of the Proposed System.

Layer(Type)	Output shape	Parameters
conv2d (Conv2D)	(None, 224, 224, 64)	640
Batch_ normalization	(None, 224, 224, 64)	256
conv2d_1	(None, 224, 224, 64)	36928
max_pooling2d	(None, 112, 112, 64)	0
batch_normalization_1	(None, 112, 112, 64)	256
conv2d_2	(None, 112, 112, 128)	73856
max_pooling2d_1	(None, 56, 56, 128)	0
batch_normalization_2	(None, 56, 56, 128)	512
conv2d_3	(None, 56, 56, 128)	147584
max_pooling2d_2	(None, 28, 28, 128)	0
batch_normalization_3	(None, 28, 28, 128)	512
conv2d_4	(None, 28, 28, 256)	295168
max_pooling2d_3)	(None, 14, 14, 256)	0
batch_normalization_4	(None, 14, 14, 256)	1024
conv2d_5	(None, 14, 14, 256)	590080
max_pooling2d_4	(None, 7, 7, 256)	0
batch_normalization_5	(None, 7, 7, 256)	1024
conv2d_6	(None, 7, 7, 512)	1180160
max_pooling2d_5	(None, 3, 3, 512)	0

batch_normalization_6	(None, 3, 3, 512)	2048
flatten	(None, 4608)	0
dense	(None, 1024)	4719616
dropout	(None, 1024)	0
dense_1	(None, 512)	524800
dropout_1	(None, 512)	0
dense_2	(None, 5)	2565

Total parameters: 7,577,029
Trainable parameters: 7,574,213
Non-trainable parameters: 2,816

The total parameters obtained from sum of all layers parametares except the parametares of batch normalization layer that is represent non trainable prameters. The prameter of each convolutional layer compute by using equation mentioned in chapter tow also the prameters of flatten layer.

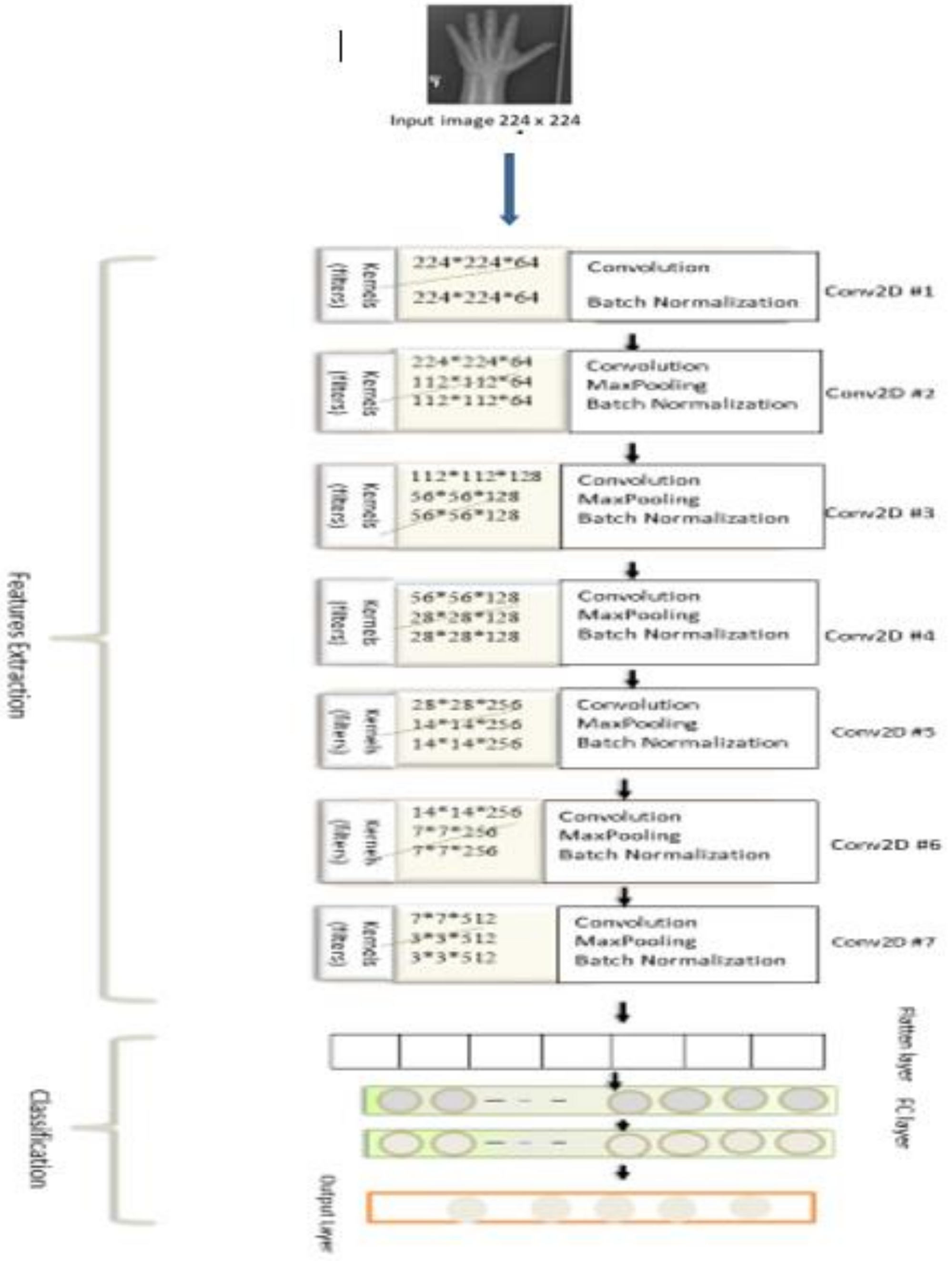


Figure 3.4: Architecture of the Proposal CNN.

3.2.4 Compile the Model

Many arguments we're utilizing in this model are the loss function, the optimizer, and the metrics. Categorical cross entropy was employed as the loss since the neural network has a multi softmax unit at the end of the network. SGD optimization used. Accuracy is used to calculate evaluation metric.

3.2.5 Fit the Model

The generator is used to fit the model to the data. In this case, the fit generator method may be used, which will generate inputs and outputs in ever-increasing batches. The steps per epoch input is used to determine how many data points will be drawn from the generator before the model declares an epoch. The steps per epoch parameter serves this purpose. The CNN model was fitted into the training dataset using 1192 iterations and each iteration includes distinct training steps and assesses the steps depending on the length of the test and training sets. For each batch, the network model receives 32 samples of training data, thus we have 12611 training photos to feed into the network model. A single epoch has 394 batches, each of which contains 32 image, and the weights of these batches, which are computed before going on to the next epoch, are called an adapted weight. Fit generator accepts a validation data parameter, much like the fit. Note that this parameter is permitted to generate data. Validation data is supposed to be generated indefinitely, hence the validation steps option specifies how many batches to extract from the validation constructor when evaluating the validation data.

Overfitting, which prevents the trained model from being generalizable to new data. It has been addressed by increasing the volume of training data and making it more accessible through a variety of random changes that yield image that appear to be logical. This technique uses rescale, shear range, zoom range, and flip as its four arguments. The suggested model's arguments may be utilized to explain the data augmentation technique's mechanism:

- Getting image into the network requires rescaling the pixel values (between 0 and 255) to the interval [0,1] as shown in equation (2.1) in chapter two.
- If we want to randomly zoom in on a image, you may use Zoom range, which has been set to 0.2.
- Randomly executing shearing transformations, the shear range parameter was assigned a value of 0.2.
- A random image flipping function called Flip was created with a value equal to (True).

3.2.6 Model Evaluation

When the training of the model is complete, it need to evaluate the performance and determine whether suitable for use with the training dataset or not, it has to be a criterion to determine how well the model performed. During the training of the model, a dataset is utilized. However, apart from the training data, which is referred to as the validation dataset, this data keeps an eye on the validation accuracy and validation loss indicators, among others. Subsequently, contrasts them with the accuracy of the model training and the loss of the model training. It is important to have this data in order to prevent overfitting, which occurs when the accuracy of the model training improves while the accuracy of the validation worsens. The objective of collecting these data is to prevent overfitting. On the other hand, the validation loss indicator of the model is contrasted with the training loss indicator of the model. Underfitting is shown when the model's training loss goes down but the validation loss continues to go up, since this indicates that the model is being overfit. By expressing it as an observer of the model's performance, the validation dataset enables the model to generalize to new data. This allows the model to be more accurate. According to the information presented in section (3.2.2), the size of the validation dataset corresponds to around 15 % of the size of the training dataset.

The performance of the model and its capacity to generalize to new data that the model has not previously encountered are both evaluated at the stage designated as the "evaluation". This is accomplished by the utilization of a number of metrics, including accuracy, recall, precision, and the f1-score, the details of which were elucidated in chapter three. Algorithm 3.1 illustrates the model evaluation step.

