Diagnosing Osteoarthritis by Proposing Red Fox Optimization for Convolutional Neural Network

Abdullah Dawood Sulaiman [1,2,*], Raid Rafi Omar Al-Nima [3,4], and Tingting Han [5]

[1] Technical Engineering College, Northern Technical University, Mosul, Iraq
[2] Directorate of Ninawa Health, Ministry of Health, Mosul, Iraq
[3] Artificial Intelligence Research Center, Northern Technical University, Mosul, Iraq
[4] Technical Engineering College for Computer and Artificial Intelligence, Northern Technical University, Mosul, Iraq
[5] School of Computing and Mathematical Sciences, Birkbeck, University of London, United Kingdom
Email: abdullah_da@ntu.edu.iq (A.D.S.); raidrafi3@ntu.edu.iq (R.R.O.A.-N.); t.han@bbk.ac.uk (T.H.)
*Corresponding author

***Abstract*—Osteoarthritis (OA) is a chronic disorder of synovial joints characterized by deterioration of joint cartilage and underlying bone. The increasing prevalence of the disease, driven by an ageing population and rising obesity rates, necessitates the development of accurate diagnostic tools for early intervention. In this study, the Red Fox Optimization for Convolutional Neural Network (RFOCNN) framework was proposed to diagnose knee OA. It integrates the Red Fox Optimization (RFO) metaheuristic with a Convolutional Neural Network (CNN). Specifically, RFO optimizes key CNN hyperparameters. The RFOCNN is evaluated on the Digital Knee X-ray Images dataset (DKXI), which comprises 1820 knee X-ray images. It is designed to perform binary classification between normal knees and OA knees. Additional validation experiments compare the proposed framework with other Deep Learning (DL) models, metaheuristic methods, and alternative inputs. The RFOCNN framework achieved a testing accuracy of 90% and an Area Under the Curve (AUC) of 0.96. In comparisons against DL models, baseline CNN, CNN tuned with random search, and CNN optimized with other metaheuristic algorithms, the RFOCNN demonstrated a consistent performance advantage over the compared methods. Such results highlight that the RFOCNN framework can successfully identify knees with OA.**

***Keywords*—Osteoarthritis, deep learning, red fox optimization**

I. Introduction

Osteoarthritis (OA) is the most prevalent type of arthritis. It is characterized by degeneration of joint cartilage and underlying bone [1, 2]. It is a chronic condition that affects synovial joints, particularly the knees, hips, lower back, wrists and small joints of the fingers. Symptoms of OA include pain, swelling, stiffness, inflammation and trouble moving the affected joint [3].

OA affects approximately 528 million people globally, representing an increase of 113% between 1990 and 2019. The majority of individuals with OA are over the age of 55, accounting for 73% of cases, and 60% of those affected are women [2, 4]. Moreover, an estimated 344 million people experience moderate to severe forms of OA [5].

OA results from a complex interaction between local factors and systemic factors that affect the joints. Systemic factors include age, genetics, female sex, high bone density, and nutrition. Local factors trigger OA in specific joints and include obesity, joint injuries, physical activity, and occupation. The most dominant risk factors for developing knee OA are obesity and joint injuries [1, 6].

The impact of OA on quality of life is considerable. This can prevent patients from participating in social, work-related, and other routine activities. Furthermore, it contributes to mental health challenges and sleep disturbance [7].

The majority of research efforts have concentrated on knee OA. It is the most prevalent type of the disease, affecting approximately 365 million individuals worldwide, followed by hip and hand OA [2, 8]. Pathological changes in the articular cartilage are fundamental to the development of OA. However, other joint tissues are affected by OA, not just the cartilage [1, 3, 9].

The knee joint is a weight-bearing joint with a high degree of mobility. Its structure includes femoral condyles articulating with tibial plateaus and patella, all covered with hyaline cartilage that absorbs mechanical stress and facilitates smooth, pain-free movement of the bones. The stability and movement of the knee joint are provided by the interaction of several structures including menisci, ligaments, tendons, morphology of the articular surfaces and surrounding muscles [3, 10].

Radiography remains the main and most practical imaging modality for assessing and distinguishing between healthy and osteoarthritic knee joints, due to its low cost, rapid acquisition, and wide availability. In a healthy knee X-ray, the femur and tibia are properly aligned without bony spurs. Also, normal articular cartilage appears even and well-preserved, with Joint

Manuscript received May 5, 2026; revised June 2, 2026; accepted August 4, 2026; published November 14, 2026.

Space Width (JSW) in X-ray images. In contrast, the key radiographic indicators of knee OA include osteophyte formation, Joint Space Narrowing (JSN), and bone deformity [11]. Fig. 1 demonstrates the differences between normal and OA knees.

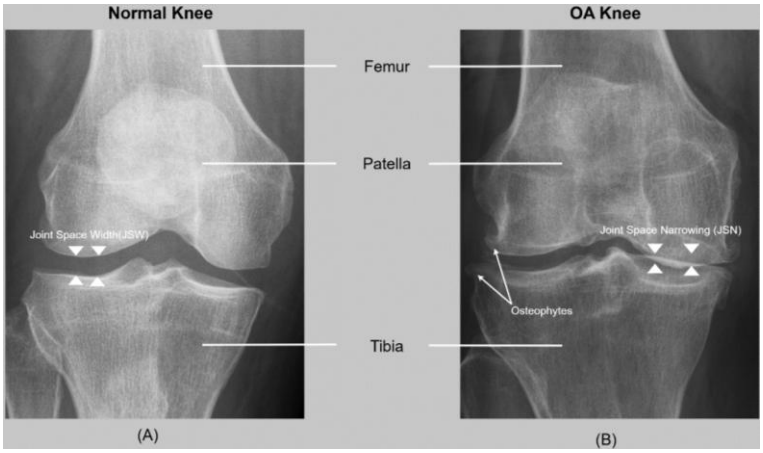


Fig. 1. Demonstration for the differences between normal and OA knees: (A) normal knee, (B) OA knee.

The development and integration of Artificial Intelligence (AI) based methods are essential in OA detection. They can reduce the need for manual evaluation and accelerate the diagnostic workflow. Additionally, they may support junior radiologists and orthopedists to improve their diagnostic performance and increase interobserver agreement levels among different specialties and experts in identifying OA [12].

In recent years, the use of Deep Learning (DL) has gained significant attention. The ability to learn detailed features has enabled DL to become a valuable tool in the medical imaging field, particularly in developing automated methods for detecting OA [13]. Although various traditional Machine Learning (ML) and DL approaches have been explored, establishing well-generalized models for diagnosing OA of knees is still a challenging task.

This study aims to improve Convolutional Neural Network (CNN) performance for OA diagnosis using knee radiography images. It proposes the Red Fox Optimization for the CNN (RFOCNN) framework, which utilizes Red Fox Optimizer (RFO) to automatically tune CNN hyperparameters, encouraging high classification performance.

The text sections will be organized as follows: the literature review is presented in Section II, the proposed RFOCNN framework is introduced in Section III, findings and discussions are presented in Section IV, and the conclusion is presented in Section V.

II. Literature Review

In the literature, many studies related to the detection and classification of OA have been presented, and they can be introduced as follows.

