MULTI-CLASS CLASSIFICATION AND DETECTION OF KNEE OSTEOARTHRITIS FROM X-RAYS

A Project Report

Submitted by

C RISHI VARDHAN REDDY	**(CB.EN.U4ECE21008)**
D MOHAMMAD SHAAHID	**(CB.EN.U4ECE21011)**
DULAM SANTHOSH	**(CB.EN.U4ECE21012)**
NUKALA SUMANTH	**(CB.EN.U4ECE21033)**

in partial fulfillment of the requirements for the award of

the degree of

Bachelor of Technology

in

Electronics and Communication Engineering

under the supervision of

Dr. Devi Vijayan

AMRITA
VISHWA VIDYAPEETHAM
DEEMED TO BE UNIVERSITY UNDER SECTION 3 OF UGC ACT, 1956

Department of Electronics and Communication Engineering,
Amrita School of Engineering,
Amrita Vishwa Vidyapeetham,
Coimbatore – 641112

May 2025

AMRITA VISHWA VIDYAPEETHAM
DEEMED TO BE UNIVERSITY UNDER SECTION 3 OF UGC ACT, 1956

Dept of Elec. & Communication Engineering,
Amrita School of Engineering,
Amrita Vishwa Vidyapeetham,
Coimbatore – 641112

Certificate

This is to certify that this project report titled **"Multi-Class Classification and Detection of Knee Osteoarthritis From X-rays"**, submitted in partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology in Electronics and Communication Engineering**, by **Mr. C Rishi Vardhan Reddy, Mr. D Mohammad Shaahid, Mr. Dulam Santhosh, Mr. Nukala Sumanth** is a bonafide record of work carried out by them, under my supervision and that it has not been submitted, to the best of my knowledge, in part or in full, for the award of any other degree or diploma.

Dr. Devi Vijayan
(Advisor)
Department of Electronics and
Communication Engineering,
Amrita School of Engg.,
Coimbatore

Dr. Madhu Mohan N
Chair
Department of Electronics and
Communication Engineering,
Amrita School of Engg.,
Coimbatore

Date:

This project was evaluated by us on ----------------------------------.

Internal Examiner **External Examiner**

Declaration

We do hereby declare that this project report titled **“Multi-Class Classification and Detection of Knee Osteoarthritis From X-rays”,** submitted in partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology in Electronics and Communication Engineering**, is a true record of work carried out by us, under the supervision of **Dr. Devi Vijayan** and that all information contained herein, which do not arise directly from my work, have been properly acknowledged and cited, using acceptable international standards. Further, we declare that the contents of this report have not been submitted, in part or in full, for the award of any other degree or diploma.

Coimbatore – 641112
Date:

C RISHI VARDHAN REDDY
D MOHAMMAD SHAAHID
DULAM SANTHOSH
NUKALA SUMANTH

We dedicate this project to everyone who has stood by us throughout the journey of this project, offering their valuable support and guidance. Our gratitude extends to the entire faculty and staff of the Department of Electronics and Communication Engineering, as well as our esteemed mentor, Dr. Devi Vijayan, whose wisdom and guidance has been beneficial to the success of the project. We also dedicate the project to our family, friends and well-wishers for their continued encouragement and beliefs in us.

Acknowledgments

We would like to extend our heartful gratitude to Amrita Vishwa Vidyapeetham for giving ample academic opportunities and a stable environment to pursue our education. We are grateful, both to the faculty and the non-teaching staff of Department of Electronics and Communication Engineering of the college for their enduring support and guidance throughout the duration of our project.

We would like to thank our project advisor, Assistant Professor (SG) Dr. Devi Vijayan, for her valuable and much-needed guidance and support, without which we would not be able to carry forward our project with firm and determination.

We are forever grateful to our parents and fellow peers for their everlasting backing, encouragement and beliefs. Their presence provided us the inspiration to gracefully complete our project with pomp and splendor.

Abstract

Osteoarthritis of the knee happens when cartilage in your knee joint breaks down. When this happens, the bones in your knee joint rub together, causing friction that makes your knees hurt, become stiff or swell. Osteoarthritis in the knee can’t be cured. So, early detection of Knee Osteoarthritis (KOA) will help patients to get treatments that can relieve symptoms and slow their condition’s progress. While CT scans and MRIs offer superior diagnostic detail for knee osteoarthritis, X-rays remain the most cost-effective option, making improvements in their accuracy highly beneficial. The Kellgren-Lawrence (KL) grading system grades knee osteoarthritis severity based on X-ray features, like joint space narrowing and osteophyte formation, categorizing OA into five grades.

The proposed work aims in developing a computer aided diagnosis system by implementing single-level and multi-level classification systems for identifying Kellgren–Lawrence (KL) grades from knee X-ray images. In single level approach (KL grade 0) we train and test all five grade images from different convolutional neural network (CNN) individually and classify accordingly. Under the single-level setting, five models of CNN were assessed, with the top performance of 71.26% from ConvNeXt. Further, we constructed a multi-level approach in which we classify whether the image comes under normal or abnormal category and then proceed accordingly. HRNet and ConvNeXt were used for multi-level setting approach and ConvNeXt outperformed HRNet in both levels of classification. The binary-level model, for classifying normal from abnormal achieved a maximum of 80.6% accuracy and abnormal an accuracy of 73.94% thus proving the strengths of a hierarchical system.

Table of Contents

List of Abbreviations

AI	Artificial Intelligence
CNN	Convolutional Neural Network
Da-ViT	Dual Attention Vision Transformer
DC-AAE	Dual-Channel Adversarial Autoencoders
DL	Deep Learning
FN	False Negative
FP	False Positive
GCViT	Global Context Vision Transformer
GELU	Gaussian Error Linear Unit
HRNet	High-Resolution Net
ILSVRC	ImageNet Large Scale Visual Recognition Challenge
KOA	Knee Osteoarthritis
KL	Kellgren and Lawrence
MAE	Mean Absolute Error
MaxViT	Multi-Axis Vision Transformer
MOST	Multicenter Osteoarthritis Study
OAI	Osteoarthritis Initiative
PIM	Plug-In Modules
ReLU	Rectified Linear Unit
ResNet	Residual Networks
SE	Squeeze-and-Excitation
TN	True Negative
TP	True Positive

VGG	Visual Geometry Group
ViT	Vision Transformers
VS Code	Visual Studio Code
YOLO	You Only Look Once

List of Symbols

σ	Activation function (ReLU)
$*$	Convolution operation
α	Scaling factors for depth
β	Scaling factors for width
γ	Scaling factors for resolution
ϕ	Compound scaling coefficient
Σ	Summation
ψ	Transformation function that upsamples or downsamples the feature map
ϵ	Constant
μ	Mean

List of Tables

List of Figures

1. Introduction

1.1 Background

Knee osteoarthritis is a degenerative state where there is progressive loss of cartilage leading to knee pain, stiffness, and limitation of movement. Breakdown of the protective cartilage increases friction and causes inflammation in the joint; it has a severely disabling effect on quality of life and the patient's ability to carry out everyday tasks of life. Common risk factors for knee osteoarthritis are advanced age, obesity, injuries, or genetic predisposition. Another major cause is mechanical load on the knee joint; knees bearing the burden of lengthy hours of standing at work, such as in the cases of teachers, health care workers, and retail sales persons, suffer additional strain. Prolonged hours of standing cause repeated stress and accelerate cartilage degeneration; this increases the risk factor for OA. The importance of recognizing the potential risks to knee health from these work-related exposures is that prevention and promotion of joint wellness would be made possible.

These are mainly elderly patients aged above 40 years, and the prevalence is estimated at 28-30% [1]. This results from continuous wear of the articular cartilage that leads to the knee bones rubbing one against the other. Several factors affect this erosion, including genetic predisposition, previous trauma, and overuse. It is dissimilar from inflammatory arthritis, which tends to worsen with activity; knee OA pain typically worsens with movement and can lead to joint instability, deformity, and reduced functionality as cartilage loss progresses and narrows the space between the knee joints.

Gender is one of the key risk factors for knee osteoarthritis. The overwhelming majority of people affected are women compared with men. The disparity is usually very evident in postmenopausal women, when hormonal fluctuations cause a decrease in bone density and an increase in body fat. About 60-70% of patients diagnosed with knee OA are women [2], and therefore gender-specific methods of preventing and treating the condition urgently require evidence. The Kellgren-Lawrence (KL) grading scale is the most commonly used for OA severity assessment: it grades joint degeneration by radiographic evidence such as features including narrowing of joint space, bone spurs, and deformity.

- Grade 0 (No OA): No definite radiographic features of osteoarthritis are present. The joint is normal with no evidence of degeneration.
- Grade 1 (Doubtful OA): This is the initial phase of OA where only minimal joint space narrowing and potentially small osteophytes are seen at the location of bone spurs. The alterations are minor and likely not to be symptomatic
- Grade 2 (Minimal OA): Osteophytes are present, and definite but minimal joint space narrowing is noted. This is the first stage in which OA can be confidently diagnosed on plain radiography. There may be mild pain.
- Grade 3 (Moderate OA): Significant joint space narrowing, moderate osteophytes, possible sclerosis of subchondral bone, and early stage of bone deformity. The pain and stiffness are more marked in this stage.
- Grade 4 (Severe OA): The most severe stage of the disease, with marked narrowing of the joint space, large osteophytes, marked subchondral sclerosis, and prominent bone deformity. The patients have chronic pain and restricted joint function.

1.2 Motivation

The motivation to adopt knee osteoarthritis as the project stems from the fact that it significantly influences global health, millions of people have been affected worldwide. It is also on the list of principal diseases that cause disability, especially in older generations. On the other hand, the knee is one of the most commonly affected joints. This vicious cycle evolves into a progressive degeneration of cartilage, and it leads to pain, stiffness, and the inability to carry out movements. All these continue to inhibit normal daily life and quality of life. Due to an increase in the population that is elderly in age, cases of knee OA are on the rise; therefore, it has become a highly critical field of research in medical imaging and diagnostics.

It's challenging to notice early knee OA as changes are usually subtle; even minor levels of joint space narrowing and osteophyte formation may occur more gradually. Due to this complexity, knee OA has recently become the subject of much interest in current literature regarding new, sophisticated image classification approaches, including deep learning, as a potential route in enhancing diagnostic accuracy to allow earlier diagnosis. It is estimated that there are numerous risk factors for OA due to excess weight [3], which imparts increased mechanical loading of the knee and, in fact, for every kilogram of weight gained, the incidence of OA increases, though again this risk is more pronounced in predisposed individuals. Indian burgeoning rates of obesity stress proper weight management techniques, including diet and exercise, with resultant decreased risk for OA. With a focus on innovative diagnostic tools and lifestyle modifications applied proactively, we may be able to increase patient outcomes and help patients who experience the condition of knee osteoarthritis to experience quality life.

1.3 Overview of the report

The overall report is composed of four sections. Section 1 gives an introductory account of knee osteoarthritis and provides a rationale for conducting the research, with importance drawn to understanding the condition. Section 2 is dedicated to a literature survey, which reviews CNN models related to KOA, informing the current states and practices in the field. Section 3 describes the methodology used to carry out the research and the approach that led to solving the problem identified, along with the techniques undertaken for the purposes of analysis. Finally, Section 4 is the results and discussion whereby its analysis and interpretation of findings within current literature in relation to the broader conversations around KOA and its implications for patients and healthcare providers.

2. Related Work

This section reviews the studies for the detection of knee osteoarthritis with the application of deep learning models. Such models use CNN architectures along with techniques for feature selection in finding the important features while others enhance the design of the machine learning and deep learning models for better accuracy.