Algorithm 3. 1 Evaluation of the Model

```
Input: Testing dataset
Output: Accuracy, Recall, Precision, F1-score
1. Load proposal CNN Model
2. Compile model
3. Calculate Accuracy                 / Equation (2.16)
4.Calculate precision                 / Equation (2.17)
5.Calculate recall                    / Equation (2.18)
6.Calculate F1-score                  / Equation (2.20)
7.Calculate Confusion matrix          / Table (2.3)
9. Proposal CNN Model= Optimal Generalizable Model
10. Save Optimal Generalizable Model
11`.End
```

3.3 Summary

This chapter explores the architecture characteristics & learning set up to design the proposed CNN model. In addition, it describes how data splitting into training, validation, and testing.

CHAPTER FOUR

RESULTS AND DISCUSSION

4.1 Overview

The results that achieved by training our suggested architecture in the classification challenges will be shown in this chapter. Clarification will be provided on the correctness of the findings produced by this model, and it will be compared with the results produced by models for related works in the fields of bone age assessment.

4.2 System Requirement

In order to implement the suggested system, the following will be used: Hardware / Central Processing Unit (CPU): Intel(R) Core(TM) i7-10750H CPU @ 2.60GHz. RAM 16GB, NVIDIA GeForce GTX 1660 Ti. Operating system: is Windows 10 (64 bit) and used python as program language.

4.3 Experimental Result

The experiments and tracked the performance determine the efficient model based on performance metrics. Before the results are offered, a background on the experimental setting is provided. This study used datasets available on the Kaggle web and the details in the next section.

4.3.1 Dataset

A public dataset was obtained from the RSNA pediatric bone age machine learning challenges. It comprises 12611 x-ray image, 6833 of which are for males and 5788 of which are for female. The majority of the samples are labeled between the ages of 96 and 168 months, with samples from other age groups being quite minor. The dataset obtained from the website: (https://www.kaggle.com/kmader/RSNA-bone-age), it is avalibale with CSV file content the id,bone age, and gender for each x_ray image. the CSV file use to labeled the x-ray images after convert tha ages from month to years and then divided into five classes.The detiales information of dataset show in tabel 4.1. Figure 4.1 is example of dataset that using in this work.

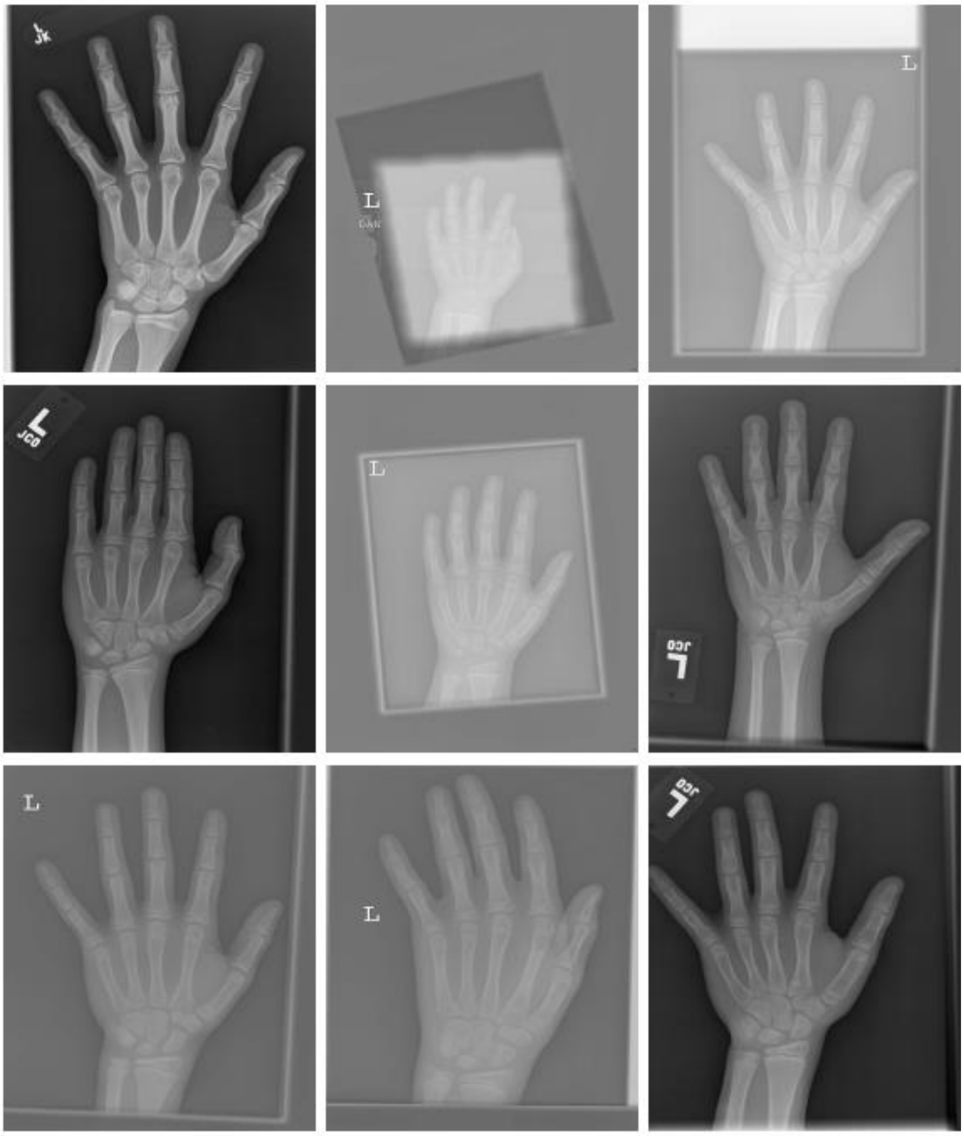


Figure 4.1: Examples of The Left Hand X-Ray Images Dataset[RSNA].

Table 4.1: Information of RSNA Dataset Before Augmentation.

Classes	Age group (year)	Number of images
A=0	[1-5]	1083
B=1	[6-8]	2179
C=2	[9-11]	3322
D=3	[12-14]	4832
E=4	[15-18]	1201

4.3.2 Proposal CNN Model

The metrics were applied to various training and testing samples while taking into consideration the architecture of the neural network as well as the number of convolution and dropout layers contained in the deep model. Consequently, because of these criteria, the model is able to categorize x-ray left-hand image and determine the bone's age group.

Several experiments tried to get the best result with changing some parameters such as the number of epochs and the learning rate. Additionally, adding or deleting layers to reach the best result. The results of some of those experiments as shown in the table (4.2). The first model designed consisted of 5 conventional layers, after each layer was maxpooling and two denes layers with output layer. The accuracy for testing was 72% for 100 epochs when increased the number of epochs obtained overfit. Another model was the same as the first one with the addition of droupt after each layer. The accuracy of testing was 81% for 200 epochs with a learning rate of 0.001 and used Adam optimizer. When the number of conventional layers was increased to 7 layers and after each layer was used maxpooling and then batch normalization layer. The accuracy of testing started to improve to 93%., The best result obtained when used the same previous neural network wih changed in some parameters such as learning rate were changed to 0.01 and optimizer was replaced with SGD the accuracy of the testing began to improve, it settled on 97%. When increased the number of epochs to500 epoch. The final hyper parameters that used in the best architecture describe in table 4.2

Table 4.2: Details of The Results Obtained During Testing of The Model.

Number of Epoch	**Recall**	**Precision**	**F1-score**	**Accuracy**
100	71.4%	72%	72%	72.31%
200	80.06%	81.98%	79.68%	81.07%
300	86%	87.80%	86.70%	86.85%
400	93.84%	93.96%	93.77%	93.84%
500	97%	97.26%	97.22%	97.33%

Table 4.3: Hyper Parameter of The Implemented Model.

Hyper parameter	Model architecture
Optimizer	SGD
Kernel size	3x3
Padding	Same
Pooling stride	2x2
Learning rate	0.01
Batch size	32
Epochs	500

Before augmentation the training folder had 9088 image, after augmentation the training data have 36352 image, the testing folder contained 1891 images, and the validation folder contained 1632 images.

Each data folders are subdivided into five distinct folders, and each of those folders has a varied number of x-ray image. Accuracy plot is displayed in figure (4.2) along with the loss and validation loss.