In 2015, Thomson *et al*. [14] proposed a fully automated system for detecting OA from knees radiographs. This system involved fusing shape and texture using Random Forest (RF) classifier, one for bone shape features which derived from an automatic bone segmentation and another for the texture of a knee radiograph based on either Fractal Signature (FS) or simple pixel ratios. A total of 500 radiographs from Osteoarthritis Initiative (OAI) database were used to train the classifier.

In 2016, Antony *et al*. [15] presented a comprehensive method for classifying and detecting knee OA severity. The study was divided into two parts. The first part described how to automatically detect the region of knee joint by using a linear Support Vector Machine (SVM) on horizontal image gradients. Whilst the second part included the assessment of knee OA grades using features extracted from different layers of pretrained CNN models, such as Visual Geometry Group-16 (VGG-16) network, Visual Geometry Group Medium-128 (VGG-M-128) and Berkeley Vision and Learning Center Caffe Network (BVLC CaffeNet). Followed by features extracted from fine-tuned BVLC CaffeNet and VGG-M-128 networks. In both steps, the extracted features were classified by a linear SVM, corresponding to labeled knee X-ray images by Kellgren-Lawrence (KL) grades. The last step treated the assessment of KL grades as a regression problem because of the progressive nature of the disease. Additionally, a fine-tuned BVLC CaffeNet was trained using regression loss, such as Mean Squared Error (MSE). The OAI dataset, which contained 8892 X-ray images for each knee joint, was used for this study. It should be noted that the classification was based on three methods: progressive, successive, and multi-class.

In 2018, Tiulpin *et al*. [16] presented an automatic Computer-Aided Diagnosis (CAD) method to diagnose OA from knee radiographs. The approach here was based on feeding a symmetrical image, which consists of the medial and flipped lateral parts of the knee joint, into a Deep Siamese Convolutional Neural Network (DSCNN) for both medial and lateral compartments of the knee joint. The processes included training three separate models with different random seeds (21, 42, and 84) and fusing their predictions. DSCNN was compared to the pretrained Residual Network-34 (ResNet-34), which was used as a reference. Class-discriminating Gradient-weighted Class

Activation Mapping (Grad-CAM) was also incorporated to highlight relevant radiological features. Entire Multicenter Osteoarthritis Study (MOST) dataset was utilized for training, and the OAI dataset was used for evaluating the performance.

In 2019, Lim *et al*. [17] suggested a DL methodology for the early detection of OA using data on health behavior and medical-related statistics. The system combined a Deep Neural Network (DNN) and scaled Principal Component Analysis (PCA). The data was derived from the 2015–2016 Korea National Health and Nutrition Examination Survey (KNHANES), which focused on data gathered from 5749 older adults on their health behaviors and medical statistics features. First, the data was preprocessed and scaled for four diverse PCAs; then, the generated features from PCA were trained using a DNN with 8 hidden layers, each layer had 90 neurons.

In 2020, Thomas *et al*. [18] presented an automated DL model for severity assessment of OA by comparing the proposed model's performance with two musculoskeletal radiologists' decisions. A pretrained DL model of Densely Connected Network (DenseNet) was employed to predict KL scores for each knee image. The method relied on splitting full X-ray images for each patient according to the left and right knees. During training, 20% of images were preprocessed, and 80% of images were randomly augmented. The training was performed twice, once using preprocessing and augmentation techniques and a second time with preprocessing only to increase the model's generalizability and distribution of images. In addition, Saliency Maps (SM) was used to detect the most dominant regions of images for the model's predictions. Images from the OAI dataset were used. The evaluation was performed essentially with three parameters: accuracy, F1-Score, and Cohen's Kappa Coefficient (*k*).

In 2021, Guida *et al*. [19] proposed a Three Dimensional (3D) CNN model for classifying knee OA using sequence Magnetic Resonance Imaging (MRI). The 3D CNN was Compared with traditional Two-Dimensional (2D) CNN that used X-ray images from the same patients' knees. Two categories of OA were considered: 5-category KL grades and a 2-category that discriminates non-OA (KL-0 and KL-1) from OA (KL-2, KL-3 and KL-4) knees. The OAI dataset of both X-ray and MRI images for the same patients, who were male and female, was utilized.

In 2022, Ahmed and Mstafa [11] proposed a hybrid approach that combines DL and ML models for assessing the severity of knee OA from radiographs. The process involved Deep Hybrid Learning-I (DHL-I) and Deep Hybrid Learning-II (DHL-II). DHL-I used a pretrained CNN to extract features, applied PCA for dimensionality reduction, and SVM for classifying OA into five severity grades. DHL-II employed a Transfer Learning (TL) approach for fine-tuning the pretrained CNN of DHL-I to categorize OA into four, three, and two classes. X-ray images of OAI were used.

In 2023, Chen *et al*. [20] developed a DL model based on Extreme Inception Network (XceptionNet) for the severity rating of hip diseases using digital radiographs. The aim here was to classify two typical hip diseases: Primary Hip Osteoarthritis (PHOA) and Osteonecrosis of Femoral Head (OAFH) in one DL model. The hip joint images were collected from Shanghai Sixth People's Hospital (SSPH) for training and testing the proposed model. The hip joint datasets that were used for evaluating the model were sourced from Foshan Chancheng Central Hospital (FCCH).

In 2023, Wang *et al*. [21] introduced a novel approach for detecting the early stage of knee OA, classifying between KL-0 and KL-2. Vision Transformer (ViT) was used as the core model, enhanced with a Selective Shuffled Position Embedding (SSPE) strategy. The SSPE operates by fixing and shuffling the position embedding of key and non-key patches in the input sequences, which was intended to provide augmentation within the model, directing the ViT's attention to patches related to KL-2. In this study, KL-0 and KL-2 scales were considered as early stages of knee OA. Knee X-rays from the OAI dataset served as inputs to the proposed model after segmenting the region of knee joint by You Only Look Once version 2 (YOLOv2).

In 2023, Pi *et al*. [22] suggested an ensemble DL network for assessing knee OA severity using the KL grading system from radiographs, aiming to overcome the subjectivity and variability of clinician assessments. Eight pretrained models included in this study were: EfficientNet-V2-s, EfficientNet-b5, DenseNet-161, ResNet-101, ResNext-50-32×4d, Wide-ResNet-50-2, RegNet-Y-8GF, and ShuffleNet-V2-×2-0. They were then combined into an ensemble network using a mixed voting algorithm under two conditions: first with fixed input size and second with optimal input size for each model. The dataset utilized in this study was derived from the OAI knee X-ray images. Furthermore, the Grad-CAM visualization technique was employed to interpret the model's predictions. It was revealed that the ensemble network could address the instability of single models and improve their performance.