2.1 Literature Survey

Farooq et al., [4] employed a dual-channel adversarial autoencoders (DC-AAE)) to classify knee OA severity. The approach builds over a multitask learning framework where the tasks are supervised and unsupervised. The architecture was applied to radiographs and trained on large datasets such as OAI and MOST. The model achieved an accuracy of 75.53%, precision of 74.10%, recall of 79.69%. The study highlights the potential of AI in medical image analysis, showing that deep learning models can assist in KOA severity classification without requiring extensive labelled data.

Wang et al., [5] introduced a refined deep learning approach that described the quantification of knee osteoarthritis with the 2 help of X-ray images. In the proposed system, a two-stage approach was utilized. Here, high-confidence sample learning was implemented using ResNet-34 architecture for extracting the features. This model classified the X-ray images into five grades of osteoarthritis and achieved accuracy of 70.13%, precision of 71.2%, recall of 69.5%, and F1 score of 70.4%.

Jain et al., [6] introduced a deep learning architecture OsteoHRNet, for the estimation of the severity of knee OA from X-ray images. The proposed HRNet can learn highly de tailed multi-scale feature representation of the knee joint. The attention mechanism has been deployed in this architecture so as to enhance the prediction accuracy. The model assessed using the OAI dataset, it achieved an accuracy of 71.74% and a mean absolute error (MAE) of 0.311.

Chen et al., [7] described an entirely automated approach for knee osteoarthritis grading severity in applying deep neural networks and ordinal loss. The model applied YOLOv2 for detection of the knee joint, followed by the fine-tuning of the CNN models, namely VGG-19, using an ordinal loss, which is trainable. The model approach classified knee joints at five ordinal stages in achieving an accuracy of 69.6%, precision of 70.4%, recall of 67.5%, and F1 score of 68.3%.

Sakib Apon et al., [8] report a comparative study of conventional machine learning, CNNs, and Vision Transformer models for detection of knee osteoarthritis severity from X-ray images. The study utilizes a multi-center OAI dataset and sophisticated preprocessing to improve diagnostic features. While CNN methods obtain moderate accuracy (65%), Vision Transformers (e.g., Da-VIT, GCVIT, Max ViT) show better performance with accuracies of about 66% and AUC scores above 0.83. These results highlight the potential of transformer-based techniques for accurate, robotic osteoarthritis evaluation in clinical use.

Lee et al., [9] proposed a plug-in deep learning model, utilizing an X-ray image to classify the severity of KOA through fine grained classification by using Plug-in Modules (PIM). The model was trained on the Osteoarthritis dataset, MOST and employed an ensemble of Swin and EfficientNet achieving an accuracy of 75.7%. The accuracy varied by grade level, ranging from 43% for KL1 to 96% for KL4.

Aslan et al., [10] suggested a hybrid deep learning model for automatic detection of knee osteoarthritis using knee X-ray images. It used three types of CNN architectures namely DenseNet201, DarkNet53, and ShuffleNet for feature extraction, followed by Nearest Component Analysis (NCA) to feature selection, and classified KOA in five grades based on the Kellgren -Lawrence grading method which resulted in an accuracy of 84.12%, precision of 87.3%, recall of 85.4% and an F1-score of 86.3%.

Patil et al., [11] proposed Densely Connected Fully Convolutional Network, DFCN, in order to classify and predict the risk of osteoarthritis of the knee through X-ray images. For the spatial features extraction process and classification of knee osteoarthritis into its five-stage classification, they employed a DFCN-based architecture. The model produced excellent performance, 94% accuracy, 94.5% precision, 93.2% recall, and an F1 score of 93.8% on the test set.

Fatema et al., [12] presented an automated optimal distance feature-based decision system for the diagnosis of knee osteoarthritis using segmented X-ray images. In this work, the XGBoost (XGB) technique has been used for extracting and classifying six distance features acquired from the segmented regions of interest that classify images as belonging to one of five severity classes: normal, doubtful, minimal, moderate, or severe. The model is, thus, capable of achieving an accuracy of 99.46%, precision of 99.25%, recall of 99.43%, and F1-score of 99.1%. According to their approach, superior classification performance will be guaranteed through the proposed six optimal features.

Teoh et al., [13] proposed a novel method of deep hybrid learning to classify features of knee OA from radiographic images. It used 16 CNN structures for feature extractions with the support of ML classifiers in nine OA features, namely Kellgren-Lawrence grades, osteophytes, joint-space narrowing, and pain intensity. It stood well on these types of models with an accuracy of 92.53%, F1 score of 0.93, and MSE of 0.18 on the basis of KL grade prediction.

Chandra et al., [14] proposed an optimized feature selection-deep learning model for the recognition and severity evaluation of knee osteoarthritis, using X-ray images. The model used CNN for the feature extraction process and leveraged GBC and PSO together to optimize the feature set. The developed model classified the KOA severities into five different grades, which achieved 98.91% accuracy, 98.90% specificity, 98.91% sensitivity, and 99.13% PPV.

Wang et al., [15] introduced a refined deep learning approach that described the quantification of knee osteoarthritis with the help of X-ray images. In the proposed system, a two-stage approach was utilized. Here, high-confidence sample learning was implemented using ResNet-34 architecture for extracting the features. This model classified the X-ray images into five grades of osteoarthritis. The model provided average accuracy to be around 70.13% along with precision being 71.2%, and it also had recall value at 69.5% while the F1 score was 70.4%. This approach gave better results, especially for the early-stage classification, compared to other state-of-the-art methods.

3. Methodology

In this study, we present two complementary methods of knee osteoarthritis classification based on the Kellgren–Lawrence (KL) grading system. The first approach is a single-level model classification that has been trained on all five KL grades (0–4), and the second method employs two separate models intended for hierarchical tasks: a binary classification model differentiating normal (KL-0) from abnormal (KL 1-4) cases, followed by an abnormal Classification model differentiating between abnormal cases. The methodologies are shown graphically in Fig. 3.1 and Fig. 3.8.

3.1 Single-Level Classification Model

The baseline method involves a dataset classification of knee X-ray images into one of five KL grades, where grade 0 represents a normal knee and grades 1-4 represent increasing levels of osteoarthritic changes. To analyse this, we employed the following CNN models VGG16 [16], ResNet50 [17], EfficientNetB0, HRNet, and Convent. The overall distribution of the dataset utilized for training and testing is listed in Table 4-1. The architecture of the model is represented in Fig. 3.1 each CNN model is pre-trained on a large dataset and fine-tuned on the osteoarthritis dataset. The key hyperparameters include learning rate, epochs, batch size, and optimizer, are chosen to ensure convergence and robust performance. The performance of each of these individual models, which was measured by accuracy, precision, recall, f1 score, classification report, and confusion matrix, will be used as the reference point for hierarchical classification approach.

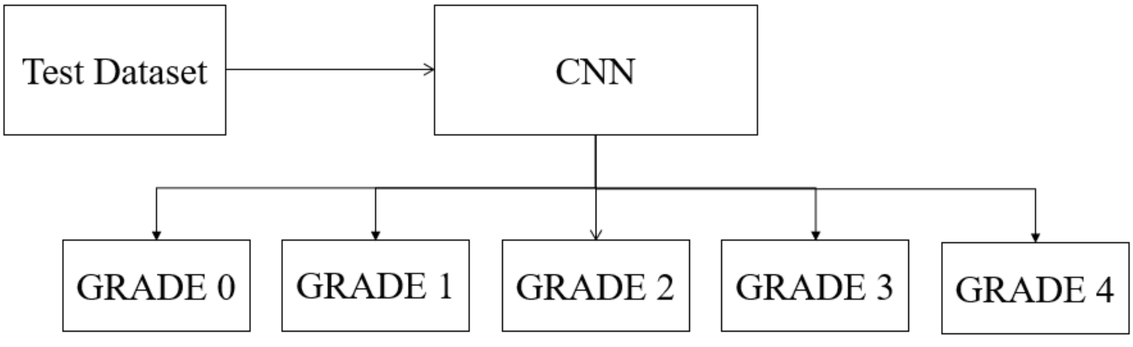


Fig. 3.1 Multi-Class Single-Level

Let’s go through the CNN models we used:

3.1.1 VGG16

VGG16 is the deep convolutional neural network that was first presented by the Visual Geometry Group at Oxford University. This model has 16 layers, with some as convolutions and max-pooling layers along with fully connected ones. What makes the feature of VGG16 the most relevant is the usage of 3x3 convolutional filters, maintaining a simple architecture yet powerful in effect. This approach allows the network to learn complex features and yet does not increase the computational requirements unmanageably. The very best performance in ILSVRC 2014 in the ImageNet Large Scale Visual Recognition Challenge was achieved by VGG16. It became foundational within computer vision.

The generalization power of VGG16 over variable datasets makes it the first choice for transfer learning. With pre-trained weights for large data sets like ImageNet, it can perform feature extraction on small data sets. Its architecture is pretty straightforward to implement and adapt to particular tasks, even though its depth is quite extensive [16]. This balance of simplicity, depth, and performance made VGG16 extremely popular in the deep.

VGG-16 uses simple convolutional blocks stacked sequentially. The basic convolution operation in VGG-16 can be expressed as:

$$y = \sigma(W * x + b) \tag{3.1}$$

Where:

- W = Weight matrix of the convolution filter.
- x = Input feature map.
- b = Bias term.
- $*$ = Convolution operation.
- σ = Non-linear activation function (ReLU).

The formula represents a typical convolution operation followed by an activation function.

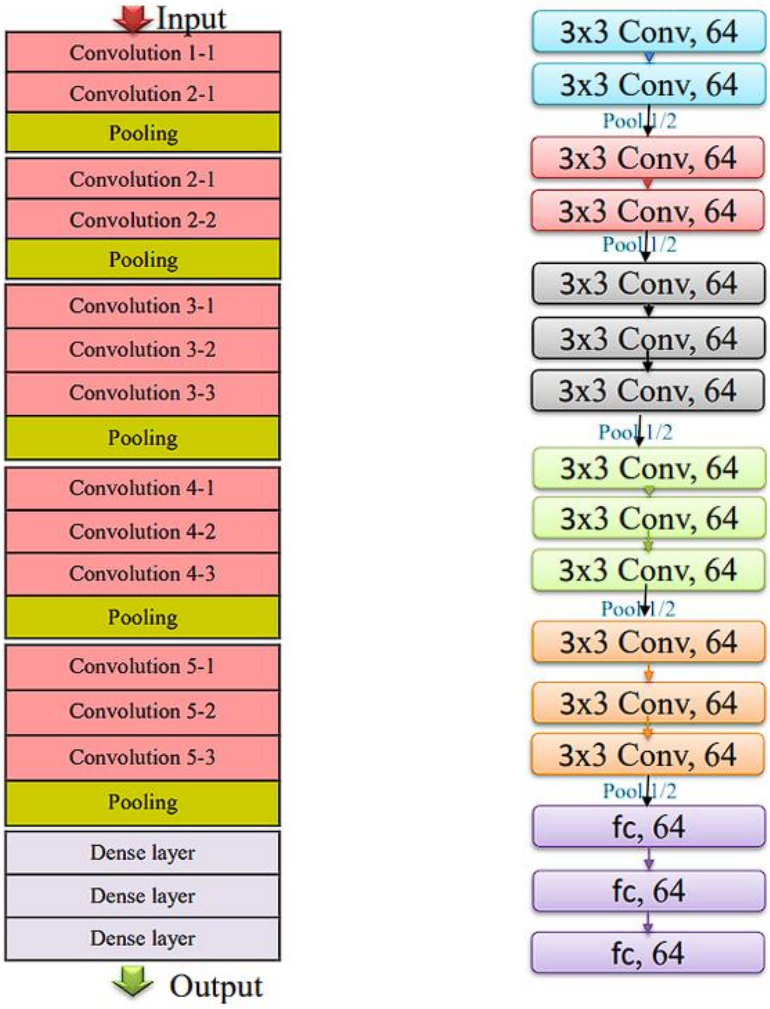


Fig. 3.2 Block Diagram of VGG16

3.1.2 ResNet50

This is a deep convolutional neural network that overcomes the vanishing gradient in deep learning. Here, residual connections are introduced in such a way that it bypasses one or more layers while training. Instead of learning directly the mappings, residuals, which is a result of input minus output, and makes easier to optimize for deeper networks [17]. The architecture of ResNet50 is detailed in:

1. Convolutional Layers: Extract feature from the input image; the first layer applies 7x7 convolution followed by max-pooling down sampling.