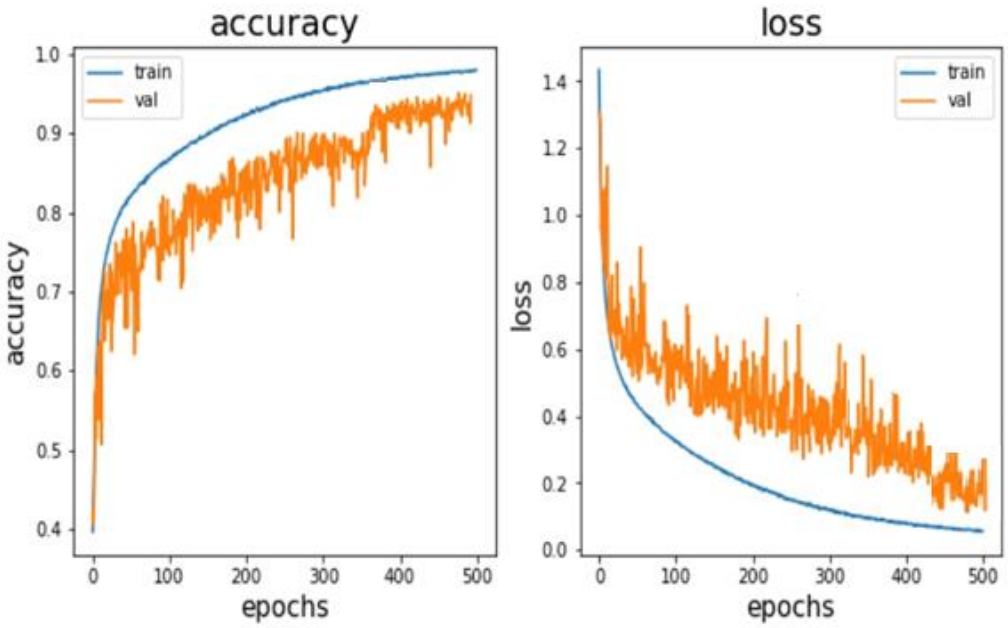


Figure 4.2 Accuracy and Loss Function of the Proposed Architecture.

This neural network model's classification result can be seen for images from a subset of selected test data, which demonstrates how each image is classified by first comparing its values to those of all the other images in its category (the range of bone age) and then selecting only those with a higher value, which represents the classification for each image. There are other methods of testing model correctness, such as comparing classification results with those from the original dataset (labels). Figure 4.3 depicts the model's confusion matrix for the five classes (A, B, C, D, E). Table (4.4) provides further context for the model categorization report and metrics data.

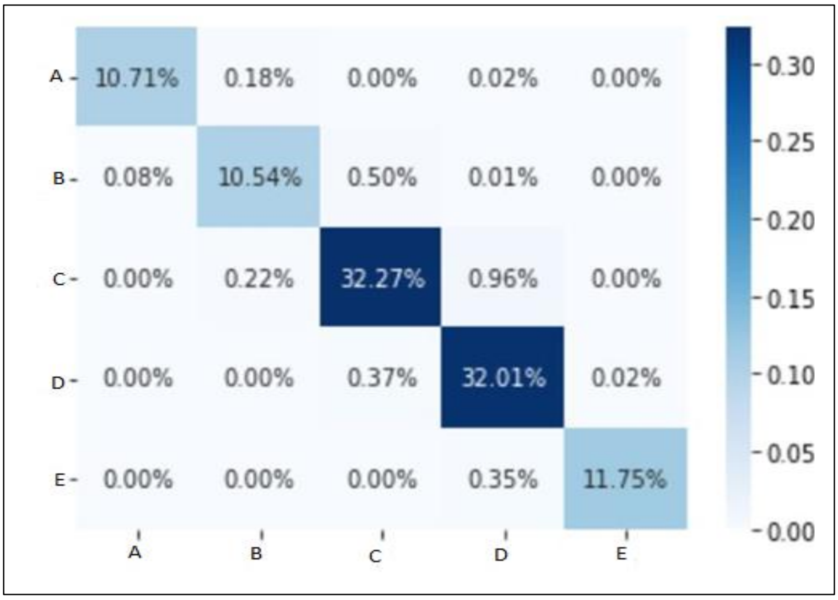


Figure 4.3: Confusion Matrix of The Proposed Architecture.

Table4.4: Model Classification Report and Metrics Values.

Performance Metrics (%)	**Age Classes**				
	A = 0	B =1	C=2	D=3	E= 4
Accuracy	97%	95.5%	96.5%	98.9%	95.8%
precision	99%	96%	98%	95%	100%
Recall	97%	96%	97%	99%	96%
F-Measure	98%	96%	97%	97%	98%

4.3.3 Comparison with State of the Art Models

An assessment of proposed CNN model is now complete, and we've derived findings for accuracy. The model's output compared with the related works on the same dataset and used the classification to solve the problem of bone age for both gender [12],[18],[24] .These results are shown in the table 4.5.

Table 4.5 :Comparison between Our Model Proposed and Related Works[12], [18], [24] .

Author	Dataset	Methodology	Accuracy
Larson at el.[12]	RSNA	Resnat101	63%
Pengyi Hao, at el. [24]	RSNA	ResNet-101	76%
		Inception-v3	80%
N. Mualla at el. [18]	Used 1684 Sample of RSNA only	AlexNet and ResNet-101for extract feature and DT,SVM clasiffier	100%
Our proposed model	RSNA	DCNN	97.22%

4.4 Summary

This chapter evaluates the performance of bone age assessment. Four performance measures were calculated: accuracy, precision, recall, and f1-score. In next chapter will explain the conclusion and future works.

CHAPTER FIVE

CONCLUSION AND FUTURE WORK

5.1 Overview

This chapter shows the conclusions of the thesis, and some suggestions for the future works.

5.2 Conclusion

During the period of this study, there are many observations that can be made the results of this study concluded:

- The classification system using deep neural networks can be considered the best approach to achieve high accuracy and give better results than other traditional approaches in terms of accuracy and loss functions.
- Adding batch Normalization after each convolutional layer decreases the overfitting in this work.
- Replace the Adam optimizer with SGD Contribute to raising accuracy and getting rid of overfitting.

Limitations in this work: the number of data is small, and it requires high specifications in both software and hardware.

5.3 Future Work

One or more of the following ideas for further research might be of interest:

- ROI and segmentation with x-ray hand image to improve training and testing accuracy.
- This work can be used to assess the age of antiques, or it can be used to determine the age of the person involved in accidents.
- Use deep auto encoder techniques for feature selection and reduction and examine how the suggested system's performance is influenced.

REFRENCES

REFRENCES

[1] P. Hao, S. Chokuwa, X. Xie, F. Wu, J. Wu, and C. Bai, “Skeletal bone age assessments for young children based on regression convolutional neural networks,” *Math. Biosci. Eng.*, vol. 16, no. 6, pp. 6454–6466, 2019, doi: 10.3934/mbe.2019323.

[2] B. D. Lee and M. S. Lee, “Automated Bone Age Assessment Using Artificial Intelligence: The Future of Bone Age Assessment,” *Korean J. Radiol.*, vol. 22, no. 5, pp. 792–800, 2021, doi: 10.3348/kjr.2020.0941.

[3] J. E. Pinsker, M. S. L. Ching, D. P. Chiles, M. Lustik, C. B. Mahnke, and V. J. Rooks, “Automated bone age analysis with lossy image files,” *Mil. Med.*, vol. 182, no. 9, pp. e1769–e1772, 2017, doi: 10.7205/MILMED-D-17-00011.

[4] P. D. C. E.-T. H.-J. S.-H. S.-R. J.-J. Nguyen, “Multiple Inputs Deep Neural Networks for Bone Age Estimation Using Whole-Body Bone Scintigraphy,” *J. Korea Multimed. Soc.*, vol. 22, no. 12, pp. 1376–1384, 2019.

[5] A. Hosny, C. Parmar, J. Quackenbush, L. H. Schwartz, and H. J. W. L. Aerts, “Artificial intelligence in radiology,” *Nat. Rev. Cancer*, vol. 18, no. 8, pp. 500–510, 2018, doi: 10.1038/s41568-018-0016-5.

[6] J. Chen, B. Yu, B. Lei, R. Feng, D. Z. Chen, and J. Wu, “Doctor Imitator: A Graph-Based Bone Age Assessment Framework Using Hand Radiographs,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 12266 LNCS, pp. 764–774, 2020, doi: 10.1007/978-3-030-59725-2_74.

[7] E. ÖZGÜL, A. YÜCEL, F. KAYA, S. B. KOCA, and A. ERTEKİN, “Determination of bone age and evaluating the applicability of Greulich-Pyle standards among the Turkish children,” *J. Surg. Med.*, vol. 6, no. 1, pp. 9–13, 2022, doi: 10.28982/josam.1021045.

[8] H. Khalid *et al.*, “A comparative systematic literature review on knee bone reports from mri, x-rays and ct scans using deep learning and machine learning methodologies,” *Diagnostics*, vol. 10, no. 8, pp. 1–43, 2020, doi: 10.3390/diagnostics10080518.