In 2024, Teh *et al*. [23] investigated the diagnosis of knee OA using multi-modal and unimodal neural networks. The multi-modal were: MF_DIKO, MF_InceptionNet and MF_Eff, which transformed from the unimodal models DIKO, InceptionV3 and EfficientNetv2 that share a common structure. Unimodal as image feature extraction backbones were combined with separate embedding layers for structured demographic data to construct the multi-modal architecture. The data from OAI included demographic data (height, weight, age, gender, and race) as well as knee X-ray images. The experiments were conducted on dataset containing of 500 and 1000 images per class and classification categories of 3, 4, and 5 KL grades. MF_DIKO and InceptionV3 emerged as the best multi-modal and unimodal neural networks, respectively. The performance of both networks generally decreased as the number of categories increased from 3 to 5 and improved with large dataset sizes.

In 2024, Ahmed and Imran [24] investigated the use of pretrained DL and Explainable Artificial Intelligence (XAI) for the automated diagnosis of knee OA from radiographs, focusing on the KL grading. The method

involved fine-tuning of five pretrained CNN models: Residual Network-50 (ResNet-50), ResNet-101, VGG-16, Visual Geometry Group-19 (VGG-19), and Efficient Convolutional Network b7 (EfficientNetb7), for both binary and multi-class classification tasks. To enhance interpretability, Grad-CAM was applied to visualize the ROI that the models determined during decision-making. The used data was for the Knee OA Severity Grading Dataset (KOSGD), which included KL-graded X-ray images from the University of Florida. For binary classification, subsets were created between normal cases (grade 0) and other individual grades (1, 2, 3 and 4).

In 2025, Mehdi *et al*. [25] presented a DL approach for identifying the early stages of OA in knee X-ray images by fusing shape and texture features. Siamese Convolutional Neural Network (SCNN) extracted shape features from the knee joint area, while a Discriminant Regularized Auto-Encoder (DRAE) extracted texture features from 5 regions of the tibial bone. The extracted features were combined by a Compact Bilinear Pooling (CBP) and a Fully Connected (FC) layer for knee OA detection based on the KL score. Also, classification was performed among early stages, KL-0 versus KL-1 and KL-1 versus KL-2. Data from the available OAI dataset was exploited. Four-fold cross-validation was employed for evaluating the model.

In 2025, Panwar *et al*. [26] proposed a ViT model on MRI images for detecting knee OA severity, aiming to improve accuracy and simplify clinical workflows. The ViT layer involved: splitting images into patches, linearly embedding each patch, applying position embedding, and sending the vector sequence through a transformer encoder. This was followed by a flattening layer to convert multidimensional output to a one-dimensional vector and batch normalization layers. The dataset consisted of MRI images from different individuals and was categorized by KL grades. It was reported that small and specialized datasets can reduce ViT performance.

It can be observed that a significant demand for proposing a framework for optimizing DL models still exists in diagnosing or detecting OA. Especially, by utilizing the potential of the RFO algorithm, which has not yet been investigated in diagnosing OA before, according to our best obtained knowledge. Therefore, this study considers addressing this gap.

III. Methodology

A. Proposed Framework

This study proposes the RFOCNN framework for classifying knee X-ray images into normal and OA classes. In this framework, CNN hyperparameters are optimized using RFO. So, the provided CNN is enhanced and can perform well. Firstly, the CNN architecture is constructed of the following: first convolutional layer with the first activation function, the first pooling layer, the second convolutional layer with second activation function, the second pooling layer, the dense layer, and the classification output. Essentially, 17 CNN hyperparameters are subjected to optimization using RFO. These hyperparameters include: number of filters of two convolutional layers, kernel sizes of the convolutional layers, strides of the convolutional layers, activation function of the convolutional layers, window sizes of two pooling layers, strides of the pooling layers, number of nodes in the dense layer, activation function of the dense layer, dropout of dense layer, optimizer type, and learning rate. A block diagram of the proposed RFOCNN framework is given in Fig. 2.

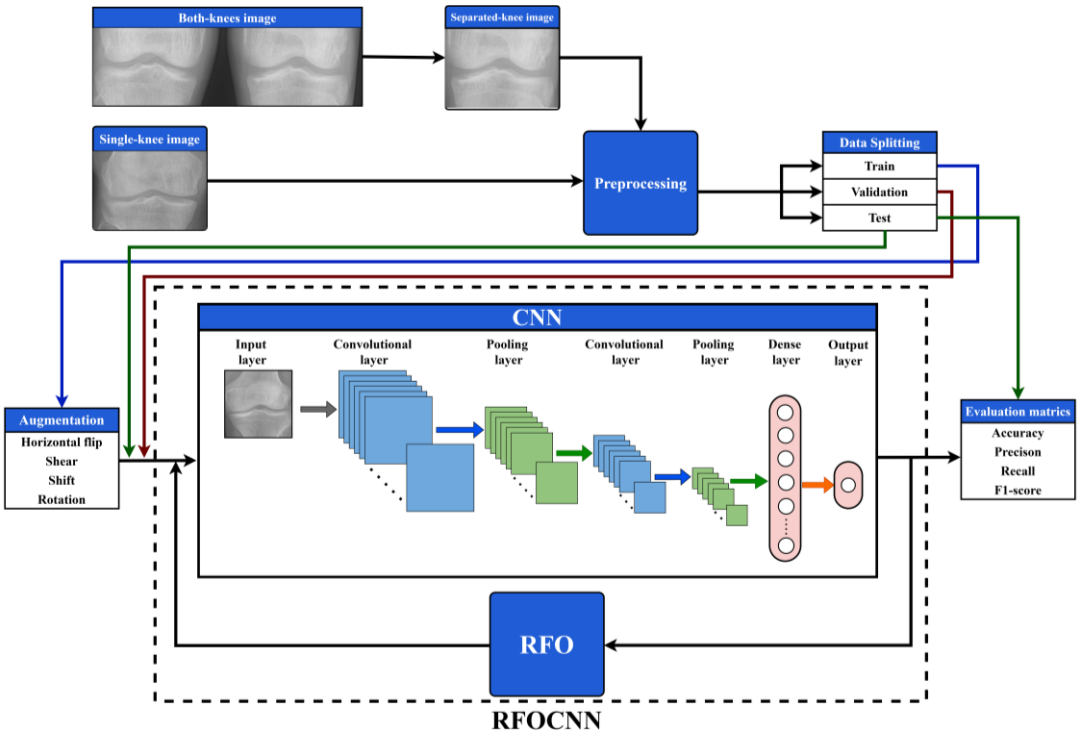


Fig. 2. The block diagram of the work, with the proposed RFOCNN framework.

In this work, the segmented Region of Interest (ROI) of knee X-ray images using a pixel density-based method [27] is considered. This is followed by resizing the images to a standard dimension and normalization. Subsequently, the data are partitioned into three subsets: training, validation, and testing. Augmentation techniques (such as flipping, shifting, shearing, and rotation) are applied to the training subset to improve generalization. Then, the proposed framework applies RFO to optimize the essential hyperparameters of the CNN. Consequently, the optimized CNN is trained and validated using the training and validation subsets. Finally, it is evaluated on the testing subset. Multiple metrics are used here, including accuracy, recall, precision, F1-Score, and Area Under the Curve (AUC).

So, the RFOCNN approach involves building a CNN structure. Then, its main hyperparameters are tuned by applying the RFO mechanism, which uses a fitness function (objective function) and its facilities to reach thr best possible CNN configuration.