2. Residual Blocks: Consist of two or three convolutional layers with skip connections that add the input directly to the output to enhance the gradient flow.

3. Batch Normalization & ReLU: Used after every convolution for normalization and for non-linearity.

4. Bottle-neck Architecture: A1x1, 3x3, and 1x1 convolutional structure to support efficiency.

5. Global Average Pooling: Transforms feature maps into a single feature vector just before classification.

$$y = \mathrm{F}(x, \{W_i\}) + x \tag{3.2}$$

Where:

- x = Input to the residual block.
- $\mathrm{F}(x, \{W_i\})$ = Residual function (typically a stack of convolutional layers).
- y = Output after adding the residual to the input.
- $\{W_i\}$ = Weights of the convolutional layers.

The formula shows that the output (y) is the sum of the input (x) and the output of the convolutional operation (F).

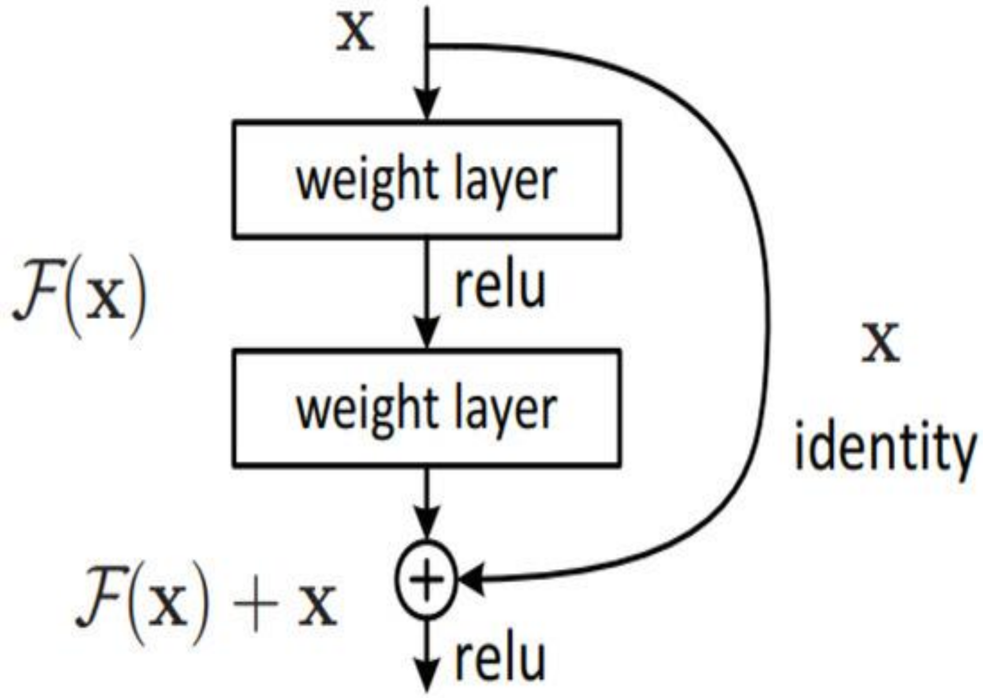

x
weight layer
$\mathcal{F}(\mathbf{x})$
relu
x
weight layer
identity
$\mathcal{F}(\mathbf{x}) + \mathbf{x}$
relu

Fig. 3.3 Skip Connection Architecture in ResNet50

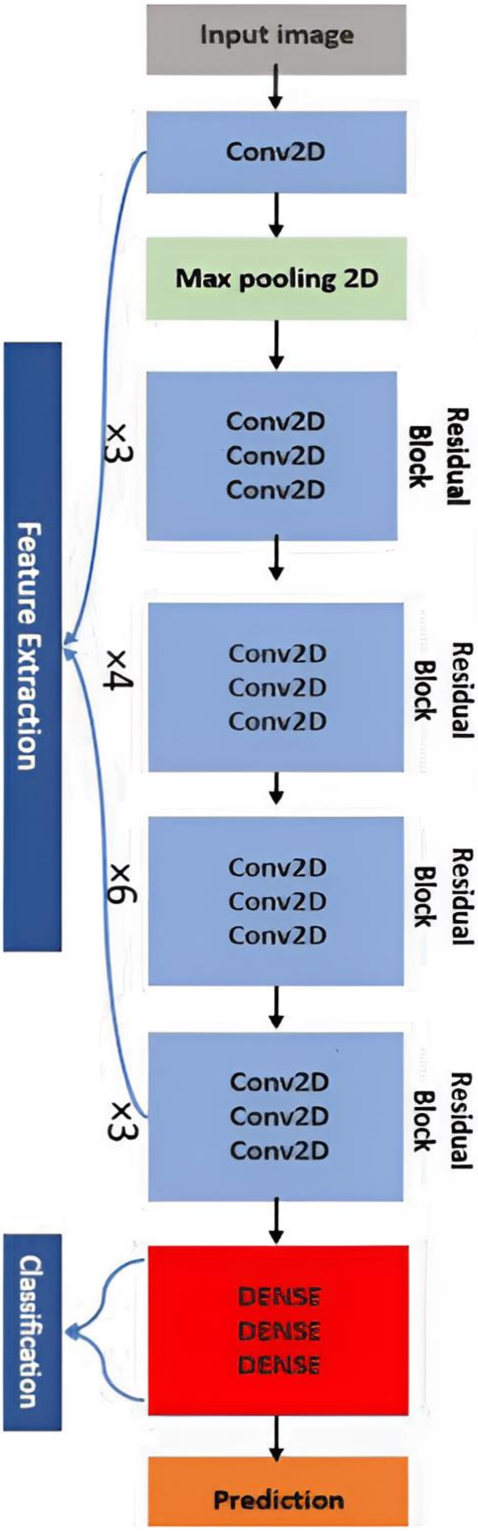


Fig. 3.4 Block Diagram of ResNet50

3.1.3 EfficientNetV2B0

Google AI researchers, proposed EfficientNet in 2019. A paper "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks" by Mingxing Tan and Quoc V. Le introduced it. The model scaling presented here by EfficientNet remained the main innovation based on compound scaling in order to gain the proper balance between depth, width, and resolution of the network, and as a result, is efficient in terms of both accuracy and computational cost.

The efficient parameters in the model make it possible to reach state-of-the-art accuracy; at the same time, they are rather more computationally efficient than architectures like ResNet and Inception. Thus, models of EfficientNet gained extensive usage in the majority of computer vision tasks, such as image classification, object detection, and medical image analysis.

This EfficientNet family ranges from B0 to B7 and comes with larger scale, increased complexity and parameter sizes for better performance or task-specific. These factors set scales, or scaling coefficients (B0 to B7), dictate what are essentially model characteristics, with B0 as smallest and B7 as biggest. The size index represents a compromise between model-size and performance.

The use is a balanced choice that entails efficiency while keeping effectiveness. In terms of image classification, the Efficientnet-B0 applies compound scaling. It optimizes the trade-offs with respect to complexity and efficiency. SE blocks boost the process of recalibration for good performance. Competitive accuracy and minimal demand for computations make Efficientnet-B0 a standard baseline in the vision community, offering a reference on how to balance the delicate equilibrium between efficiency and effectiveness. Model Size and Accuracy Relationship in Resource-Constrained Scenarios.

$$depth = \alpha^{\emptyset} , width = \beta^{\emptyset} , reszolution = \gamma^{\emptyset} \tag{3.3}$$

Where:

- ϕ = Compound scaling coefficient.
- α, β, γ = Scaling factors for depth, width, and resolution respectively.

The formula balances model width, depth, and resolution to achieve better performance with efficiency.

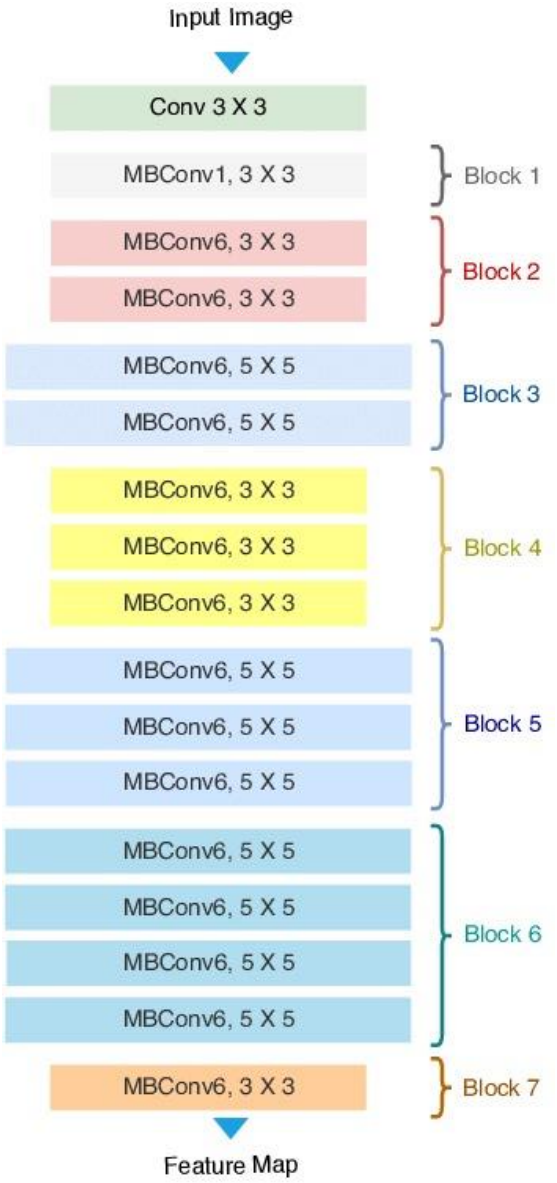


Fig. 3.5 Block Diagram of EfficientNet

3.1.4 HRNet

HRNet (High-Resolution Network) is a deep learning architecture specifically designed to handle computer vision tasks with the need for high-resolution representations like human pose estimation, image classification, and segmentation. HRNet was introduced by researchers headed by Jingdong Wang of Microsoft Research Asia in the year 2019. HRNet architecture appeared in the paper "Deep High-Resolution Representation Learning for Visual Recognition.".

HRNet is meant to preserve high-resolution feature representations across the whole network instead of reducing the resolution significantly and then trying to upsample it afterwards. This is different from traditional architectures such as ResNet, which downsample the resolution early and recover it only partially in the latter stages. The major innovation of HRNet is its parallel multi-resolution subnetworks that continuously exchange information. Rather than progressively downsampling, HRNet begins with a high-resolution stream and adds low-resolution streams incrementally in parallel. The parallel streams are linked by fusion layers that merge features across different resolutions, enabling rich multi-scale contextual information.