[9] D. Govender and M. Goodier, “Bone of contention: The applicability of the Greulich-Pyle method for skeletal age assessment in South Africa,” *South African J. Radiol.*, vol. 22, no. 1, pp. 1–6, 2018, doi: 10.4102/SAJR.V22I1.1348.

[10] African Development Bank, “Revised Guidelines on Project Completion Report (Pcr) Evaluation Note and Project Performance Evaluation Report (Pper),” vol. 19, pp. 16–41, 2001, [Online]. Available:

https://www.afdb.org/fileadmin/uploads/afdb/Documents/Evaluation-Reports-_Shared-With-OPEV_/00157887-EN-2001-REVISED-EVALUATION-GUIDELINES.PDF.

[11] A. Rakhlin, “Diabetic Retinopathy detection through integration of Deep Learning classification framework,” *bioRxiv*, p. 225508, 2018, [Online]. Available: https://www.biorxiv.org/content/10.1101/225508v2%0Ahttps://www.biorxiv.org/content/10.1101/225508v2.abstract.

[12] D. B. Larson, M. C. Chen, M. P. Lungren, S. S. Halabi, N. V Stence, and C. P. Langlotz, “Performance of a Deep-learning neural network Model in assessing skeletal Maturity on Pediatric hand radiographs 1 PEDIATRIC IMAGING: Neural Network to Assess Skeletal Maturity on Pediatric Hand Radiographs Larson et al Materials and Methods Data Acquisit,” *Radiology*, vol. 287, no. 1, pp. 313–322, 2018, [Online]. Available: https://doi.org/10.1148/radiol.2017170236.

[13] C. Spampinato, S. Palazzo, D. Giordano, M. Aldinucci, and R. Leonardi, “Deep learning for automated skeletal bone age assessment in X-ray images,” *Med. Image Anal.*, vol. 36, no. October, pp. 41–51, 2017, doi: 10.1016/j.media.2016.10.010.

[14] H. Lee *et al.*, “Fully Automated Deep Learning System for Bone Age Assessment,” *J. Digit. Imaging*, vol. 30, no. 4, pp. 427–441, 2017, doi: 10.1007/s10278-017-9955-8.

[15] M. C. Chen, “Automated Bone Age Classification with Deep Neural Networks,” *Stanford Univ. USA,Technical Rep.*, pp. 1–7, 2016, [Online]. Available: http://cs231n.stanford.edu/reports/2016/pdfs/310_Report.pdf.

[16] H. Lee *et al.*, “Fully Automated Deep Learning System for Bone Age Assessment,” *J. Digit. Imaging*, vol. 30, no. 4, pp. 427–441, 2017, doi: 10.1007/s10278-017-9955-8.

[17] S. Mutasa, P. D. Chang, C. Ruzal-Shapiro, and R. Ayyala, “MABAL: a Novel Deep-Learning Architecture for Machine-Assisted Bone Age Labeling,” *J. Digit. Imaging*, vol. 31, no. 4, pp. 513–519, 2018, doi: 10.1007/s10278-018-0053-3.

[18] D. Souza and M. M. Oliveira, “End-to-End Bone Age Assessment with Residual Learning,” *Proc. - 31st Conf. Graph. Patterns Images, SIBGRAPI 2018*, pp. 197–203, 2019, doi: 10.1109/SIBGRAPI.2018.00032.

[19] T. D. Bui, J. J. Lee, and J. Shin, “Incorporated region detection and classification using deep convolutional networks for bone age assessment,” *Artif. Intell. Med.*, vol. 97, no. March, pp. 1–8, 2019, doi: 10.1016/j.artmed.2019.04.005.

[20] J. Guo *et al.*, “Automated Bone Age Assessment Using Artificial Intelligence: The Future of Bone Age Assessment,” *Korean J. Radiol.*, vol. 22, no. 5, pp. 792–800, 2021, doi: 10.3348/kjr.2020.0941.

[21] M. Prokop-Piotrkowska *et al.*, “An Ensemble-Based Densely-Connected Deep

Learning System for Assessment of Skeletal Maturity," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 22, no. 5, pp. 1–24, Jan. 2019, doi: 10.3390/diagnostics11050765.

[22] F. Dehghani, A. Karimian, and M. Sirous, "Assessing the Bone Age of Children in an Automatic Manner Newborn to 18 Years Range," *J. Digit. Imaging*, vol. 33, no. 2, pp. 399–407, 2020, doi: 10.1007/s10278-019-00209-z.

[23] N. S. Rani, M. Chandrajith, B. R. Pushpa, and B. J. B. Nair, "A deep convolutional architectural framework for radiograph image processing at bit plane level for gender & age assessment," *Comput. Mater. Contin.*, vol. 62, no. 2, pp. 679–694, 2020, doi: 10.32604/cmc.2020.08552.

[24] P. Hao, "Overlap Classification Mechanism for Skeletal Bone Age Assessment," pp. 0–6, 2021.

[25] K. Li *et al.*, "Automatic Bone Age Assessment of Adolescents Based on Weakly-Supervised Deep Convolutional Neural Networks," *IEEE Access*, vol. 9, pp. 120078–120087, 2021, doi: 10.1109/ACCESS.2021.3108219.

[26] M. Prokop-Piotrkowska, K. Marszałek-Dziuba, E. Moszczyńska, M. Szalecki, and E. Jurkiewicz, "Traditional and new methods of bone age assessment-an overview," *JCRPE J. Clin. Res. Pediatr. Endocrinol.*, vol. 13, no. 3, pp. 251–262, 2021, doi: 10.4274/jcrpe.galenos.2020.2020.0091.

[27] S. H. Tajmir *et al.*, "Artificial intelligence-assisted interpretation of bone age radiographs improves accuracy and decreases variability," *Skeletal Radiol.*, vol. 48, no. 2, pp. 275–283, 2019, doi: 10.1007/s00256-018-3033-2.

[28] M. A. Zulkifley, N. A. Mohamed, S. R. Abdani, N. A. M. Kamari, A. M. Moubark, and A. A. Ibrahim, "Intelligent bone age assessment: An automated system to detect a bone growth problem using convolutional neural networks with attention mechanism," *Diagnostics*, vol. 11, no. 5, pp. 1–24, 2021, doi: 10.3390/diagnostics11050765.

[29] J. Guo, J. Zhu, H. Du, and B. Qiu, "A bone age assessment system for real-world X-ray images based on convolutional neural networks," *Comput. Electr. Eng.*, vol. 81, Jan. 2020, doi: 10.1016/j.compeleceng.2019.106529.

[30] Y. Kang, N. Cho, J. Yoon, S. Park, and J. Kim, "Transfer learning of a deep learning model for exploring tourists' urban image using geotagged photos," *ISPRS Int. J. Geo-Information*, vol. 10, no. 3, 2021, doi: 10.3390/ijgi10030137.

[31] H. Perez, J. H. M. Tah, and A. Mosavi, "Deep learning for detecting building defects using convolutional neural networks," *Sensors (Switzerland)*, vol. 19, no. 16, 2019, doi: 10.3390/s19163556.

[32] V. Dumoulin and F. Visin, “A guide to convolution arithmetic for deep learning,” pp. 1–31, 2016, [Online]. Available: http://arxiv.org/abs/1603.07285.

[33] O. Oren, B. J. Gersh, and D. L. Bhatt, “Artificial intelligence in medical imaging: switching from radiographic pathological data to clinically meaningful endpoints,” *Lancet Digit. Heal.*, vol. 2, no. 9, pp. e486–e488, 2020, doi: 10.1016/S2589-7500(20)30160-6.

[34] A. Khan, A. Sohail, U. Zahoora, and A. S. Qureshi, “A survey of the recent architectures of deep convolutional neural networks,” *Artif. Intell. Rev.*, vol. 53, no. 8, pp. 5455–5516, 2020, doi: 10.1007/s10462-020-09825-6.

[35] J. D. Álvarez, J. A. Matias-Guiu, M. N. Cabrera-Martín, J. L. Risco-Martín, and J. L. Ayala, “An application of machine learning with feature selection to improve diagnosis and classification of neurodegenerative disorders,” *BMC Bioinformatics*, vol. 20, no. 1, pp. 1–12, 2019, doi: 10.1186/s12859-019-3027-7.

[36] G. Eraslan, Ž. Avsec, J. Gagneur, and F. J. Theis, “Deep learning: new computational modelling techniques for genomics,” *Nat. Rev. Genet.*, vol. 20, no. 7, pp. 389–403, 2019, doi: 10.1038/s41576-019-0122-6.

[37] V. Sze, Y. H. Chen, T. J. Yang, and J. S. Emer, “Efficient Processing of Deep Neural Networks: A Tutorial and Survey,” *Proc. IEEE*, vol. 105, no. 12, pp. 2295–2329, 2017, doi: 10.1109/JPROC.2017.2761740.