B. CNN Principles

CNN is a specialized type of DL; it has been designed to preserve spatial analyses of the employed data [13]. The inputs to a CNN are typically multi-dimensional matrices of images. The image can be either grayscale or colored (Red, Green and Blue (RGB)).

Any CNN contains at least one convolutional layer. The convolutional layer is a core or essential layer in a CNN. It extracts features from the input images. It works by applying learnable filters that convolve across the inputs and produce feature maps. Each filter detects specific features such as edges, curves, and other features [28, 29]. Eq. (1) represents the mathematical formula for a single 2D convolution operation.

$$y_n^l = \sum_{m=1}^{M^{l-1}} \omega_{n,m}^l * x_m^l + b_n^l \quad (1)$$

where: y_n^l is the n-th output feature map, l is the current layer, $l-1$ is the previous layer, $\omega_{n,m}^l$ is the convolution kernel for the mth input and nth output, $*$ indicates the convolution operation, x_m^l is the mth of input feature map, and b_n^l is the bias [30].

After convolution, the Rectified Linear Unit (ReLU) is usually used. It is the most common activation function. It provides non-linearity to the CNN. It prevents the vanishing gradient problem by maintaining positive values and setting negative values to zero. Its equation can be expressed as the following Eq. (2):

$$f_{rel}(y_n^l) = \max(y_n^l, 0) \quad (2)$$

where: $f_{rel}(y_n^l)$ is the output of the ReLU activation function and max is the operation of maximum [31, 32]. It is worth mentioning that other activation functions can be used instead of ReLU, such as Leaky Rectified Linear Unit (Leaky ReLU), Exponential Linear Units (ELU) and Hyperbolic Tangent (Tanh). These functions are mathematically provided as follows:

$$f_{tan}(y_n^l) = \frac{e^{y_n^l} - e^{-y_n^l}}{e^{y_n^l} + e^{-y_n^l}} \quad (3)$$

where: $f_{tan}(y_n^l)$ is the output of the Tanh activation function, its value is between the range of $[-1, 1]$.

$$f_{lea}(y_n^l) = \begin{cases} y_n^l, \text{if } y_n^l > 0 \\ \partial y_n^l, \text{if } y_n^l \leq 0, \end{cases} \quad (4)$$

where: $f_{lea}(y_n^l)$ is the output of the Leaky ReLU activation function and ∂ is the leak factor, which is a small constant value.

$$f_{\text{elu}}(y_n^l) = \begin{cases} y_n^l, \text{if } y_n^l > 0 \\ \beta\left(e^{y_n^l} - 1\right), \text{ if } y_n^l \leq 0 \end{cases} \quad (5)$$

where: $f_{\text{elu}}(y_n^l)$ is the output of the ELU activation function and β is a non-negative hyperparameter which controls the saturation level of the ELU activation function for negative inputs [33].

The pooling layer is also utilized in CNN; it is applied to reduce the spatial dimensions of feature maps that are obtained from the previous layer. So, it generates other feature maps but with smaller sizes. It operates by extracting either the maximum or average value from a local region within each previous feature map [34, 35]. Eq. (6) shows the mathematical expression for max pooling.

$$p_n^l = \max(\mathrm{Z}^{l-1}) \quad (6)$$

where: p_n^l is an output of a max pooling layer and Z^{l-1} is a pooling window vector in previous layer [30].

A fully connected (Dense) layer is also employed in CNNs to connect the pooling layer to the next layer, where each of its neurons (or nodes) is linked to all neurons (or nodes) in the preceding layer. Eq. (7) fully reflects how the corresponding layer operates as follows:

$$Q_c^l = \sum_{hi=1}^{H} \sum_{wj=1}^{W} \sum_{ck=1}^{C} v_{c,(hi,wj,ck)}^l \cdot p_{(hi,wj,ck)}^{l-1} + b_c^l \quad (7)$$

where: Q_c^l is the output of a fully connected layer neuron, c is the neuron's index, H, W and C are respectively the height, width, and number of feature maps from the previous layer (pooling layer), $v_{c,(hi,wj,ck)}^l$ is the connected weight to the current layer, $p_{(hi,wj,ck)}^{l-1}$ is an input from the previous layer, after flattening the pooling layer (p_n^l), and b_c^l is the bias [36].

The final CNN layer can be for a sigmoid. It employs the sigmoid activation function for classification. It maps the outputs of the final neurons to a probability value in the range [0, 1]. The sigmoid function is defined as:

$$f_{sig}(Q_c^l) = \frac{1}{1+e^{-Q_c^l}} \quad (8)$$

where: $f_{sig}(Q_c^l)$ is the sigmoid activation function value, it is within the range [0, 1] [33].

CNN operates in three phases: training, validation, and testing. The training phase learns the network to provide desired outputs from certain inputs by using the backpropagation algorithm. The validation phase is used to monitor the training performance and also helps in preventing overfitting. The testing phase is employed to evaluate the network by exploiting data that has not been seen before [37].

C. RFO Fundamentals

RFO algorithm is a nature-inspired metaheuristic that models the hunting and social behaviors of red fox. Połap and Woźniak [38] from Silesian University of Technology formulated and developed the mathematical model of the RFO algorithm. It operates through three main stages: global searching for food, local searching represents approaching to prey, and population control for reproduction and migration.

In the 1st stage, foxes explore the territory to find the most promising areas of food. The herd shares information, and individual foxes move toward the best-performing fox. The algorithm calculates the square of Euclidean distance between the foxes and the best-performing fox for each iteration. The Euclidean distance equation is performed as follows:

$$d = \sqrt{\|((\bar{F}_i)^t - (\bar{F}_{best})^t)\|} \tag{9}$$

where: d is the Euclidean distance between two foxes, t is the number of iterations, $\bar{F}_i$ is the location of the ith fox in the considered population and $\bar{F}_{best}$ is the location of the best-performing fox in the same considered population [38].

By updating Eq. (9), foxes move toward the best-performing individual as follows:

$$(\bar{F}_i)^t = (\bar{F}_i)^t + \alpha \cdot \text{sign}((\bar{F}_{best})^t - (\bar{F}_i)^t) \tag{10}$$

where: α is a random scaling parameter which sets for each iteration to all individuals of foxes and sign is the sign function, which determines the direction of a fox toward the best-performing one [39].

The foxes return to their previous locations if the fitness value of the new location is not better than the previous one. This mimics the foxes returning after an unsuccessful exploration. The aforementioned procedures represent global searching implementation in each iteration.

In the 2nd stage, fox approaches cautiously when it observes the prey and begins circling maneuvers or deceptive tactics to minimize the distance to prey before attacking. The arbitrary value determines the probability of the fox's detection by the prey while approaching. The corresponding fox's movement is defined in the Eq. (11):

$$A = \begin{cases} \mu > 0.75, & \text{move closer} \\ \mu \leq 0.75, & \text{stay in the current position} \end{cases} \tag{11}$$

where: A is the fox's location coordinate and μ is arbitrary value ranging from 0 to 1, it sets per iteration to all foxes [40].