HRNet architecture has multiple stages. The Stem Network contains early convolutional layers for low-level feature extraction. The High-Resolution Stage preserves high-resolution representation. The Multi-Resolution Stages incorporate lower-resolution branches while persistently combining features from various branches. Fusion Layers are utilized to combine multi-resolution features by upsampling and downsampling appropriately. Lastly, the Prediction Head combines the combined features to produce the output. This architecture enables HRNet to preserve fine spatial details while efficiently capturing multi-scale contextual information.

HRNet has been successfully applied in knee osteoarthritis (OA) grading to preserve high-resolution spatial details while extracting deep features from X-ray images. The network effectively classifies the Kellgren-Lawrence (KL) grade of knee OA by detecting subtle joint changes that low-resolution networks may not capture. Because it can preserve high-resolution details and fuse multi-scale information, HRNet is very appropriate for knee OA classification in which small yet valuable changes in joint spaces are important for proper grading.

HRNet uses a multi-resolution fusion strategy to combine features from different resolutions. The basic formula for multi-resolution fusion in HRNet is as follows:

$$y^r = \sum_{s=1}^{S} \psi^{r \to s}(x^s) \tag{3.4}$$

Where:

- y^r= Output feature map at resolution r.
- x^s= Input feature map at resolution s.
- S = Total number of parallel streams (resolutions).
- $\psi^{r \to s}$ = Transformation function that upsamples or downsamples the feature map from resolution s to r.

HRNet maintains high-resolution representations by gradually adding low-resolution branches while preserving the high-resolution stream. The representation at each stage can be formulated as:

$$h_{k+1} = f_k(h_k) \tag{3.5}$$

Where:

- h_k = High-resolution representation at stage k.
- f_k = Transformation function (a series of convolutional operations).
- This formula indicates that the next stage builds on the previous high-resolution representation.

HRNet utilizes bottleneck blocks to reduce computational complexity while preserving spatial information:

$$y = x + \sigma\big(W_2 \cdot \sigma(W_1 \cdot x)\big) \tag{3.6}$$

Where:

- x = Input feature map.
- W_1, W_2 = Convolutional weight matrices.
- σ = Activation function (usually ReLU).
- This residual structure helps in efficient feature learning while maintaining resolution.

In classification tasks, the final prediction uses the SoftMax function:

$$P(y = k \mid x) = \frac{e^{z_k}}{\sum_j e^{z_j}} \tag{3.7}$$

Where:

- Z_k = Logit for class k obtained from the final layer.
- The SoftMax function normalizes these logits into probabilities across all output classes.

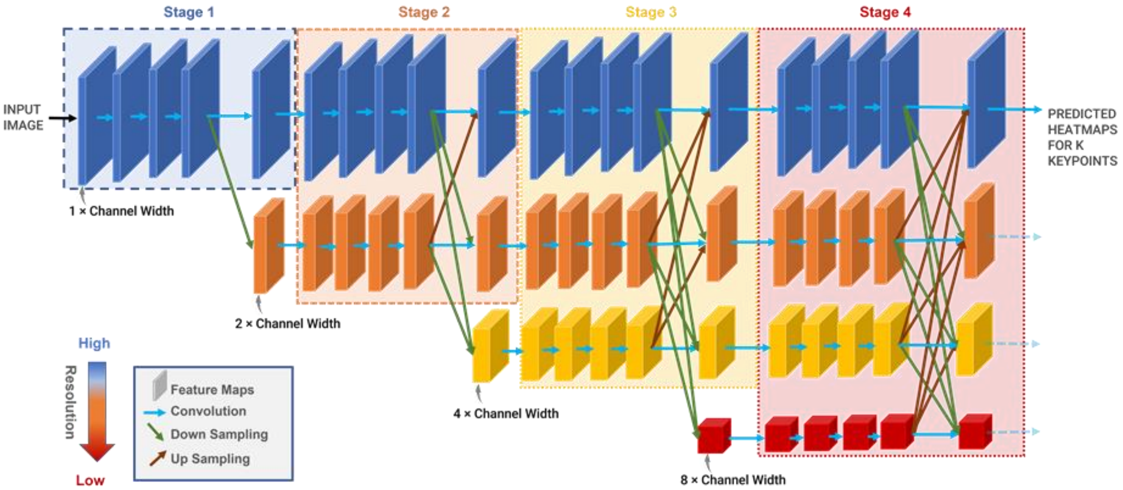


Fig. 3.6 HRNet Model Architecture

3.1.5 ConvNeXt

Among the various CNN architectures, ConvNeXt is employed as a feature extractor, incorporating standard convolutional layers alongside depthwise separable convolutions. Layer Normalization and GELU activation were employed for stabilizing the training, while residual connections similar to ResNet were added for facilitating gradient flow [18]. Additionally pooling layers were also incorporated at significant positions, reducing spatial dimensions while measuring discriminative features. The model architecture process 224×224×3 images, adhering to standard ImageNet dimension. Stage 1 applies a moderate number of convolutional filters for low-level pattern extraction, followed by normalization and activation to be used for strengthened learning. Stage 2 to 4 progressively raise filters and rely on the blocks employing residual connection techniques for burrowing deeper into even more complex representations. In the final stage global average pooling is employed to compress spatial dimensions down to a feature vector per channel. This final representation is fed into the final fully connected layer for classification. This hierarchical pyramidal geometry increasing channel sizes as spatial sizes decrease facilitates the collection of a large range of features.

ConvNeXt builds upon ResNet with modifications to improve performance. One of the important formulas used in ConvNeXt is related to Layer Normalization:

$$LayerNorm(x) = \frac{x-\mu}{\sqrt{\sigma^2+\epsilon}} \cdot \gamma + \beta \tag{3.8}$$

Where:

- x = Input feature map.
- μ = Mean of the input.

- σ^2 = Variance of the input.
- ϵ = A small constant to prevent division by zero.
- γ and β = Learnable parameters for scaling and shifting.

This formula ensures stable training and reduces internal covariate shift.

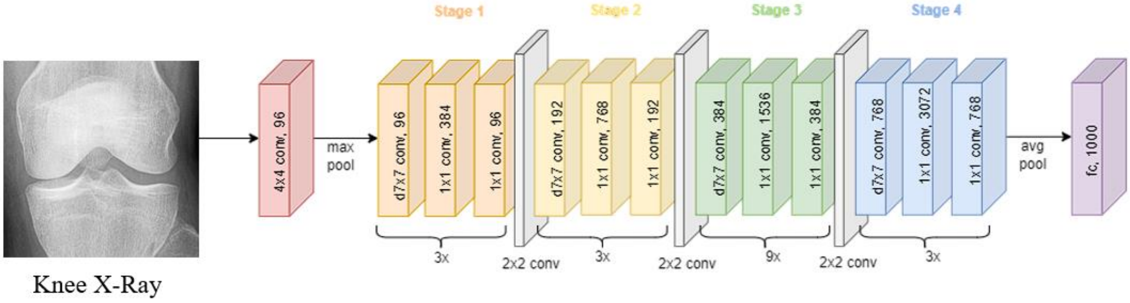


Fig. 3.7 ConvNeXt Model Architecture

3.2 Multi-Level Classification Approach

To enhance classification performance and better distinguish subtle differences among abnormal cases, a hierarchical approach is proposed. This approach consists of two independently trained models: Level-1 Binary Classification: The first model is designed to separate the normal (KL grade 0) from the abnormal (KL grades 1-4) cases. The data for this purpose is shown in Table 4-2. By focusing only on this binary decision, the model can reliably differentiate between normal cases from those with any degree of osteoarthritic change. Level-2 Abnormal Classification: For all images classified as abnormal in the entire dataset, the second model is trained to further classify between the abnormal classes (KL grade 1-4). Data distribution for this task is described in Table 4-sss3. The model is specifically trained to capture the subtle differences between the abnormal cases. The overall architecture of the multi-level framework is shown in Fig. 3.8. This hierarchical structure distinguishes normal and abnormal cases first followed by abnormal images classification. Training was conducted using regular protocols, with the above methods used during dataset loading. Both tasks were measured using accuracy, precision, recall, f1 score, classification report, and confusion matrix, which gave straightforward insights into model performance.

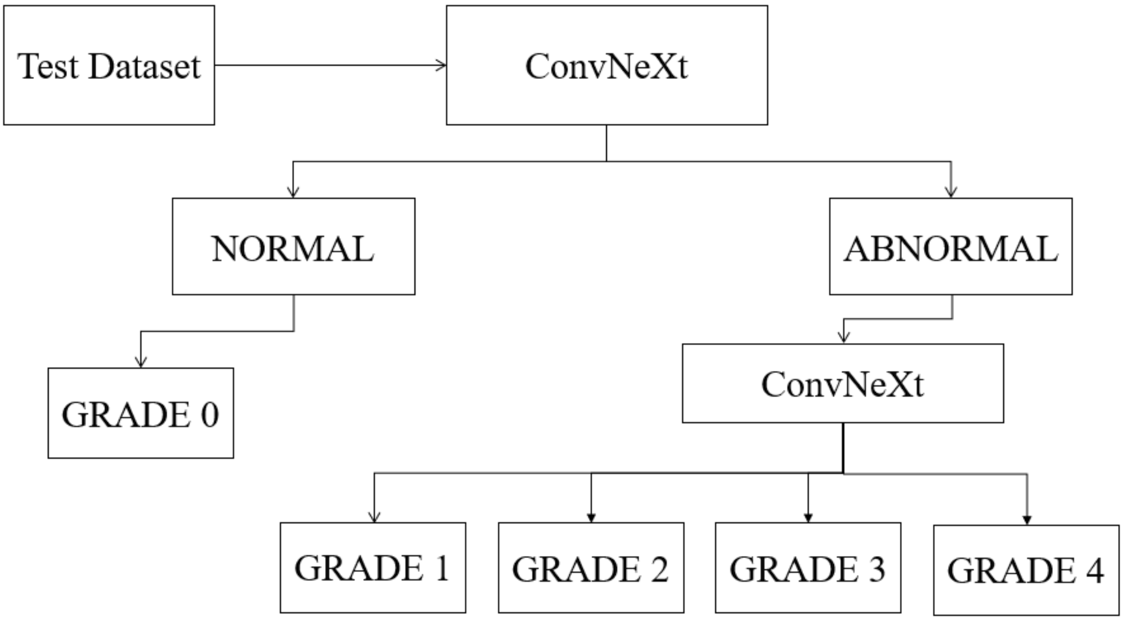


Fig. 3.8 Multi-Class Multi-Level

3.3 Software

3.3.1 Visual Studio Code

Visual Studio Code, an initiative of Microsoft, is a light but powerful code editor that is used on multiple platforms; its markets include Windows, macOS, and Linux. Its versatility and performance make it one of the best tools used by developers from different parts of the world. Though light in weight, VS Code has all the specific features of an integrated environment, and that allows developers to write, test, or debug programs easily. With wide language support that includes Python, JavaScript, C++, and many other programming languages, it is perfectly apt for a wide array of developmental work-from software, web design, and machine learning projects.

Among the most important strengths of VS Code is extensibility. Visual Studio Code Marketplace comes with thousands of extensions and can be added in quick succession for enhancing one's capabilities of the editor. Such extensions could be third-party support for new programming languages, frameworks, or other tools. For example, while working on a machine learning project or on any data science project, extensions like "Python" and "Jupyter" are convenient. These extensions ensure an easy ride in the development, testing, and debugging of any machine learning model, such as an image classification model.