[38] F. Sultana, A. Sufian, and P. Dutta, “Advancements in image classification using convolutional neural network,” *Proc. - 2018 4th IEEE Int. Conf. Res. Comput. Intell. Commun. Networks, ICRCICN 2018*, pp. 122–129, 2018, doi: 10.1109/ICRCICN.2018.8718718.

[39] A. Bhandare, M. Bhide, P. Gokhale, and R. Chandavarkar, “Applications of Convolutional Neural Networks,” *Int. J. Comput. Sci. Inf. Technol.*, vol. 7, no. 5, pp. 2206–2215, 2016, [Online]. Available: http://ijcsit.com/docs/Volume 7/vol7issue5/ijcsit20160705014.pdf.

[40] D. X. Zhou, “Universality of deep convolutional neural networks,” *Appl. Comput. Harmon. Anal.*, vol. 48, no. 2, pp. 787–794, 2020, doi: 10.1016/j.acha.2019.06.004.

[41] S. Indolia, A. K. Goswami, S. P. Mishra, and P. Asopa, “Conceptual Understanding of Convolutional Neural Network- A Deep Learning Approach,” *Procedia Comput. Sci.*, vol. 132, pp. 679–688, 2018, doi: 10.1016/j.procs.2018.05.069.

[42] J. Á. Román-Gallego, M. L. Pérez-Delgado, and S. V. San Gregorio, “Convolutional Neural Networks Used to Date Photographs,” *Electron.*, vol. 11, no. 2, 2022, doi: 10.3390/electronics11020227.

[43] Z. Liao *et al.*, "CNN Attention Guidance for Improved Orthopedics Radiographic Fracture Classification," pp. 1–12.

[44] Y. Lecun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015, doi: 10.1038/nature14539.

[45] G. Litjens *et al.*, "A survey on deep learning in medical image analysis," *Med. Image Anal.*, vol. 42, no. December 2012, pp. 60–88, 2017, doi: 10.1016/j.media.2017.07.005.

[46] A. Wibisono, J. Rachmad, and E. Anderson, "Deep Learning and Classic Machine Learning Approach for Automatic Bone Age Assessment," *2019 4th Asia-Pacific Conf. Intell. Robot Syst.*, pp. 235–240, 2019.

[47] K. Zaborowicz, B. Biedziak, A. Olszewska, and M. Zaborowicz, "Tooth and bone parameters in the assessment of the chronological age of children and adolescents using neural modelling methods," *Sensors*, vol. 21, no. 18, pp. 1–18, 2021, doi: 10.3390/s21186008.

[48] S. Chopra, P. Bansal, and P. Bansal, "Journal of Advanced Medical and Dental Sciences Research |Vol. 8|Issue 1|," *J Adv Med Dent Scie Res*, vol. 8, no. 1, pp. 184–186, 2020, doi: 10.21276/jamdsr.

[49] A. Patil and M. Rane, "Convolutional Neural Networks: An Overview and Its Applications in Pattern Recognition," *Smart Innov. Syst. Technol.*, vol. 195, pp. 21–30, 2021, doi: 10.1007/978-981-15-7078-0_3.

[50] P. Hao *et al.*, "Radiographs and texts fusion learning based deep networks for skeletal bone age assessment," *Multimed. Tools Appl.*, vol. 80, no. 11, pp. 16347–16366, 2021, doi: 10.1007/s11042-020-08943-1.

[51] B. W. Ballard, *Hands-On Deep Learning for Images with TensorFlow*. 2018.

[52] L. Si, X. Xiong, Z. Wang, and C. Tan, "A Deep Convolutional Neural Network Model for Intelligent Discrimination between Coal and Rocks in Coal Mining Face," *Math. Probl. Eng.*, vol. 2020, 2020, doi: 10.1155/2020/2616510.

[53] B. Agarwal, V. E. Balas, L. C. Jain, R. C. Poonia, and M. Sharma, *Deep learning techniques for biomedical and health informatics*. 2020.

[54] C. Chhabra and M. Sharma, "Machine Learning, Deep Learning and Image Processing for Healthcare: A Crux for Detection and Prediction of Disease," in *Lecture Notes on Data Engineering and Communications Technologies*, vol. 91, 2022.

[55] H. Gholamalinezhad and H. Khosravi, "Pooling Methods in Deep Neural Networks, a Review," 2020, [Online]. Available: http://arxiv.org/abs/2009.07485.

[56] X. Huang, Q. Yang, and H. Qiao, "Lightweight Two-Stream Convolutional Neural Network for SAR Target Recognition," *IEEE Geosci. Remote Sens. Lett.*, vol. 18, no.

4, 2021, doi: 10.1109/LGRS.2020.2983718.

[57] S. Albelwi and A. Mahmood, “A framework for designing the architectures of deep Convolutional Neural Networks,” *Entropy*, vol. 19, no. 6, 2017, doi: 10.3390/e19060242.

[58] Y. Wang, Y. Li, Y. Song, and X. Rong, “The influence of the activation function in a convolution neural network model of facial expression recognition,” *Appl. Sci.*, vol. 10, no. 5, 2020, doi: 10.3390/app10051897.

[59] W. H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, and Y. Wei, “Deformable Conv,” *Iccv 2017*, p. 6003, 2017, [Online]. Available: https://github.com/msracver/Deformable-ConvNets.

[60] W. Ketwongsa, S. Boonlue, and U. Kokaew, “A New Deep Learning Model for the Classification of Poisonous and Edible Mushrooms Based on Improved AlexNet Convolutional Neural Network,” *Appl. Sci.*, vol. 12, no. 7, p. 3409, 2022, doi: 10.3390/app12073409.

[61] S. H. S. Basha, S. R. Dubey, V. Pulabaigari, and S. Mukherjee, “Impact of fully connected layers on performance of convolutional neural networks for image classification,” *Neurocomputing*, vol. 378, pp. 112–119, 2020, doi: 10.1016/j.neucom.2019.10.008.

[62] G. Saini, A. Khamparia, and A. K. Luhach, “Classification of plants using convolutional neural network,” *Adv. Intell. Syst. Comput.*, vol. 1045, no. 2, pp. 551–561, 2020, doi: 10.1007/978-981-15-0029-9_44.

[63] C. Nwankpa, W. Ijomah, A. Gachagan, and S. Marshall, “Activation Functions: Comparison of trends in Practice and Research for Deep Learning,” pp. 1–20, 2018, [Online]. Available: http://arxiv.org/abs/1811.03378.

[64] S. H. Chon, “Hyper-parameter Optimization of a Convolutional Neural Network,” p. 91, 2019, [Online]. Available: https://scholar.afit.edu/etd/2297%0A%0A.

[65] R. E. Engineering, J. Hauke, and T. Kossowski, “Ground Truth Data, Content, Metrics, and Analysis What Is Ground Truth Data?,” *Quaest. Geogr.*, vol. 30, no. 2, pp. 1–13, 2012, [Online]. Available: https://www.embedded-vision.com/sites/default/files/apress/computervisionmetrics/chapter7/9781430259299_Ch07.pdf.

[66] V. Andreieva and N. Shvai, “GENERALIZATION OF CROSS-ENTROPY LOSS.”

[67] Q. Wang, Y. Ma, K. Zhao, and Y. Tian, “A Comprehensive Survey of Loss Functions in Machine Learning,” *Ann. Data Sci.*, vol. 9, Apr. 2022, doi: 10.1007/s40745-020-00253-5.

[68] V. Andreieva and N. Shvai, “GENERALIZATION OF CROSS-ENTROPY LOSS.” Vol.519, no. 6, 2020, DOI: https://doi.org/10.18523/2617-7080320203-10.

[69] N. Syafri, Edi; Endrizal, “Deep Learning for Computer Vision with Python,” *J. Chem. Inf. Model.*, vol. 53, no. 9, pp. 1689–1699, 2013.

[70] S. Hochreiter, “The Vanishing Gradient Problem During Learning Recurrent Neural Nets and Problem Solutions,” *Int. J. Uncertainty, Fuzziness Knowledge-Based Syst.*, vol. 6, pp. 107–116, Apr. 1998, doi: 10.1142/S0218488598000094.

[71] N. Syafri, Edi; Endrizal, “Deep Learning for Computer Vision with Python,” *J. Chem. Inf. Model.*, vol. 53, no. 9, pp. 1689–1699, 2013.

[72] J. Gilmer *et al.*, “A Loss Curvature Perspective on Training Instability in Deep Learning,” 2021, [Online]. Available: http://arxiv.org/abs/2110.04369.

[73] L. Zhang, H. J. Gao, J. Zhang, and B. Badami, “Optimization of the Convolutional Neural Networks for Automatic Detection of Skin Cancer,” *Open Med.*, vol. 15, no. 1, pp. 27–37, 2020, doi: 10.1515/med-2020-0006.