When the fox chooses to move, a mathematical representation is employed to simulate the movement of each individual. Fox's radius of observation relates on two parameters: a scaling factor $a \in (0, 0.2)$, which models distance varying to prey applied to all individuals per iteration, and the initial observation angle $\phi_0 \in (0, 2\pi)$, which is assigned to the entire population at the beginning of the algorithm. Eq. (12) refers to the radius of fox's vision (R).

$$R = \begin{cases} a\frac{sin(\phi_0)}{\phi_0}, \text{ if } \phi_0 \neq 0 \\ \theta, \text{ if } \phi_0 = 0 \end{cases} \tag{12}$$

where: $a\frac{sin(\phi_0)}{\phi_0}$ is the mathematical representation of simulating the movement of each individual and θ is the weather conditions factor, ranging between (0, 1) [41].

The movement of each individual is represented by Eq. (13) as a spatial coordinate:

$$SC = \begin{cases} F_0^{\text{new}} = aR \cdot cos(\phi_1) + F_0^{\text{actual}} \\ F_1^{\text{new}} = aR \cdot sin(\phi_1) + aR \cdot cos(\phi_2) + F_1^{\text{actual}} \\ F_2^{\text{new}} = aR \cdot sin(\phi_1) + aR \cdot sin(\phi_2) + aR \cdot cos(\phi_3) + F_2^{\text{actual}} \\ \vdots \\ F_{u-2}^{\text{new}} = aR \cdot \sum_{k=1}^{u-2} sin(\phi_k) + aR \cdot cos(\phi_{u-1}) + F_{u-2}^{\text{actual}} \\ F_{u-1}^{\text{new}} = aR \cdot sin(\phi_1) + aR \cdot sin(\phi_2) + \cdots + aR \cdot sin(\phi_{u-1}) + F_{u-1}^{\text{actual}} \end{cases} \tag{13}$$

where: SC is a spatial coordinate, u is the number of dimensions in the solution space, $F_0^{\text{new}}, F_1^{\text{new}}, \ldots, F_{u-2}^{\text{new}}, F_{u-1}^{\text{new}}$ are new position coordinates, $F_0^{\text{actual}}, F_1^{\text{actual}}, \ldots, F_{u-2}^{\text{actual}}, F_{u-1}^{\text{actual}}$ are the current position coordinates, and $\phi_1, \phi_2, \ldots, \phi_{u-1}$ are angular values, which are randomly selected from the interval $(0, 2\pi)$ [39].

In the 3rd stage, reproduction and migration are concentrated. The algorithm replaces the worst 5% of the population to maintain diversity and avoid local extrema. In order to keep the population number stable, new foxes replace existing ones based on a territorial strategy defined as the alpha couple. For each repetition, the replacement mechanism for foxes is applied according to the following definition:

$$B = \begin{cases} \text{New nomadic individual}, & \text{if } \kappa \geq 0.45 \\ \text{Reproduction of alpha couple}, & \text{if } \kappa < 0.45 \end{cases} \tag{14}$$

where: B is replacement mechanism of foxes, κ is arbitrary parameter, which takes values between 0 and 1 [41].

When $\kappa \geq 0.45$, a new nomadic fox is generated by randomly selecting a position within the search space and outside the habitat of the alpha couple. When $\kappa < 0.45$, the new individual is reproduced from the alpha couple, which represents the two best fitted foxes in the population. The habitat center of the alpha pair is defined as the center of their coordinates, while the habitat diameter is calculated as the Euclidean distance between them. These calculations are given in the following Eqs. (15)–(17):

$$(H_{cen})^t = \frac{(\bar{F}_{(1)})^t + (\bar{F}_{(2)})^t}{2} \tag{15}$$

$$(H_{dia})^t = \sqrt{\left\| (\bar{F}_{(1)})^t - (\bar{F}_{(2)})^t \right\|} \tag{16}$$

$$(F_{rep})^t = \kappa \frac{(\bar{F}_{(1)})^t + (\bar{F}_{(2)})^t}{2} \tag{17}$$

where: $(H_{cen})^t$ is habitat center of the alpha couple in each iteration, $(H_{dia})^t$ is the habitat diameter in each iteration, (F_{rep}) is the new individual and $\bar{F}_{(1)}$ and $\bar{F}_{(2)}$ is the locations of alpha couple in each iteration [38].

The foxes in the OA classification scenario represent the candidate solutions; the location of each one relates to a specific set of parameters within the searching space. The prey denotes the optimal solution, which is the combination of hyperparameters aiming to achieve the best fitness function value. RFO ensures quick convergence toward the global optimum solution by exploring a prey in a land, then exploiting a deception tactic during the hunt. Furthermore, population sustainability is maintained by replacing eliminated foxes through reproduction or herd departure.

D. Integrating between CNN and RFO

As previously mentioned, this paper proposes the RFOCNN framework. It integrates between RFO and CNN, where RFO is utilized to fine-tune essential CNN hyperparameters for classifying normal and OA knees. The RFO algorithm operates through three key stages: scanning for prey, deceptive hunting, and population replacement. These stages collectively guide the searching toward optimal hyperparameters configuration. Like other metaheuristic approaches, RFO evaluates potential solutions using an effective fitness function that can be customized according to the aims of the task, such as enhancing accuracy or reducing loss in a CNN model. This function evaluates the framework through iterations. In this study, validation accuracy is used as the fitness function, where the optimization here is looking to maximize the fitness function of the framework. The equation for validation accuracy can be given as follows:

$$Acc_{val} = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}} \tag{18}$$

where: Acc_{val} is the validation accuracy [42].

Effectively, the RFOCNN framework is proposed to optimize 17 hyperparameters of CNN by using RFO. Firstly, the CNN architecture is considered to have two convolutional layers with two activation functions, two pooling layers, a dense layer, and a classification output. The optimized hyperparameters include: the number of filters in the two convolutional layers, kernel sizes in the convolutional layers, kernel strides in the convolutional layers, activation function of the convolutional layers, window size in the two pooling layers, window strides in the pooling layers, number of neurons or nodes in the dense layer, activation function of the dense layer, dropout rates in the dense layer, learning rate and optimizer type. The optimization of these main hyperparameters can play a vital role in improving CNN's performance. The RFOCNN framework works in three phases: training, validation, and testing phases. The training phase enables the RFOCNN to learn labelled knee images and classify them into normal and OA. The validation phase is used to monitor the RFOCNN training performance to prevent overfitting. Also, the validation accuracy is considered here as a fitness function to guide the optimization following [42, 43]. The testing phase is employed to evaluate the framework on unseen data and provide the output predictions.

IV. Result and Discussion

A. Employed Dataset

In this study, the Digital Knee X-ray Images (DKXI) dataset of knee images is acquired from the Mendeley Data repository. It is a publicly available dataset of 1650 X-ray images which are collected from reputed local hospitals and health centers in India. These images have been acquired by using the X-ray machine of type (PROTEC PRS 500E). Each image is manually labeled by two medical experts. The knee region in any image has been automatically segmented according to a pixel density-based method. The dataset consists of 1464 segmented single-knee images with dimensions of 300 × 162 × 3 pixels and 186 segmented both-knees images with dimensions of 640 × 161 × 3 pixels. Each image has the format of Portable Network Graphics (PNG) [27].