The debugging capabilities of VS Code are also very strong. It comes with a feature set for in-built breakpoints, stepping through code, and live monitoring of variables. Version control integration on top of Git can be provided for simplification of workflow by allowing users to do repository management, track their changes, and collaborate on code right from their editor. Another add-in like the built-in terminal helps execute command-line tasks, which are really useful for running scripts and managing packages right within the development environment.

One of the strengths of VS Code is that it allows for customization. Developers can make the editor look the way they want through a large selection of themes, personalized keyboard shortcuts, and customized workspace settings. This flexibility enables an experience better suited to the individual's or user's preference. What's more is that the Live Share extension enables live sharing. Provided that team members have installed the latest edition of the Live Share extension, they ought to be able to work on the same codebase from a remote location easily.

3.3.2 Google Colab

Google Colaboratory, or Google Colab, is a cloud-based, free platform that enables the user to write and run Python code within a browser. It is Google-developed and runs on the technology of Jupyter Notebook and has become an essential tool for data scientists, machine learning engineers, researchers, and students. You can easily run Python code on high-performance machines without having a local environment in place. It's particularly well-liked in research and education because it's easy to access, has collaboration capabilities, and is integrated with cloud services such as Google Drive.

The central concept of Google Colab is to provide powerful computing resources-particularly those needed for machine learning-widely accessible. Colab makes high-performance computing accessible to all by providing GPUs and TPUs for free, which would otherwise be out of reach or too costly for individuals. This has rendered Colab extremely useful for users working on computationally intensive projects such as training deep learning models or working with large data sets. It reduces the barrier to entry, enabling users to experiment with ideas, execute code, and conduct experiments without having to invest in costly hardware.

Google Colab is closely coupled with Google Drive, enabling notebooks to be saved and synced automatically in real-time. This removes the hassle of saving your work manually and collaboration is as simple as sharing a Google Doc. You can structure your notebooks in folders, access them with any device, and switch to previous versions when necessary. Furthermore, Google Drive can be mounted inside the notebook so you can read and write files from your Drive while code is being run. This provides ease of use when working with datasets, model checkpoints, or any type of external file.

One of the largest benefits of using Colab is its collaborative features in real-time. Any number of users can edit the same notebook at the same time, making it perfect for pair programming, code review, or classroom settings. Users can add comments to cells, propose edits, and chat while collaborating. This feature, shared with other Google Workspace apps such as Docs and Sheets, distinguishes Colab from regular IDEs or local Jupyter notebooks, which tend to be isolated to an individual user.

Colab includes most of the most popular Python libraries pre-installed, such as NumPy, Pandas, Matplotlib, Seaborn, TensorFlow, Keras, PyTorch, Scikit-learn, and OpenCV. Users can therefore begin developing and training models without needing to install anything by hand. Still, for more sophisticated or project-oriented needs, the user can install any library with pip or apt commands. Such installations remain valid throughout the active session but must be re-installed if the session is restarted or times out. Still, this flexibility allows you to personalize your environment according to your requirements.

Colab excels at AI and machine learning development. Regardless of whether you're developing a regression model in Scikit-learn or a convolutional neural network (CNN) in TensorFlow, Colab has the facilities and compute for you to dive in. A beginner can read step-by-step guides to find out the fundamental things, or an expert can leverage hardware acceleration to optimize enormous language models or do hyperparameter tuning.

Although Colab provides access to GPUs and TPUs, actual availability and performance can differ depending on user demand and account type (free or Pro). Free-tier users can experience limitations during peak hours. Also, package installation or repeated retrieval of large datasets in each session can hinder workflows. Therefore, notebooks need to be optimized by caching data in Drive, avoiding redundant operations, and keeping sessions idle for less time. These allow for session efficiency and minimizing friction in experimentation.

Google Colab rivals other cloud notebook offerings like Kaggle Kernels, Amazon SageMaker, Azure Notebooks, and JupyterHub. Each of these has advantages, but Colab is remarkable due to the ease of use, integration with Google Drive, real-time collaboration, and free use of high-end hardware. Kaggle provides an identical interface with combined datasets and competitions.

Google Colab has revolutionized Python programming, data analysis, and machine learning among developers, instructors, and researchers. Its zero-setup setup, cloud runtime, access to GPU and TPU, and collaborative features have established it as the first choice of both newcomers and experts alike. Although it could be limiting with regards to session length and data retention, the advantages outweigh such limitations for most users. By bringing the capability of Jupyter notebooks together with Google's infrastructure, Colab has made high-performance computing accessible to everyone and continues to spur innovation in the areas of AI and data science.

3.3.3 Jupyter notebook

Visual Studio Code is considered to be very great support for Jupyter Notebooks since it makes this tool come alive with an extension-its Jupyter which developers and data scientists use to support their data analysis, machine learning, as well as doing research projects. With the Jupyter extension, VS Code allows users to create, edit, and run Jupyter Notebooks directly within the VS Code environment, meaning no need to switch between different tools. By making notebook support available in VS Code, it brings a unified interface onto the screen which combines the interactive nature of Jupyter with the full capabilities of a traditional code editor.

Another significant aspect of Jupyter usage in VS Code is that it offers an interactive coding environment. Code cells can be run one at a time and their results seen inline, step by step, just like the original interface for the Jupyter Notebook. But when this functionality is brought to VS Code, it brings numerous benefits, such as IntelliSense or "smart" autocompletion, debugging tools, and better code formatting, which really make the workflow a lot more streamlined for coding and for interactivity in notebooks.

Other important features of the Jupyter extension in VS Code include the Variable Explorer and Data Viewer. This is mainly because it makes it possible to inspect variables, datasets, and other data structures in more intuitive manners. In fact, when dealing with big-size datasets or complicated arrays, the Data Viewer representation usually comes out to be tabular and more intuitive to deal with. That makes VS Code an excellent environment for data science tasks wherein the exploration as well as visualization of data are considered critical.

VS Code also provides kernel management for Jupyter Notebooks. It can have several kernels, so this would work with different languages like Python and R, and others. The user may easily switch over from one kernel to another, changing the environment, so all dependencies would be in place. Moreover, upon installation of the Python extension into VS Code, it will detect and integrate with multiple environments of Python, such as Conda or virtual environments, and let you be easily able to work with Jupyter Notebooks and the desired libraries. Thus, it can easily ensure usage with the right environment for the notebook, increasing efficiency and reproducibility from one project to another.

4. Results and Discussion

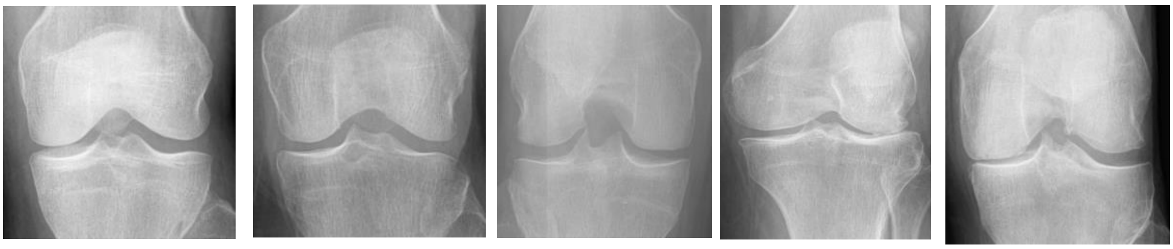
4.1 Data Preparation

The dataset distribution is given in Table 4-1. The proposed work utilizes Knee Osteoarthritis Severity Grading Dataset [19] where the dataset is segregated into different KL grades, representing the severity of knee osteoarthritis. KL Grade 0 has the highest number of samples, with 3,253 images, KL Grade 1 has 1495 images, KL Grade 2 has 2175 images, KL Grade 3 has 1086 images and KL Grade 4 has 251 images. These images are further divided between train, validation, test images.

Table 4-1 Data Set Information

KL grade	Train	Test	Validation	Total
0	2286	639	328	3253
1	1046	296	153	1495
2	1516	447	212	2175
3	757	223	106	1086
4	173	51	27	251

These are the X-ray images of knee which are classified using KL grading scale. From the images we can observe the narrowing of cartilage as we move across the grades which means increasing severity of KOA.



Grade 0 Grade 1 Grade 2 Grade 3 Grade 4

Fig. 4.1 Normal X-ray Images

4.1.1 Level-1 Binary Class Distribution

Table 4-2 shows the distribution of the images used for training and testing the Level-1 model of the multi-level methodology

The multi-level approach consists of two levels, normal and abnormal classification. In the normal and abnormal classification comes under level-1 and further classification of abnormal images comes under level-2. For this the dataset into two parts considering grade 0 as normal and grade 1 to 4 as abnormal. So, the total number of normal images are 3253 and the total number of abnormal images are 5017, which are internally divided between train, validation and test. Table 4-2 shows the distribution of the images used for training and testing the Level-1 model of the multi-level methodology.

Table 4-2 Dataset Distribution

	Normal	**Abnormal**
Trian	2286	3502
Test	639	1017
Evaluation	328	498
Total	3253	5017

4.1.2 Level-2 Abnormal Class Distribution

Table 4-3 shows the distribution of the images used for training and testing the Level-2 model of the multi-level methodology.

The multi-level approach consists of two levels, normal and abnormal classification. In the normal and abnormal classification comes under level-1 and further classification of abnormal images comes under level-2. Now in level-2 we classify the abnormal images further between grade 1,2,3,4. Total number of images for grade 1 are 1495, for grade 2 are 2175, for grade 3 are 1086 and for grade 4 are 251 images. All these images are classified between train, validation and test.

Table 4-3 Data Distribution Across Grades

	Grade1	Grade2	Grade3	Grade4
Trian	1046	1516	757	173
Test	296	447	223	51
Evaluation	153	212	106	27
Total	1495	2175	1086	251

4.2 Evaluation Metrics

Accuracy: Accuracy [20] is the proportion of correctly predicted instances (both positive and negative) out of all predictions. It provides a general measure of the model's performance.

$$Accuracy = \frac{TP + TN}{TP+TN+FP+FN} \tag{4.1}$$

Where:

- TP = True Positives
- TN = True Negatives
- FP = False Positives
- FN = False Negatives

Precision: Precision is how many images Precision calculates the ratio of true positive predictions to the total predicted positives. It reflects the model’s accuracy in predicting relevant instances.

$$Precision = \frac{TP}{TP+FP} \tag{4.2}$$

Recall: Recall is the ratio of true positive predictions to the total actual positives. It measures the model's ability to identify all relevant instances.

$$Recall = \frac{TP}{TP + FN} \tag{4.3}$$

F1-Score: The F1 score is the harmonic mean of precision and recall. It provides a single metric to evaluate the balance between false positives and false negatives. A high F1 score (close to 1) indicates the model performs well in both identifying true positives and avoiding false positives.

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{4.4}$$

4.3 Performance of Proposed Models

4.3.1 Performance of Multi-Class Single-Level

Table 4-4 is a tabular representation of the performance of single-level CNN models trained with five different architectures, namely ResNet50, VGG16, EfficientNetB0, HRNet, and ConvNeXt. ConvNeXt achieved an overall accuracy of 71.26% with a high precision, and F1-score values when compared to other state of the art model architectures are provided in the table. An examination of the confusion matrix for the ConvNext model showed that when the five original classes (0-4) are considered in a binary sense, class 0 being" Normal" and classes 1-4 together being" Abnormal", there was a high recall of 91.07% for the normal class. The average weighted recall for the abnormal classes was a mere 58.80% as depicted in Table 4-6.