[74] G. Lin and W. Shen, “Research on convolutional neural network based on improved Relu piecewise activation function,” *Procedia Comput. Sci.*, vol. 131, pp. 977–984, 2018, doi: 10.1016/j.procs.2018.04.239.

[75] Y. D. Zhang, C. Pan, X. Chen, and F. Wang, “Abnormal breast identification by nine-layer convolutional neural network with parametric rectified linear unit and rank-based stochastic pooling,” *J. Comput. Sci.*, vol. 27, pp. 57–68, 2018, doi: 10.1016/j.jocs.2018.05.005.

[76] D. P. Kingma and J. L. Ba, “Adam: A method for stochastic optimization,” *3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc.*, pp. 1–15, 2015.

[77] Ž. Vujović, “Classification Model Evaluation Metrics,” *Int. J. Adv. Comput. Sci. Appl.*, vol. 12, no. 6, pp. 599–606, 2021, doi: 10.14569/IJACSA.2021.0120670.

[1] P. Hao, S. Chokuwa, X. Xie, F. Wu, J. Wu, and C. Bai, “Skeletal bone age assessments for young children based on regression convolutional neural networks,” *Math. Biosci. Eng.*, vol. 16, no. 6, pp. 6454–6466, 2019, doi: 10.3934/mbe.2019323.

[2] B. D. Lee and M. S. Lee, “Automated Bone Age Assessment Using Artificial Intelligence: The Future of Bone Age Assessment,” *Korean J. Radiol.*, vol. 22, no. 5, pp. 792–800, 2021, doi: 10.3348/kjr.2020.0941.

[3] J. E. Pinsker, M. S. L. Ching, D. P. Chiles, M. Lustik, C. B. Mahnke, and V. J. Rooks, “Automated bone age analysis with lossy image files,” *Mil. Med.*, vol. 182, no. 9, pp. e1769–e1772, 2017, doi: 10.7205/MILMED-D-17-00011.

[4] P. D. C. E.-T. H.-J. S.-H. S.-R. J.-J. Nguyen, “Multiple Inputs Deep Neural Networks

for Bone Age Estimation Using Whole-Body Bone Scintigraphy," *J. Korea Multimed. Soc.*, vol. 22, no. 12, pp. 1376–1384, 2019.

[5] A. Hosny, C. Parmar, J. Quackenbush, L. H. Schwartz, and H. J. W. L. Aerts, "Artificial intelligence in radiology," *Nat. Rev. Cancer*, vol. 18, no. 8, pp. 500–510, 2018, doi: 10.1038/s41568-018-0016-5.

[6] J. Chen, B. Yu, B. Lei, R. Feng, D. Z. Chen, and J. Wu, "Doctor Imitator: A Graph-Based Bone Age Assessment Framework Using Hand Radiographs," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 12266 LNCS, pp. 764–774, 2020, doi: 10.1007/978-3-030-59725-2_74.

[7] E. ÖZGÜL, A. YÜCEL, F. KAYA, S. B. KOCA, and A. ERTEKİN, "Determination of bone age and evaluating the applicability of Greulich-Pyle standards among the Turkish children," *J. Surg. Med.*, vol. 6, no. 1, pp. 9–13, 2022, doi: 10.28982/josam.1021045.

[8] H. Khalid *et al.*, "A comparative systematic literature review on knee bone reports from mri, x-rays and ct scans using deep learning and machine learning methodologies," *Diagnostics*, vol. 10, no. 8, pp. 1–43, 2020, doi: 10.3390/diagnostics10080518.

[9] D. Govender and M. Goodier, "Bone of contention: The applicability of the Greulich-Pyle method for skeletal age assessment in South Africa," *South African J. Radiol.*, vol. 22, no. 1, pp. 1–6, 2018, doi: 10.4102/SAJR.V22I1.1348.

[10] African Development Bank, "Revised Guidelines on Project Completion Report (Pcr) Evaluation Note and Project Performance Evaluation Report (Pper)," vol. 19, pp. 16–41, 2001, [Online]. Available: https://www.afdb.org/fileadmin/uploads/afdb/Documents/Evaluation-Reports-_Shared-With-OPEV_/00157887-EN-2001-REVISED-EVALUATION-GUIDELINES.PDF.

[11] A. Rakhlin, "Diabetic Retinopathy detection through integration of Deep Learning classification framework," *bioRxiv*, p. 225508, 2018, [Online]. Available: https://www.biorxiv.org/content/10.1101/225508v2%0Ahttps://www.biorxiv.org/content/10.1101/225508v2.abstract.

[12] D. B. Larson, M. C. Chen, M. P. Lungren, S. S. Halabi, N. V Stence, and C. P. Langlotz, "Performance of a Deep-learning neural network Model in assessing skeletal Maturity on Pediatric hand radiographs," *Radiology*, vol. 287, no. 1, pp. 313–322, 2018, [Online]. Available: https://doi.org/10.1148/radiol.2017170236.

[13] C. Spampinato, S. Palazzo, D. Giordano, M. Aldinucci, and R. Leonardi, "Deep learning for automated skeletal bone age assessment in X-ray images," *Med. Image Anal.*, vol. 36, no. October, pp. 41–51, 2017, doi: 10.1016/j.media.2016.10.010.

[14] H. Lee *et al.*, "Fully Automated Deep Learning System for Bone Age Assessment," *J.*

Digit. Imaging, vol. 30, no. 4, pp. 427–441, 2017, doi: 10.1007/s10278-017-9955-8.

[15] A. Wibisono, J. Rachmad, and E. Anderson, “Deep Learning and Classic Machine Learning Approach for Automatic Bone Age Assessment,” *2019 4th Asia-Pacific Conf. Intell. Robot Syst.*, pp. 235–240, 2019.

[16] F. Dehghani, A. Karimian, and M. Sirous, “Assessing the Bone Age of Children in an Automatic Manner Newborn to 18 Years Range,” *J. Digit. Imaging*, vol. 33, no. 2, pp. 399–407, 2020, doi: 10.1007/s10278-019-00209-z.

[17] J. Guo *et al.*, “Automated Bone Age Assessment Using Artificial Intelligence: The Future of Bone Age Assessment,” *Korean J. Radiol.*, vol. 22, no. 5, pp. 792–800, 2021, doi: 10.3348/kjr.2020.0941.

[18] N. Mualla, E. H. Houssein, and M. R. Hassan, “Automatic bone age assessment using hand x-ray images,” *J. Theor. Appl. Inf. Technol.*, vol. 98, no. 2, pp. 267–278, 2020.

[19] T. D. Bui, J. J. Lee, and J. Shin, “Incorporated region detection and classification using deep convolutional networks for bone age assessment,” *Artif. Intell. Med.*, vol. 97, no. March, pp. 1–8, 2019, doi: 10.1016/j.artmed.2019.04.005.

[20] P. Hao, “Overlap Classification Mechanism for Skeletal Bone Age Assessment,” pp. 0–6, 2021.

[21] N. S. Rani, M. Chandrajith, B. R. Pushpa, and B. J. B. Nair, “A deep convolutional architectural framework for radiograph image processing at bit plane level for gender & age assessment,” *Comput. Mater. Contin.*, vol. 62, no. 2, pp. 679–694, 2020, doi: 10.32604/cmc.2020.08552.

[22] M. C. Chen, “Automated Bone Age Classification with Deep Neural Networks,” *Stanford Univ. USA,Technical Rep.*, pp. 1–7, 2016, [Online]. Available: http://cs231n.stanford.edu/reports/2016/pdfs/310_Report.pdf.

[23] H. Lee *et al.*, “Fully Automated Deep Learning System for Bone Age Assessment,” *J. Digit. Imaging*, vol. 30, no. 4, pp. 427–441, 2017, doi: 10.1007/s10278-017-9955-8.

[24] B. S. Reddy, D. Sreenivasarao, and S. K. Saheb, “An automated system for identification of skeletal maturity using convolutional neural networks based mechanism,” *Int. J. Innov. Technol. Explor. Eng.*, vol. 8, no. 11, pp. 2221–2227, 2019, doi: 10.35940/ijitee.K2049.0981119.

[25] S. Mutasa, P. D. Chang, C. Ruzal-Shapiro, and R. Ayyala, “MABAL: a Novel Deep-Learning Architecture for Machine-Assisted Bone Age Labeling,” *J. Digit. Imaging*, vol. 31, no. 4, pp. 513–519, 2018, doi: 10.1007/s10278-018-0053-3.

[26] M. Prokop-Piotrkowska *et al.*, “An Ensemble-Based Densely-Connected Deep Learning System for Assessment of Skeletal Maturity,” *Lect. Notes Comput. Sci. (including*

Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 22, no. 5, pp. 1–24, Jan. 2019, doi: 10.3390/diagnostics11050765.