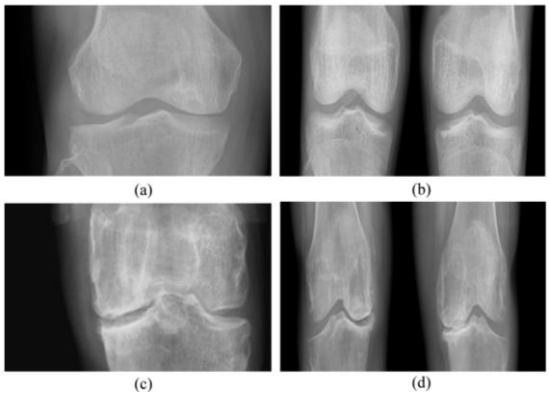


Fig. 3. Samples of segmented images from the employed DKXI dataset: (a) normal single-knee, (b) normal both-knees, (c) OA single-knee, (d) OA both-knees.

For the classification task, the images in the dataset are categorized into having normal knees or having OA knees following [19, 44, 45]. Fig. 3 shows examples of the segmented single-knee and both-knees images for the normal and OA categories.

B. Standardizations

To ensure consistency and reliability, images with disagreeing labels between the two experts have been excluded in this study. Furthermore, each image containing both knees has been split into two separate images, one for each knee. The number of samples has increased following preprocessing, with a total of 1820 images. Then, they are divided into 80% training, 10% validation, and 10% testing subsets. Any image in the subsets is resized to a uniform dimension of 128 × 128 × 3 pixels, and its pixel values are normalized to the range [0, 1]. The distribution of the employed dataset is presented in Table I.

The experiments in this work are carried out on an HP Victus laptop. Its specifications are: Intel Core i5 processor operating at 2.10 GHz and 8 GB of Random Access Memory (RAM). In addition of using Google Colab platform, which provides a Graphics Processing Unit (GPU) with memory of 40 GB, Python programming language (version: 3.12.12) and TensorFlow environment (version: 2.21.0).

As previously described, the proposed RFOCNN framework exploits RFO to optimize CNN hyperparameters. The considered CNN architecture here contains eight layers structured as: 1st convolutional layer with 1st activation function, 1st pooling layer, 2nd convolutional layer with 2nd activation function, 2nd pooling layer, dense layer, and classification output. RFO optimizes hyperparameters of these layers and other hyperparameters related to the training phase. These are essential 17 CNN hyperparameters and they are as follows: number of filters in the 1st convolutional layer, kernel size in the 1st convolutional layer, stride in the 1st convolutional layer, activation function in the 1st convolutional layer, window size in the 1st pooling layer, stride in the 1st pooling layer, number of filters in the 2nd convolutional layer, kernel size in the 2nd convolutional layer, stride in the 2nd convolutional layer, activation function in the 2nd convolutional layer, window size in the 2nd pooling layer, stride in the 2nd pooling layer, number of nodes in the dense layer, activation function in the dense layer, dropout rate in the dense layer, optimizer in the training phase and learning rate in the training phase.

TABLE I. THE DISTRIBUTION OF THE EMPLOYED DKXI DATASET IN THIS STUDY

Classes	Training Subset	Validation Subset	Testing Subset	Total
Normal knees	859 images	108 images	107 images	1074 images
OA knees	597 images	74 images	75 images	746 images
Both normal and OA knees	1456 images	182 images	182 images	1820 images

C. Pre-Processing

To enhance network performance and prevent overfitting, data augmentation techniques are applied to the training subset. The augmentation operations include: shearing using the factor of 0.2, rotation for up to 10 degrees in either direction, width and height shifting by the factor of 0.1, and horizontally flipping. Such transformations are employed to increase the diversity among input images and improve the framework's ability to generalize well.

D. Optimization

As mentioned, the proposed RFOCNN framework employs RFO to optimize CNN hyperparameters by using the validation accuracy equation as the fitness function. The applied RFO is set as: population size of 5 foxes, iterations of 10 with 65 epochs for each iteration, and early stopping patience of 9 epochs. Such a setting is empirically considered, where increasing some values cause out of memory. The progression of the best fitness accuracy is achieved as demonstrated in Fig. 4.

As can be seen from Fig. 4, the validation accuracies improved rapidly through the initial iterations and gradually stabilized in later iterations. The best validation fitness accuracy for the optimization is reached and fixed at approximately 0.90 (around 90%), meaning that the RFO is successfully convergence toward optimal values of the hyperparameters. The absence of providing more improvements in the final iterations indicating that the considered settings of 5 foxes and 10 iterations can be considered sufficient for optimizing the values of main CNN hyperparameters in the proposed RFOCNN framework. Table II summarizes the initial and determined values of the 17 CNN hyperparameters, which are reached by RFO.

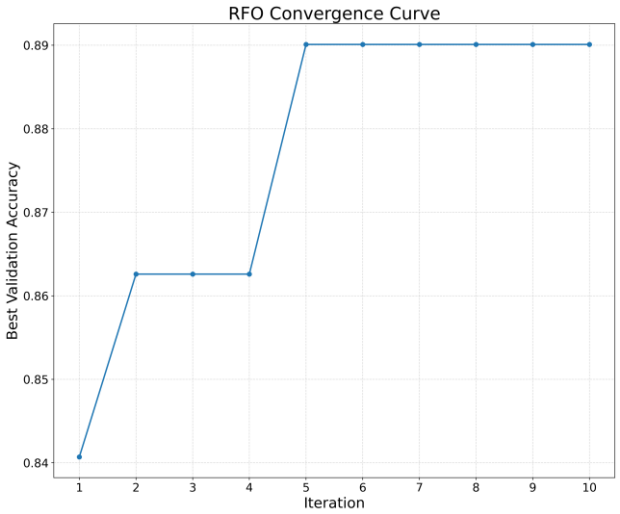


Fig. 4. The progression of the best validation fitness accuracy for the proposed RFOCNN framework.

The obtained values for the 1st convolutional layers are: number of filters equals to 86, kernel size equals to 3 × 3 pixels, stride equals to 1 pixel, and activation function equals to ReLU type. The attained values for the 1st pooling layer are: window size equals to 5×5 pixels, and stride equals to 2 pixels. The assigned values for the 2nd convolutional layer are: number of filters equals to 74, kernel size equals to 3 × 3 pixels, stride equals to 1 pixel and activation function equals to ELU type. The reached values for the 2nd pooling layer are: window size equals to 5 × 5 pixels, and stride equals to 1 pixel. The determined values for the dense layer are: number of nodes equals to 227 neurons, activation function equals to Tanh type and dropout rate equals to 0.24 (24%). The final values for the training hyperparameters are: optimizer equals to ADAM type and learning rate equals to 0.0001. Fig. 5 represents the DL construction of the proposed RFOCNN framework after optimizing the hyperparameters.