For the single-level classification method, some of the prominent Convolutional Neural Networks (CNNs), viz. VGG16, ResNet50, EfficientNet, HRNet, and ConvNeXt, were used to compare how well they worked on the image dataset provided. All these structures were trained and tested under uniform experimental settings in order to carry out a similar comparison. The results of these individual models with respect to specific evaluation metrics such as accuracy, precision, and F1-score were recorded methodically and explored.

Of these models, ConvNeXt showed the most encouraging performance. It recorded the highest accuracy of 71.26%, which signifies its best capability to classify the input images correctly. Moreover, ConvNeXt registered a precision of 71.56%, which indicates its strength in reducing false positives, and an F1-score of 69.38%, which indicates a good balance between precision and recall. These findings indicate that ConvNeXt can learn more from the image in terms of relevant features than the other CNNs in this configuration, and thus it is the best model for this particular task.

Table 4-4 Performance Comparison of Single-Level Models

Model	Accuracy (%)	Precision (%)	F1-Score (%)
VGG 16	63.77	61.83	60.64
ResNet50	59.36	63.59	60.92
EfficientNet	62.86	63.60	62.76
HRNet	64.73	65.05	64.49
ConvNeXt	71.26	71.56	69.38

The confusion matrix of multi class single-level model for five deep learning models VGG16, ResNet50, EfficientNetV2B0, HRNet, and ConvNeXt, are represented from Fig. 4.2 to Fig. 4.6 respectively. The matrices give a visual representation which shows the correct and incorrect number of predictions per class. Analyzing these confusion matrices, we are able to have a clearer idea of each architecture's classification accuracy, precision, and recall.

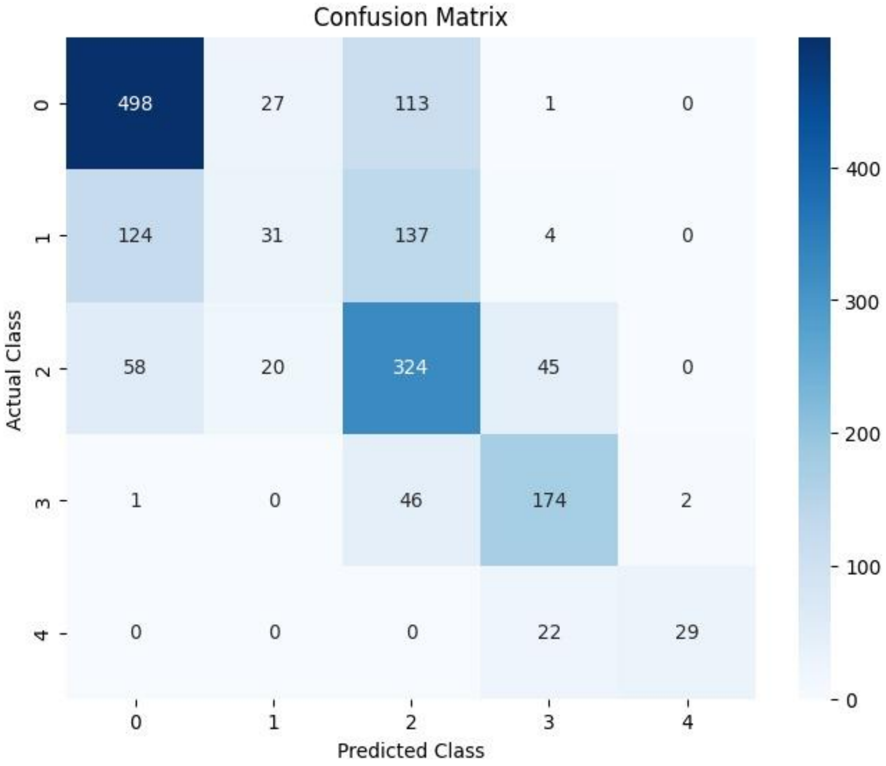


Fig. 4.2 Confusion Matrix of VGG16

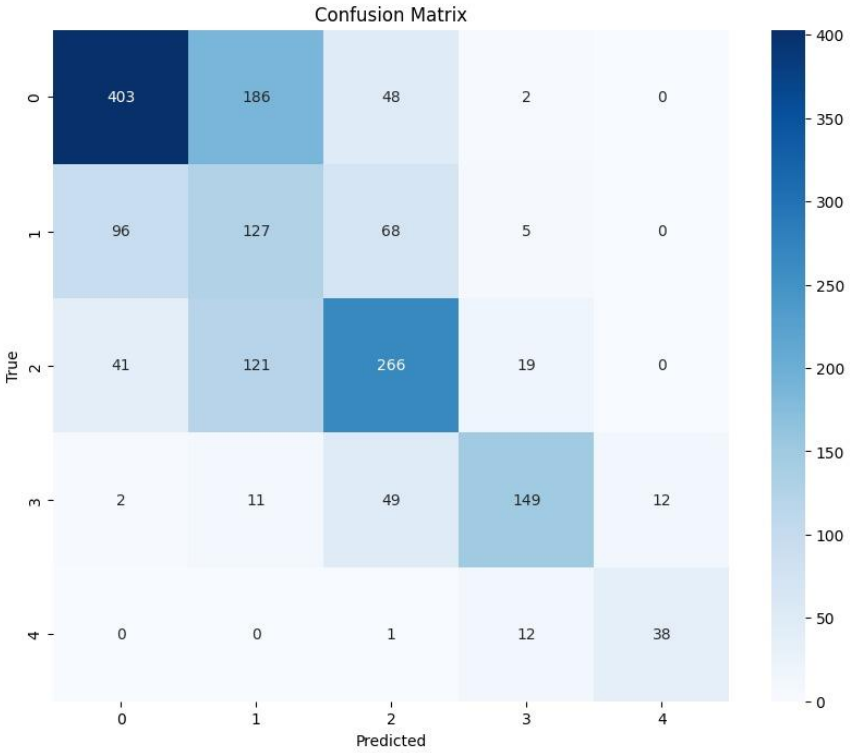


Fig. 4.3 Confusion Matrix of ResNet50

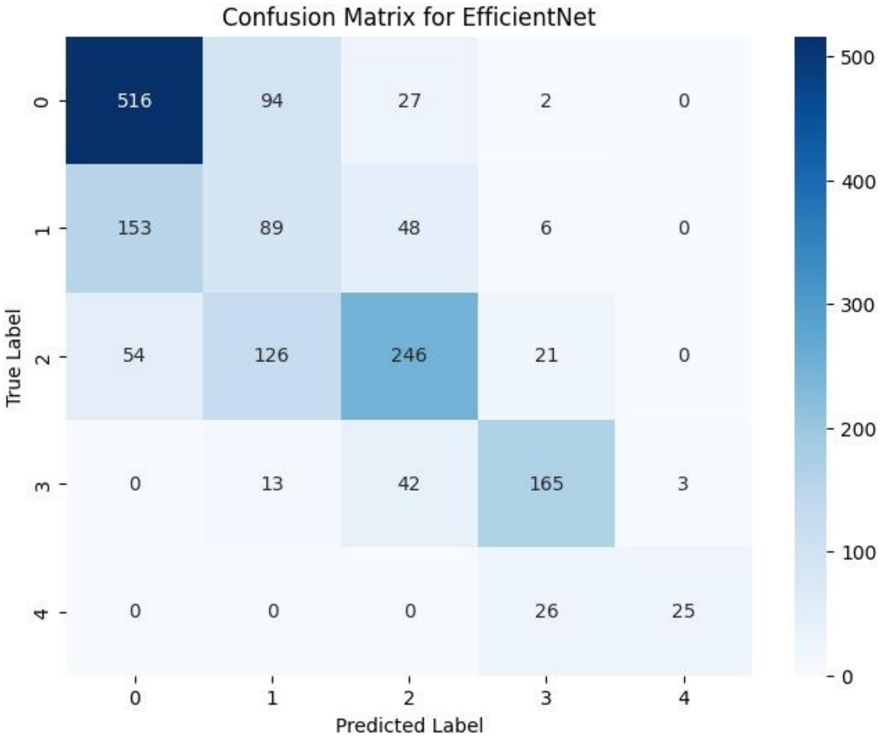


Fig. 4.4 Confusion Matrix of EfficientNet

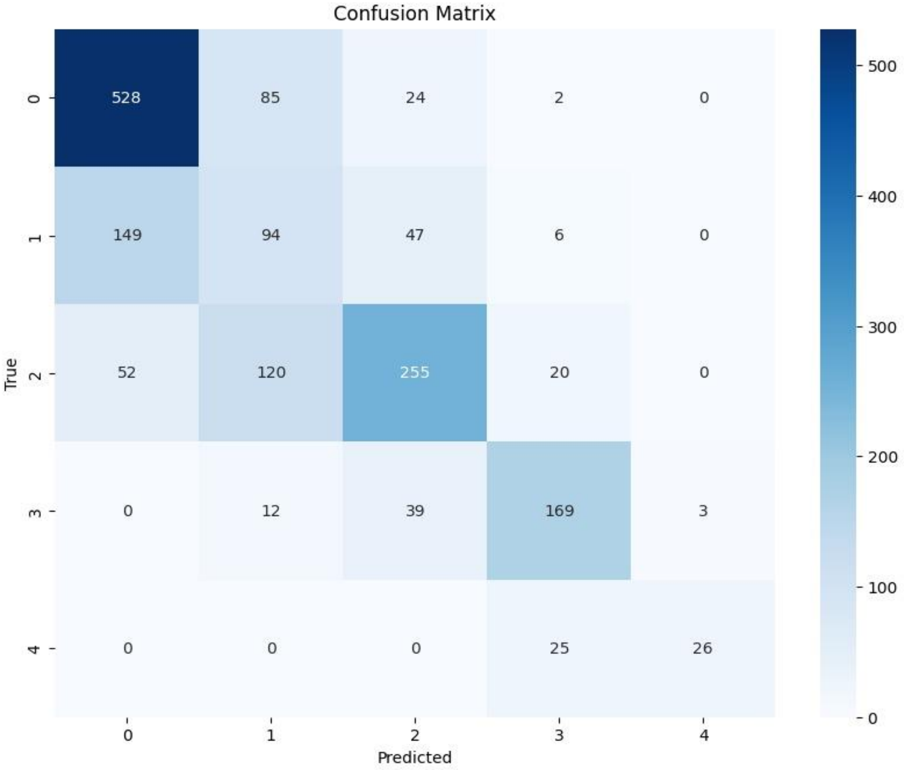


Fig. 4.5 Confusion Matrix of HRNet

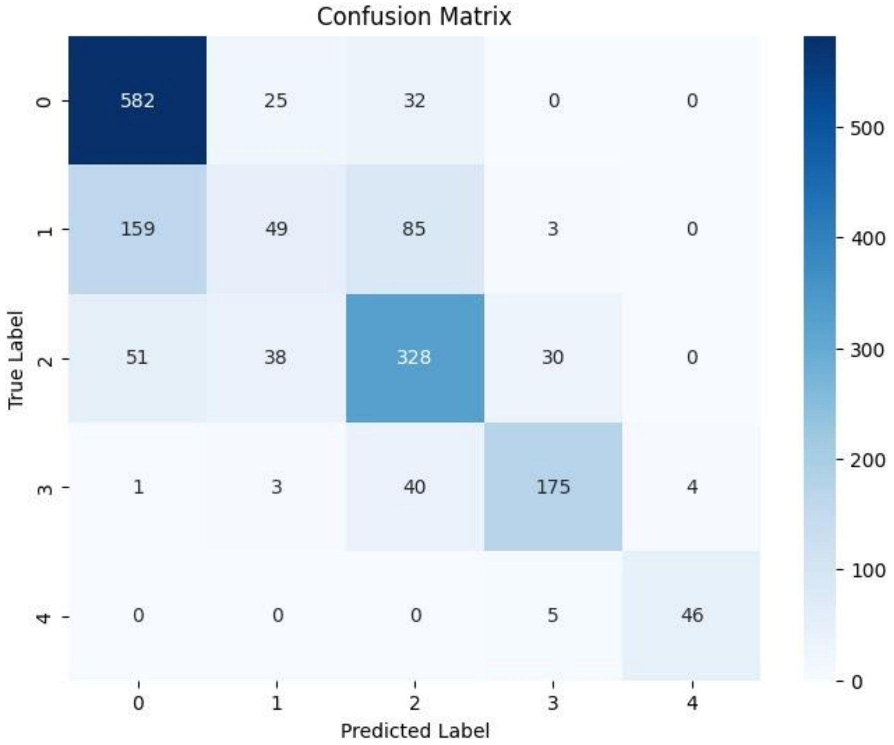


Fig. 4.6 Confusion Matrix of ConvNeXt

To bridge this gap, a multi-level classification as strategy was employed. In the initial phase, a binary classifier model was trained to differentiate between normal and abnormal classes. For this experiment, the best-performing single-level models, HRNet and