[27] K. Li *et al.*, “Automatic Bone Age Assessment of Adolescents Based on Weakly-Supervised Deep Convolutional Neural Networks,” *IEEE Access*, vol. 9, pp. 120078–120087, 2021, doi: 10.1109/ACCESS.2021.3108219.

[28] M. Prokop-Piotrkowska, K. Marszałek-Dziuba, E. Moszczyńska, M. Szalecki, and E. Jurkiewicz, “Traditional and new methods of bone age assessment-an overview,” *JCRPE J. Clin. Res. Pediatr. Endocrinol.*, vol. 13, no. 3, pp. 251–262, 2021, doi: 10.4274/jcrpe.galenos.2020.2020.0091.

[29] S. H. Tajmir *et al.*, “Artificial intelligence-assisted interpretation of bone age radiographs improves accuracy and decreases variability,” *Skeletal Radiol.*, vol. 48, no. 2, pp. 275–283, 2019, doi: 10.1007/s00256-018-3033-2.

[30] M. A. Zulkifley, N. A. Mohamed, S. R. Abdani, N. A. M. Kamari, A. M. Moubark, and A. A. Ibrahim, “Intelligent bone age assessment: An automated system to detect a bone growth problem using convolutional neural networks with attention mechanism,” *Diagnostics*, vol. 11, no. 5, pp. 1–24, 2021, doi: 10.3390/diagnostics11050765.

[31] J. Guo, J. Zhu, H. Du, and B. Qiu, “A bone age assessment system for real-world X-ray images based on convolutional neural networks,” *Comput. Electr. Eng.*, vol. 81, Jan. 2020, doi: 10.1016/j.compeleceng.2019.106529.

[32] O. Oren, B. J. Gersh, and D. L. Bhatt, “Artificial intelligence in medical imaging: switching from radiographic pathological data to clinically meaningful endpoints,” *Lancet Digit. Heal.*, vol. 2, no. 9, pp. e486–e488, 2020, doi: 10.1016/S2589-7500(20)30160-6.

[33] A. Khan, A. Sohail, U. Zahoora, and A. S. Qureshi, “A survey of the recent architectures of deep convolutional neural networks,” *Artif. Intell. Rev.*, vol. 53, no. 8, pp. 5455–5516, 2020, doi: 10.1007/s10462-020-09825-6.

[34] J. D. Álvarez, J. A. Matias-Guiu, M. N. Cabrera-Martín, J. L. Risco-Martín, and J. L. Ayala, “An application of machine learning with feature selection to improve diagnosis and classification of neurodegenerative disorders,” *BMC Bioinformatics*, vol. 20, no. 1, pp. 1–12, 2019, doi: 10.1186/s12859-019-3027-7.

[35] G. Eraslan, Ž. Avsec, J. Gagneur, and F. J. Theis, “Deep learning: new computational modelling techniques for genomics,” *Nat. Rev. Genet.*, vol. 20, no. 7, pp. 389–403, 2019, doi: 10.1038/s41576-019-0122-6.

[36] V. Sze, Y. H. Chen, T. J. Yang, and J. S. Emer, “Efficient Processing of Deep Neural Networks: A Tutorial and Survey,” *Proc. IEEE*, vol. 105, no. 12, pp. 2295–2329, 2017,

doi: 10.1109/JPROC.2017.2761740.

[37] F. Sultana, A. Sufian, and P. Dutta, “Advancements in image classification using convolutional neural network,” *Proc. - 2018 4th IEEE Int. Conf. Res. Comput. Intell. Commun. Networks, ICRCICN 2018*, pp. 122–129, 2018, doi: 10.1109/ICRCICN.2018.8718718.

[38] Y. Kang, N. Cho, J. Yoon, S. Park, and J. Kim, “Transfer learning of a deep learning model for exploring tourists’ urban image using geotagged photos,” *ISPRS Int. J. Geo-Information*, vol. 10, no. 3, 2021, doi: 10.3390/ijgi10030137.

[39] H. Perez, J. H. M. Tah, and A. Mosavi, “Deep learning for detecting building defects using convolutional neural networks,” *Sensors (Switzerland)*, vol. 19, no. 16, 2019, doi: 10.3390/s19163556.

[40] V. Dumoulin and F. Visin, “A guide to convolution arithmetic for deep learning,” pp. 1–31, 2016, [Online]. Available: http://arxiv.org/abs/1603.07285.

[41] A. Bhandare, M. Bhide, P. Gokhale, and R. Chandavarkar, “Applications of Convolutional Neural Networks,” *Int. J. Comput. Sci. Inf. Technol.*, vol. 7, no. 5, pp. 2206–2215, 2016, [Online]. Available: http://ijcsit.com/docs/Volume 7/vol7issue5/ijcsit20160705014.pdf.

[42] D. X. Zhou, “Universality of deep convolutional neural networks,” *Appl. Comput. Harmon. Anal.*, vol. 48, no. 2, pp. 787–794, 2020, doi: 10.1016/j.acha.2019.06.004.

[43] S. Indolia, A. K. Goswami, S. P. Mishra, and P. Asopa, “Conceptual Understanding of Convolutional Neural Network- A Deep Learning Approach,” *Procedia Comput. Sci.*, vol. 132, pp. 679–688, 2018, doi: 10.1016/j.procs.2018.05.069.

[44] J. Á. Román-Gallego, M. L. Pérez-Delgado, and S. V. San Gregorio, “Convolutional Neural Networks Used to Date Photographs,” *Electron.*, vol. 11, no. 2, 2022, doi: 10.3390/electronics11020227.

[45] Z. Liao *et al.*, “CNN Attention Guidance for Improved Orthopedics Radiographic Fracture Classification,” pp. 1–12.

[46] Y. Lecun, Y. Bengio, and G. Hinton, “Deep learning,” *Nature*, vol. 521, no. 7553, pp. 436–444, 2015, doi: 10.1038/nature14539.

[47] G. Litjens *et al.*, “A survey on deep learning in medical image analysis,” *Med. Image Anal.*, vol. 42, no. December 2012, pp. 60–88, 2017, doi: 10.1016/j.media.2017.07.005.

[48] K. Zaborowicz, B. Biedziak, A. Olszewska, and M. Zaborowicz, “Tooth and bone parameters in the assessment of the chronological age of children and adolescents using neural modelling methods,” *Sensors*, vol. 21, no. 18, pp. 1–18, 2021, doi: 10.3390/s21186008.

[49] S. Chopra, P. Bansal, and P. Bansal, “Journal of Advanced Medical and Dental Sciences Research |Vol. 8|Issue 1|,” *J Adv Med Dent Scie Res*, vol. 8, no. 1, pp. 184–186, 2020, doi: 10.21276/jamdsr.

[50] A. Patil and M. Rane, “Convolutional Neural Networks: An Overview and Its Applications in Pattern Recognition,” *Smart Innov. Syst. Technol.*, vol. 195, pp. 21–30, 2021, doi: 10.1007/978-981-15-7078-0_3.

[51] P. Hao *et al.*, “Radiographs and texts fusion learning based deep networks for skeletal bone age assessment,” *Multimed. Tools Appl.*, vol. 80, no. 11, pp. 16347–16366, 2021, doi: 10.1007/s11042-020-08943-1.

[52] B. W. Ballard, *Hands-On Deep Learning for Images with TensorFlow*. 2018.

[53] L. Si, X. Xiong, Z. Wang, and C. Tan, “A Deep Convolutional Neural Network Model for Intelligent Discrimination between Coal and Rocks in Coal Mining Face,” *Math. Probl. Eng.*, vol. 2020, 2020, doi: 10.1155/2020/2616510.

[54] B. Agarwal, V. E. Balas, L. C. Jain, R. C. Poonia, and M. Sharma, *Deep learning techniques for biomedical and health informatics*. 2020.

[55] C. Chhabra and M. Sharma, “Machine Learning, Deep Learning and Image Processing for Healthcare: A Crux for Detection and Prediction of Disease,” in *Lecture Notes on Data Engineering and Communications Technologies*, vol. 91, 2022.

[56] H. Gholamalinezhad and H. Khosravi, “Pooling Methods in Deep Neural Networks, a Review,” 2020, [Online]. Available: http://arxiv.org/abs/2009.07485.

[57] X. Huang, Q. Yang, and H. Qiao, “Lightweight Two-Stream Convolutional Neural Network for SAR Target Recognition,” *IEEE Geosci. Remote Sens. Lett.*, vol. 18, no. 4, 2021, doi: 10.1109/LGRS.2020.2983718.

[58] S. Albelwi and A. Mahmood, “A framework for designing the architectures of deep Convolutional Neural Networks,” *Entropy*, vol. 19, no. 6, 2017, doi: 10.3390/e19060242.