TABLE II. THE ESSENTIAL CNN HYPERPARAMETERS WHICH ARE CONSIDERED BY RFO, WITH THEIR INITIAL AND DETERMINED VALUES

Index	Hyperparameter	Initial Values or Types	Determined Values or Type
1	Number of filters in the 1st convolutional layer	3 to 128 filters	86 filters
2	Kernel size in the 1st convolutional layer	3 × 3, 4 × 4 or 5 × 5 pixels	3 × 3 pixels
3	Stride in the 1st convolutional layer	1 or 2 pixel(s)	1 pixel
4	Activation function in the 1st convolutional layer	ReLU, Tanh, ELU or Leaky ReLU type	ReLU type
5	Window size in the 1st pooling layer	2 × 2, 3 × 3, 4 × 4 or 5 × 5 pixels	5 × 5 pixels
6	Stride in the 1st pooling layer	1 or 2 pixel(s)	2 pixels
7	Number of filters in the 2nd convolutional layer	3 to 128 filters	74 filters
8	Kernel size in the 2nd convolutional layer	3 × 3, 4 × 4 or 5 × 5 pixels	3 × 3 pixels
9	Stride in the 2nd convolutional layer	1 or 2 pixel(s)	1 pixel
10	Activation function in the 2nd convolutional layer	ReLU, Tanh, ELU or Leaky ReLU type	ELU type
11	Window size in the 2nd pooling layer	2 × 2, 3 × 3, 4 × 4 or 5 × 5 pixels	5 × 5 pixels
12	Stride in the 2nd pooling layer	1 or 2 pixel(s)	1 pixel
13	Number of nodes in the dense layer	10 to 256 neurons	227 neurons
14	Activation function in the dense layer	ReLU, Tanh, ELU or Leaky ReLU type	Tanh type
15	Dropout rate in the dense layer	0.2 (20%) to 0.5 (50%)	0.24 (24%)
16	Optimizer in the training phase	ADAM, RMSprop or SGD type	ADAM type
17	Learning rate in the training phase	0.01, 0.001 or 0.0001	0.0001

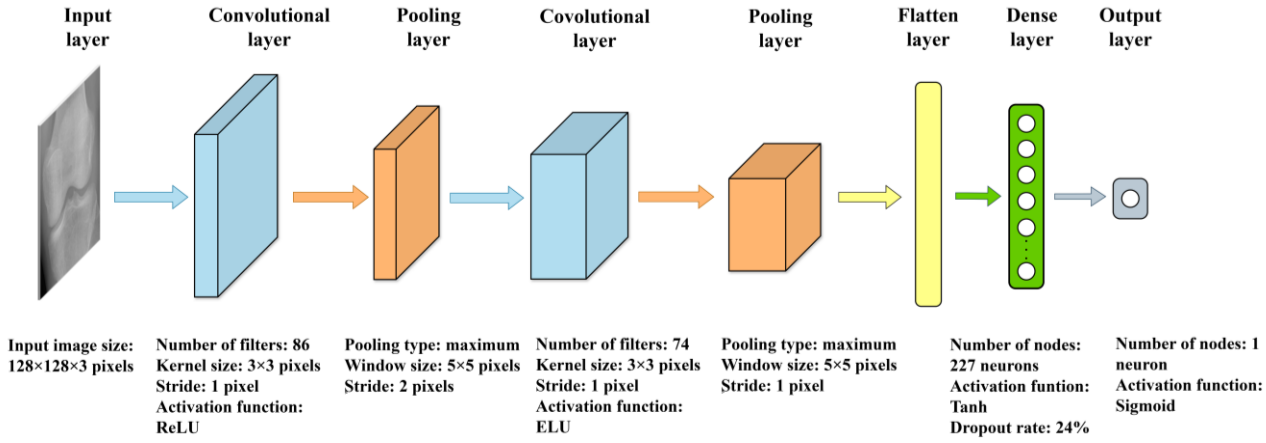


Fig. 5. The DL construction of the proposed RFOCNN framework with the determined values of hyperparameters.

E. Training and Validation

The RFOCNN framework has been compiled using the ADAM optimizer with a learning rate of 0.0001, as determined by the RFO. The framework is trained with a batch size of 32 for 200 epochs, while validation accuracy is monitored to assess training performance improvements.

As shown in Fig. 6(a), at the initial epochs, training accuracy rises rapidly, indicating that the framework can effectively learn patterns from the provided data. While validation accuracy follows a similar rising path, though with observable fluctuations, reflecting variability in the framework's generalization capability during early training. As training continues over epochs, both accuracy curves continue to improve, with training accuracy consistently moving closer to a high value and validation accuracy stays nearly aligned with training accuracy despite showing some oscillations. Both the training and validation accuracies converge in the range 91%–93% at the last epoch. This yields that the framework is successfully learned by using the optimal hyperparameters.

As given in Fig. 6(b), the training loss begins at a high value and decays rapidly toward lower values over a few epochs. Also, the validation loss has moved downward alongside the training loss, indicating that the framework can truly learn features from the employed images. For the remaining epochs, the training loss curve remains moving consistently toward the low values, while the validation loss curve continues moving relatively low at the same values, its curve fluctuates but remains within small values. Such outcomes for training and validation indicate that the framework generalizes well during these phases.