ConvNext, were employed as the backbone for building the two-stage classifier. As illustrated in Table 4-5, the level-1 model based on ConvNext demonstrated a significant improvement in binary classification performance with an overall accuracy of 80.62%. While the recall for class 0, normal dropped from 91.07% to 74.49%, the recall for the abnormal class rose from 58.80% to 84.46%. These results undoubtedly show the enhancement of greater balance between specificity and sensitivity within the binary classification step when a multilevel method is used rather than applying a single level model.

As shown in Table 4-5, the level-1 model based on ConvNext demonstrated a significant improvement in binary classification performance with an overall accuracy of 80.62%. Thereafter, a level-2 model was trained only on the abnormal classes (grades 1-4) in order to better categorize the abnormal grades. ConvNeXt performed well for the abnormal image classification with an overall accuracy of 73.94% which is a considerable improvement from the average weighted recall of 58.80% recorded by the single-level model for the four abnormal grades.

Table 4-5 Performance Comparison of Multi-Level Models

Model	Accuracy (%)	Precision (%)	F1-Score (%)
HRNet (Binary)	74.40	77.73	75.07
ConvNeXt (Binary)	80.62	80.53	84.26
HRNet (Abnormal)	69.81	70.60	69.60
ConvNeXt (Abnormal)	73.94	74.29	74.22

Thereafter, a level-2 model was trained only on the ab normal classes (grades 1-4) in order to better categorize the abnormal grades. The level-2 model performed an overall accuracy of 73.94%, which is a considerable improvement from the average weighted recall of 58.80% recorded by the single-level model for the four abnormal grades. A close look at the classification reports (Table 4-5) indicates that grade 1 recall enhanced from 16.55% in the single-level model to 66.55% when using the level-2 model. For grade 2, recall moderately moved from 73.37% to 76.06%, whereas for grade 3, it shifted from 78.47% to 76.23%. Grade 4 dropped slightly from 90.19% to 88.23%; still, the general impact was improved performance with level-2 model producing a considerably improved performance.

4.3.2 Performance of Multi-Class Multi-Level

From table 4-6 we can see the recall for class 0 (normal) dropped from 91.07% to 74.49%, the recall for the abnormal class rose from 58.80% to 84.46%. These results undoubtedly show the enhancement of greater balance between specificity and sensitivity within the binary classification step when a multilevel method is used rather than applying a single level model**.**

Table 4-6 presents a per-class recall value comparison between the single-level model's binary interpretation and the level-1 output of the multilevel model, indicating greater discrimination in the multilevel framework. Also, the single level model and level-2 model per-grade recall comparisons point to notably dramatic improvements, notably for grade 1, there by achieving early detection. Overall, the multilevel modeling method not only enhanced the performance of binary classification (from 71.26% to 80.62% accuracy in the abnormal detection) but also efficiently improved the fine-grained classification between Abnormal grades (with an overall level-2 accuracy of 73.94%). In both phases, ConvNeXt-based model performed outperformed HR Net, as can be seen from the comparative results reported in Table 4-5.

Table 4-6 Model Performance Comparison

Class	**Single-Level Model (Recall %)**	**Multi-Level Model Level-2 (Recall %)**
Grade 1	16.55	66.55
Grade 2	73.37	76.06
Grade 3	78.47	76.23
Grade 4	90.19	88.23
Weighted-Average	58.80	73.94

Table 4-7 presents a per-class recall value comparison between the single-level model’s binary interpretation and the level-1 output of the multilevel model, indicating greater discrimination in the multilevel framework. If we look the values in table 4-7 grade 1 recall enhanced from 16.55% in the single-level model to 66.55% when using the level-2 model. For grade 2, recall moderately moved from 73.37% to 76.06%, whereas for grade 3, it shifted from 78.47% to 76.23%. Grade 4 dropped slightly from 90.19% to 88.23%, still the general impact was improved performance with level-2 model producing a considerably improved performance. Also, the single level model and level 2 model per-grade recall comparisons point to notably dramatic improvements, notably for grade 1, there by achieving early detection.

Table 4-7 Comparison of Single-Level Model and Multi-Level Model Level-2

Class	Single-level Model Interpreted in Binary form (Recall %)	Multi-Level Model Level-1 (Recall %)
Grade 0(Normal)	91.07	74.49
Grade 1-4(Abnormal)	58.80	84.46
Weighted Average	71.26	80.62

The confusion matrix of the multi-class multi-level model for two deep learning models HRNet, and ConvNeXt are represented from Fig. 4.7 to Fig. 4.10 respectively. The matrices give a visual representation which shows the correct and incorrect number of predictions per class. Analyzing these confusion matrices, we are able to have a clearer idea of each architecture's classification accuracy, precision, and recall.