[59] Y. Wang, Y. Li, Y. Song, and X. Rong, “The influence of the activation function in a convolution neural network model of facial expression recognition,” *Appl. Sci.*, vol. 10, no. 5, 2020, doi: 10.3390/app10051897.

[60] W. H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, and Y. Wei, “Deformable Conv,” *Iccv 2017*, p. 6003, 2017, [Online]. Available: https://github.com/msracver/Deformable-ConvNets.

[61] W. Ketwongsa, S. Boonlue, and U. Kokaew, “A New Deep Learning Model for the Classification of Poisonous and Edible Mushrooms Based on Improved AlexNet Convolutional Neural Network,” *Appl. Sci.*, vol. 12, no. 7, p. 3409, 2022, doi:

10.3390/app12073409.

[62] S. H. S. Basha, S. R. Dubey, V. Pulabaigari, and S. Mukherjee, "Impact of fully connected layers on performance of convolutional neural networks for image classification," *Neurocomputing*, vol. 378, pp. 112–119, 2020, doi: 10.1016/j.neucom.2019.10.008.

[63] G. Saini, A. Khamparia, and A. K. Luhach, "Classification of plants using convolutional neural network," *Adv. Intell. Syst. Comput.*, vol. 1045, no. 2, pp. 551–561, 2020, doi: 10.1007/978-981-15-0029-9_44.

[64] C. Nwankpa, W. Ijomah, A. Gachagan, and S. Marshall, "Activation Functions: Comparison of trends in Practice and Research for Deep Learning," pp. 1–20, 2018, [Online]. Available: http://arxiv.org/abs/1811.03378.

[65] S. H. Chon, "Hyper-parameter Optimization of a Convolutional Neural Network," p. 91, 2019, [Online]. Available: https://scholar.afit.edu/etd/2297%0A%0A.

[66] R. E. Engineering, J. Hauke, and T. Kossowski, "Ground Truth Data, Content, Metrics, and Analysis What Is Ground Truth Data?," *Quaest. Geogr.*, vol. 30, no. 2, pp. 1–13, 2012, [Online]. Available: https://www.embedded-vision.com/sites/default/files/apress/computervisionmetrics/chapter7/9781430259299_Ch07.pdf.

[67] Q. Wang, Y. Ma, K. Zhao, and Y. Tian, "A Comprehensive Survey of Loss Functions in Machine Learning," *Ann. Data Sci.*, vol. 9, Apr. 2022, doi: 10.1007/s40745-020-00253-5.

[68] V. Andreieva and N. Shvai, "GENERALIZATION OF CROSS-ENTROPY LOSS." Vol.519, no. 6, 2020, DOI: https://doi.org/10.18523/2617-7080320203-10.

[68] Q. Wang, Y. Ma, K. Zhao, and Y. Tian, "A Comprehensive Survey of Loss Functions in Machine Learning," *Ann. Data Sci.*, vol. 9, Apr. 2022, doi: 10.1007/s40745-020-00253-5.

[69] L. Boué, "Deep learning for pedestrians: backpropagation in CNNs," pp. 1–44, 2018, [Online]. Available: http://arxiv.org/abs/1811.11987.

[70] S. Hochreiter, "The Vanishing Gradient Problem During Learning Recurrent Neural Nets and Problem Solutions," *Int. J. Uncertainty, Fuzziness Knowledge-Based Syst.*, vol. 6, pp. 107–116, Apr. 1998, doi: 10.1142/S0218488598000094.

[71] N. Syafri, Edi; Endrizal, "Deep Learning for Computer Vision with Python," *J. Chem. Inf. Model.*, vol. 53, no. 9, pp. 1689–1699, 2013.

[72] M. Rezaei, "Carpal Bone Analysis using Geometric and Deep Learning Models," 2020.

[73] J. Gilmer *et al.*, "A Loss Curvature Perspective on Training Instability in Deep

Learning," 2021, [Online]. Available: http://arxiv.org/abs/2110.04369.

[74] L. Zhang, H. J. Gao, J. Zhang, and B. Badami, "Optimization of the Convolutional Neural Networks for Automatic Detection of Skin Cancer," *Open Med.*, vol. 15, no. 1, pp. 27–37, 2020, doi: 10.1515/med-2020-0006.

[75] G. Lin and W. Shen, "Research on convolutional neural network based on improved Relu piecewise activation function," *Procedia Comput. Sci.*, vol. 131, pp. 977–984, 2018, doi: 10.1016/j.procs.2018.04.239.

[76] Y. D. Zhang, C. Pan, X. Chen, and F. Wang, "Abnormal breast identification by nine-layer convolutional neural network with parametric rectified linear unit and rank-based stochastic pooling," *J. Comput. Sci.*, vol. 27, pp. 57–68, 2018, doi: 10.1016/j.jocs.2018.05.005.

[77] D. P. Kingma and J. L. Ba, "Adam: A method for stochastic optimization," *3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc.*, pp. 1–15, 2015.

[78] S. Joseph, "Australian Literary Journalism and 'Missing Voices': How Helen Garner finally resolves this recurring ethical tension," Journal. Pract., vol. 10, no. 6, pp. 730–743, 2016, doi: 10.1080/17512786.2015.1058180

[79] T. Bezdan and N. Bačanin Džakula, "Convolutional Neural Network Layers and Architectures," no. July, pp. 445–451, 2019, doi: 10.15308/sinteza-2019-445-451

[80] Ž. Vujović, "Classification Model Evaluation Metrics," *Int. J. Adv. Comput. Sci. Appl.*, vol. 12, no. 6, pp. 599–606, 2021, doi: 10.14569/IJACSA.2021.0120670.

[81] T. T. J. Kiran, "Computer Vision Accuracy Analysis with Deep Learning Model Using TensorFlow," *SSRN Electron. J.*, pp. 319–325, 2020, doi: 10.2139/ssrn.3673214.

[82] K. M. Ting, "Confusion Matrix," *Encycl. Mach. Learn. Data Min.*, no. October, pp. 260–260, 2017, doi: 10.1007/978-1-4899-7687-1_50.

الخلاصة

التطورات السريعة في التكنولوجيا والذكاء الاصطناعي أدت إلى استخدام الأنظمة الآلية في مجموعة متنوعة من القطاعات ، بما في ذلك الطب .أحد التطبيقات هو التقييم الآلي لعمر العظام المستخدم في الطب الشرعي .يمكن استخدام تحديد عمر العظام لتحديد اضطرابات الغدد الصماء والعديد من ألامراض أو عمر العظام.بالاعتمادعلى صور الاشعة السينية لليد اليسرى. يمكن تحديد عمر عظام اليد للطفل من خلال تتبع التغييرات في أنماط نمو عظم اليد ، والتي يمكن القيام بها حتى يكتمل نضج الطفل بعمر 18 سنة .مع تقدم الطفل في السن تحدث تغيرات كبيرة حتى سن البلوغ ، وعند هذه النقطة قد تظهر فجوة كبيرة بين عمر العظام والعمر الزمني وهذا يشير إلى وجود مشكلة في صحة الطفل.

في هذا العمل تم بناء نموذج لتقييم الفئة العمرية للأشخاص بناءً تقنيات التعلم العميق ، ولتحقيق هذا الهدف بما يتلائم مع مكونات الكمبيوتر تم بناء النظام المقترح ،النظام المقترح تضمن ثلاث خطوات رئيسية :معالجة البيانات ، استخراج الفيجرات ، والتصنيف للفئات العمرية كمخرجات للنموذج .

عظام اليد اليسرى للأطفال من عمر سنة حتى سن الثامنة عشر عامًا بالاعتماد على صور الأشعة السينية لليد اليسرى (RSNA) التي تحتوي على 12611 صورة حيث استخدمت كبيانات في هذا العمل مقسمة إلى خمس فئات عمرية اعتماداً على تقارب الفيجرات في كل فئة عمرية. كانت الدقة التي حصلنا عليها لكل فئة (97٪ ، 95٪ ، 96٪ ، 98٪ ، 95٪) على التوالي ، بينما كانت الدقة الكلية 97.22٪.

جامعة كربلاء
كلية علوم الحاسوب وتكنولوجيا المعلومات
قسم علوم الحاسوب

تقييم عمر العظام بأستخدام معمارية شبكة عصبية عميقة جديدة

رسالة ماجستير
مقدمة الى مجلس كلية علوم الحاسوب وتكنولوجيا المعلومات /جامعة كربلاء وهي جزء من متطلبات نيل درجة الماجستير في علوم الحاسوب

كتبت بواسطة

الاء جمال جبار كاظم

بإشراف

الأستاذ المساعد الدكتورة أشوان أنور عبدالمنعم

1444 هـ **2023 م**