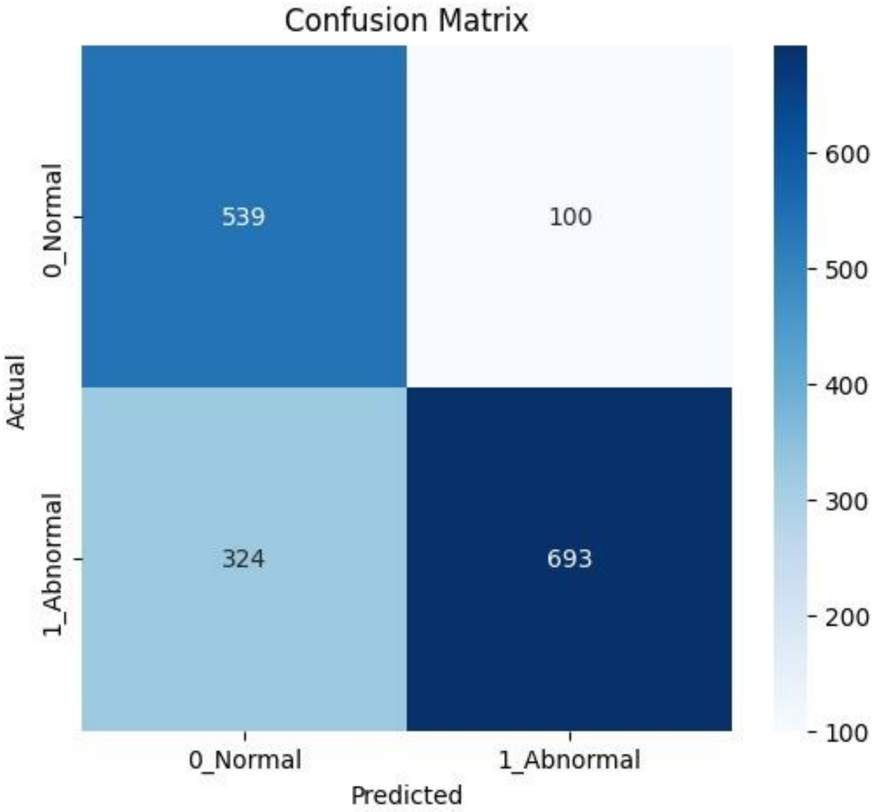


Fig. 4.7 HRNet Binary

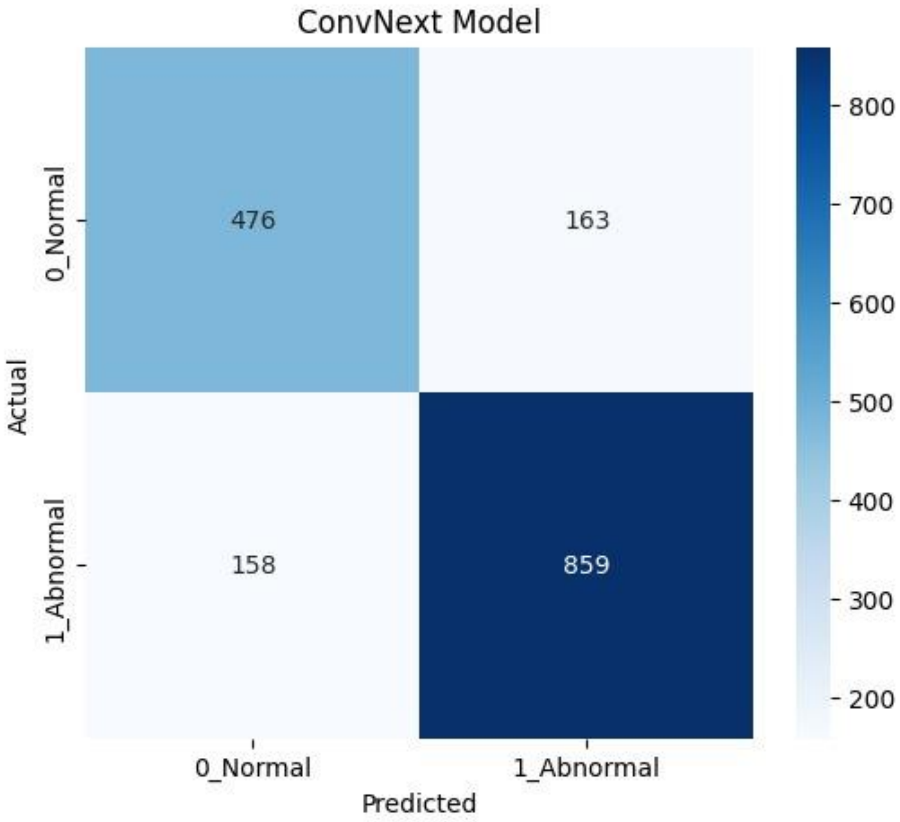


Fig. 4.8 ConvNeXt Binary

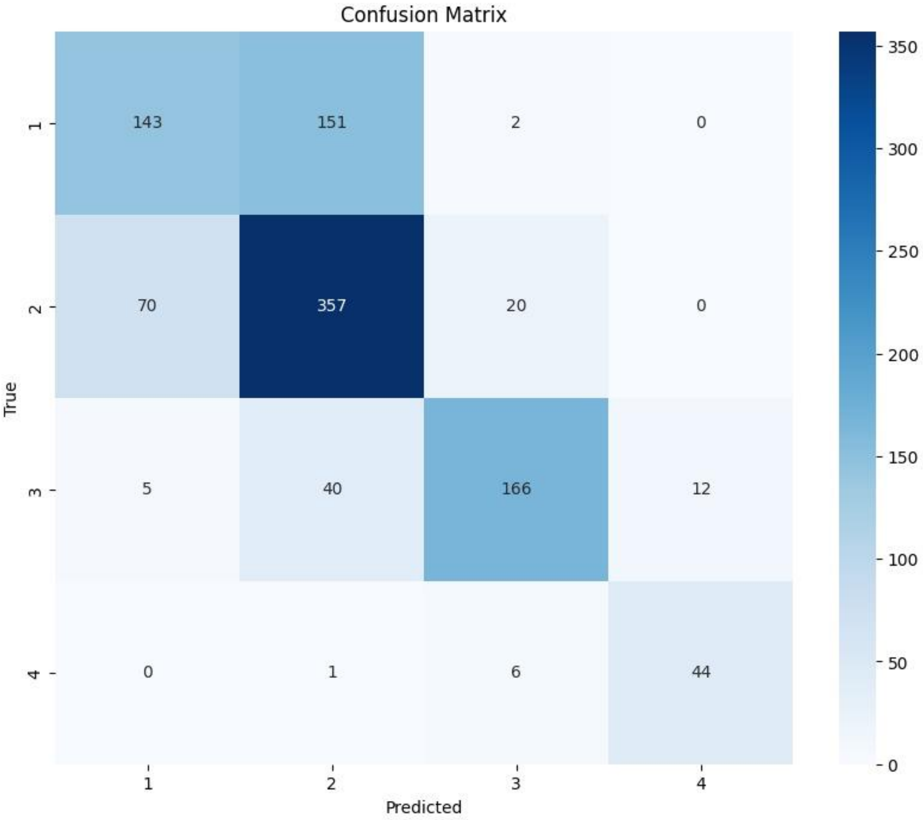


Fig. 4.9 HRNet Abnormal

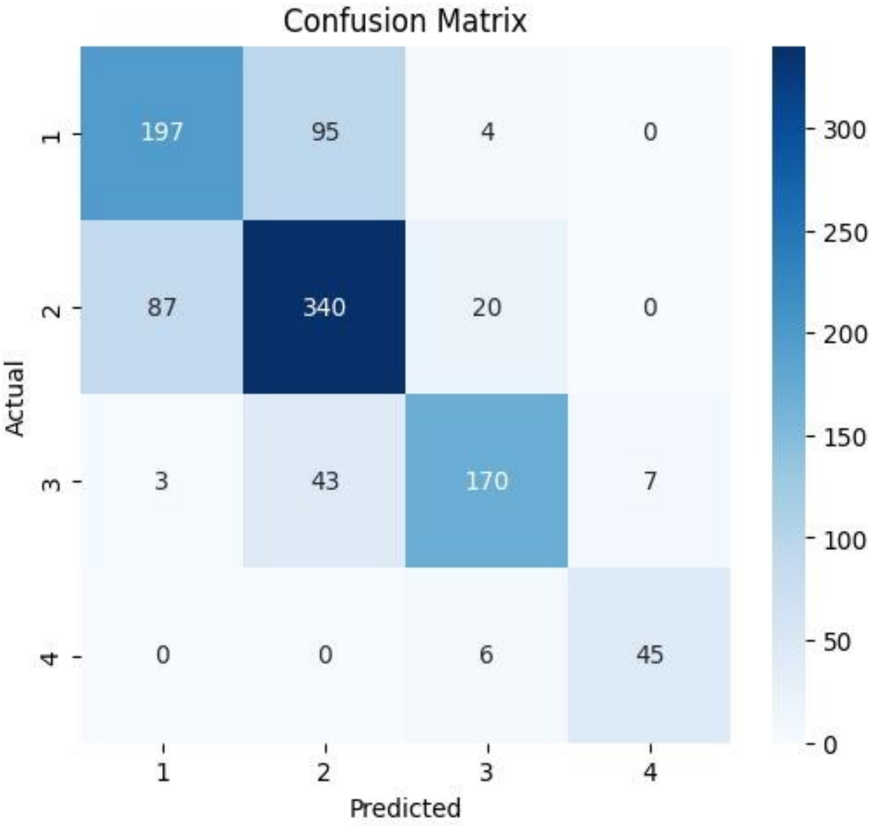


Fig. 4.10 ConvNeXt Abnormal

5. Conclusion

In the proposed work, single-level and multi-level classification approaches for identifying Kellgren-Lawrence (KL) grades in knee X-ray images were employed. In the single level classification framework, five convolutional neural network (CNN) models were evaluated, with ConvNeXt achieving a maximum accuracy of 71.26% with precision and sensitivity of 71.56% and 71.26% respectively. Based on these findings, we constructed a multi-level approach using the best performing models, HRNet and ConvNeXt. The multi-level classification system demonstrated improved performance over the single-level model. The binary classification model, achieved an accuracy of 80.6% for normal (KL-0) abnormal (KL 1-4) classification, highlighting the advantages of a hierarchical strategy. Additionally, the abnormal classification model, which categorized KL grades 1-4, achieved an accuracy of 73.94%, further reinforcing the effectiveness of a multi-level approach.

The results indicate that a hierarchical framework enhances classification performance by enabling early detection and severity prediction. For future research, ensemble learning framework can be employed to further improve classification accuracy and robustness. Additionally, integrating the binary-level and abnormal-level models into a unified frame work could allow for simultaneous normal/abnormal classification and KL grade prediction. This integrated approach would streamline clinical workflows by providing comprehensive diagnostic information, ultimately facilitating early diagnosis and improved patient management for knee osteoarthritis.

For future work, applying ensemble techniques to continue enhancing the accuracy and robustness of the classification system. Additionally, working towards combining the binary level and abnormal-level models under a common framework, such that simultaneous prediction of both normal/abnormal categorization and KL grade determination becomes possible. Through such an integration, clinical work can be facilitated, and whole-spectrum diagnostic data can be returned in one pass of inference, ultimately enabling enhanced early diagnosis and patient management for knee osteoarthritis.

References

[1] E. Bahar, D. Shamarina, Y. Sergerie, and P. Mukherjee, “Descriptive overview of pertussis epidemiology among older adults in Europe during 2010–2020,” *Infect. Dis. Ther.*, vol. 11, no. 5, pp. 1821–1838, Jul. 2022.

[2] E. A. Fallon, “Prevalence of diagnosed arthritis — United States, 2019–2021,” *MMWR Morb. Mortal. Wkly. Rep.*, vol. 72, no. 41, Oct. 2023.

[3] L. Jiang et al., “Body mass index and susceptibility to knee osteoarthritis: A systematic review and meta-analysis,” *Jt. Bone Spine*, vol. 79, no. 3, pp. 291 297, May 2012.

[4] M. U. Farooq, Z. Ullah, A. Khan, and J. Gwak, “DC-AAE: Dual channel adversarial autoencoder with multitask learning for KL-grade classification in knee radiographs,” *Comput. Biol. Med.*, vol. 167, pp. 107570–107570, Oct. 2023.

[5] Y. Wang et al., “Learning from highly confident samples for automatic knee osteoarthritis severity assessment: Data from the Osteoarthritis Initiative,” *IEEE J. Biomed. Health Inform.*, vol. 26, no. 3, pp. 1239–1250, Mar. 2022.

[6] Rohit Kumar Jain, Prasen Kumar Sharma, S. Gaj, A. Sur, and P. Ghosh, “Knee osteoarthritis severity prediction using an attentive multi-scale deep convolutional neural network,” *Multimed. Tools Appl.*, Jun. 2023.

[7] P. Chen, L. Gao, X. Shi, K. Allen, and L. Yang, “Fully automatic knee osteoarthritis severity grading using deep neural networks with a novel ordinal loss,” *Comput. Med. Imaging Graph.*, vol. 75, pp. 84–92, Jul. 2019.

[8] T. S. Apon, M. F. Islam, N. I. Rafin, J. Akter, and M. G. R. Alam, “Transforming precision: A comparative analysis of vision transformers, CNNs, and traditional ML for knee osteoarthritis severity diagnosis,” *2024 6th Int. Conf. Electr. Eng. Inf. Commun. Technol. (ICEEICT)*, Dhaka, Bangladesh, May 2024, pp. 1–6.

[9] D. W. Lee, D. S. Song, H.-S. Han, and D. H. Ro, “Accurate, automated classification of radiographic knee osteoarthritis severity using a novel method of deep learning: Plug-in modules,” *Knee Surg. Relat. Res.*, vol. 36, no. 1, Aug. 2024.

[10] S. Aslan, “Automatic Detection of Knee Osteoarthritis Disease with the Developed CNN, NCA and SVM Based Hybrid Model,” *Traitement du Signal*, vol. 40, no. 1, pp. 317–326, Feb. 2023.

[11] A. R. Patil and S. Salunkhe, “Classification and Risk Estimation of Osteoarthritis Using Deep Learning Methods,” *Measurement: Sensors*, p. 101279, Jul. 2024.

[12] K. Fatema et al., “Development of an automated optimal distance feature-based decision system for diagnosing knee osteoarthritis using segmented X-ray images,” *Heliyon*, vol. 9, no. 11, p. e21703, Nov. 2023.

[13] Y. X. Teoh, A. Othmani, K. W. Lai, and S. L. Goh, “Stratifying knee osteoarthritis features through multitask deep hybrid learning: Data from the osteoarthritis initiative,” *Comput. Methods Programs Biomed.*, vol. 242, p. 107807, Sep. 2023.

[14] A. S. C. Bose, C. Srinivasan, and S. I. Joy, “Optimized feature selection for enhanced accuracy in knee osteoarthritis detection and severity classification with machine learning,” *Biomed. Signal Process. Control*, vol. 97, 106670, Aug. 2024.

[15] Y. Wang, Z. Bi, Y. Xie, T. Wu, X. Zeng, S. Chen, and D. Zhou, “Learning from highly confident samples for automatic knee osteoarthritis severity assessment: Data from the Osteoarthritis Initiative,” *IEEE J. Biomed. Health Inform.*, vol. 26, no. 3, pp. 1239–1250, Mar. 2022.

[16] M. B. Manoj, N. Mohan, S. K. S, and K. P. Soman, “Automated detection of kidney stone using deep learning models,” in *Proc. 2022 2nd Int. Conf. Intell. Technol. (CONIT)*, vol. 14, no. 1, pp. 448–458, Jun. 2022.

[17] Srividhya L, Sowmya V, V. Ravi, Gopalakrishnan E. A., and Soman K. P., “Deep learning-based approach for multi-stage diagnosis of Alzheimer’s disease,” *Multimed. Tools Appl.*, vol. 83, no. 6, pp. 16799–16822, Jul. 2023.

[18] Erdogan, “ConvNeXt: Next generation of convolutional networks,” *Medium*, Feb. 1, 2023. [Online]. Available: https://medium.com/@atakanerdogan305/convnext-next-generation-of-convolutional-networks-325607a08c46. [Accessed: Mar. 5, 2025].

[19] P. Chen, “Knee osteoarthritis severity grading dataset,” *Mendeley Data*, v1, 2018. [Online]. Available: https://tinyurl.com/4h7pwaaa. [Accessed: Sept. 12, 2024].

[20] T. Tariq, Z. Suhail, and Z. Nawaz, “Knee osteoarthritis detection and classification using X-rays,” *IEEE Access*, vol. 11, pp. 48292–48303, May 2023.

Publication

[1] C. R. Vardhan Reddy, D. M. Shaahid, D. Santhosh, N. Sumanth, and D. Vijayan, "Multi-Class Classification and Detection of Knee Osteoarthritis From X-rays," *Proceedings of the International Conference on Artificial Intelligence and Computation (AIC 2025*) (Targeted), Under preparation.