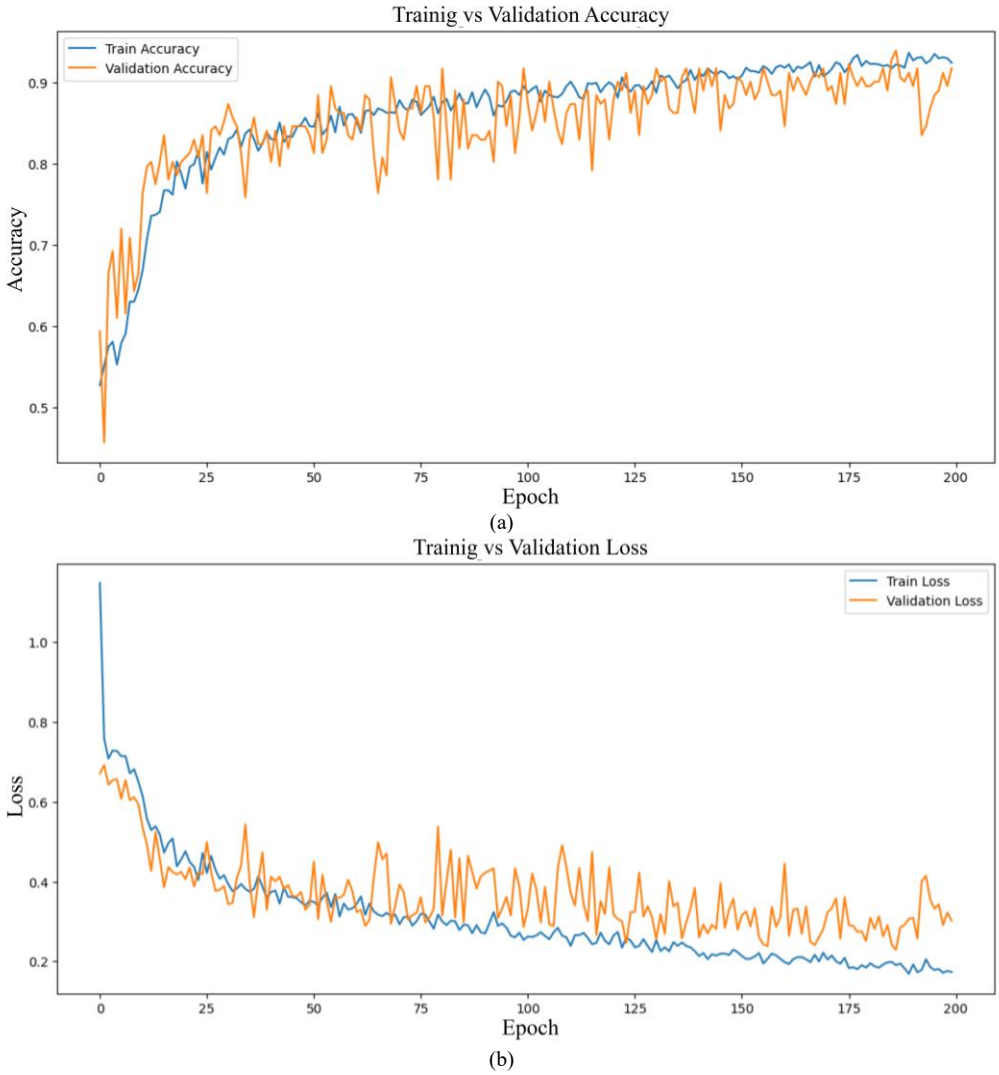


Fig. 6. The accuracy and loss curves of the training process: (a) curves of training and validation accuracy, (b) curves of training and validation loss.

F. Testing

The performance of the proposed RFOCNN framework is evaluated using the testing subset of the DKXI dataset. The confusion matrix presented in Fig. 7 summarizes the classification results in terms of True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN). The framework achieves 62 TP predictions by correctly identifying OA knees and 102 TN predictions by correctly recognizing normal knees. Only 5 FP cases are observed, where normal knees are incorrectly diagnosed as OA. While 13 FN cases corresponding to OA knees are incorrectly classified as normal. These results demonstrate that the proposed framework provides reliable classification performance for diagnosing OA knees, with an overall accuracy of 90%. Furthermore, the framework has acceptably classified images of normal knees by achieving a recall of 0.95 and precision of 0.89. For the OA class, the framework has also reasonably recognized images of OA knees by obtaining a recall of 0.83 and precision of 0.93. It also has attained comparably F1-Score values of 0.92 for the normal knees class and 0.87 for the OA knees class. Statistical measurements of testing the RFOCNN are given in Table III.

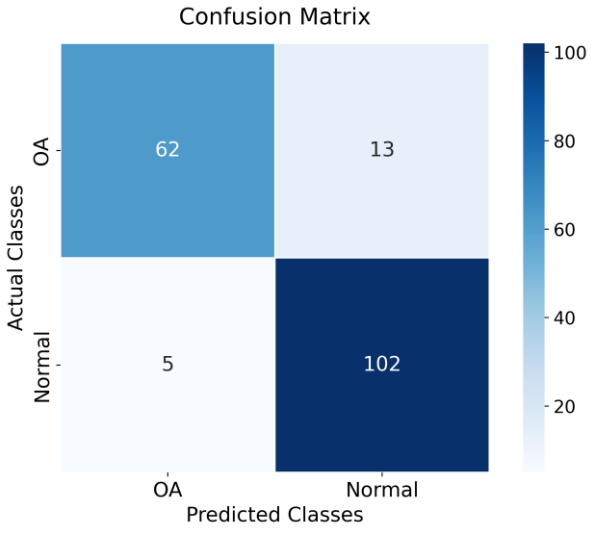


Fig. 7. Confusion matrix of the proposed RFOCNN on the testing subset.

TABLE III. STATISTICAL MEASUREMENTS OF TESTING THE RFOCNN

Class	Metric		
	Precision	Recall	F1-Score
OA	0.93	0.83	0.87
Normal	0.89	0.95	0.92

Fig. 8 illustrates the Receiver Operating Characteristic (ROC) curve of the proposed framework. The ROC curve lies above the diagonal reference line toward the upper left corner of the graph, indicating that the framework achieves a high TP rate with a relatively low FP rate. It obtains an AUC of 0.96, confirming that the RFOCNN framework can reliably discriminate between OA and normal knees. Such findings reveal the potential of the framework as an automated and dependable approach to diagnose OA of knees.

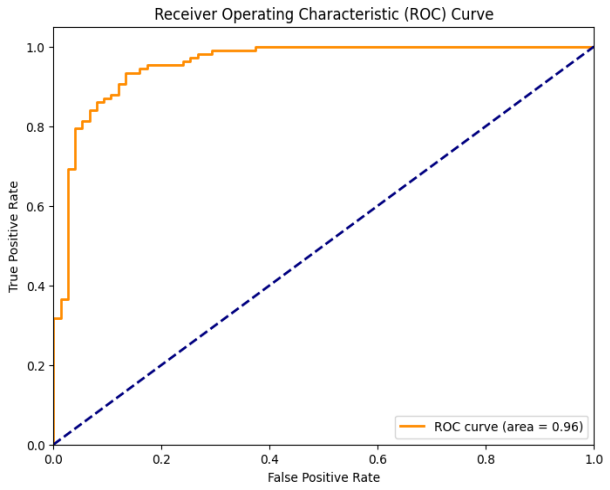


Fig. 8. ROC curve demonstrating discriminative performance of the RFOCNN framework.

The dataset exhibits a moderate class imbalance, consisting of 1074 images of normal knees and 746 images of OA knees (ratio of approximately 1.44:1). To address the class imbalance issue, class-weighted loss has been employed. It has been computed from the training subset using a balanced weighting strategy which incorporates with the loss function. The resulting class-weighted values are found to be 0.85 for the normal class and 1.22 for the OA class, thereby assigning a higher penalty to misclassified OA samples and improving the framework's sensitivity to the minority class. To evaluate the effectiveness of this strategy, additional experiments have been conducted on the proposed RFOCNN framework but with class-weighted loss. So, the difference in recall between both classes is decreased, indicating reduced bias toward the normal class. The overall classification accuracy has attained approximately 90%, which is so close to the benchmarked results without using the class-weighted loss. Therefore, the high classification performance of the proposed RFOCNN is also confirmed even with using more balanced data.

To improve generalizability, the framework is repeated using seven different random seeds (21, 42, 123, 246, 456, 789, and 1024). For each seed, the batch size of 32 and epochs of 200 are considered as training settings. Consequently, to ensure a robust evaluation, experiments are implemented using the seven different random seeds, and the final performances are reported by using the mean and Standard Deviation (SD) for each run. The proposed RFOCNN has achieved an overall average accuracy percentage of 90.5% ± 2.81%, overall recall percentages of 91.43% ± 3.41% for the normal class and 89.29% ± 5.44% for the OA class. The framework has also reached overall average precision percentages of 92.29 ± 3.73% for normal knees and 88.14 ± 4.49% for OA knees. It also attained comparably high F1-Score percentages of 92% ± 2.16% for the normal knee class and 88.57 ± 3.55% for the OA knee class. Generally low SD values can be observed across the results, demonstrating that the proposed framework produces reliable and reproducible results. These outcomes indicate that the obtained results of splitting the data into 80% training, 10% validation, and 10% testing are acceptable, as they are near to the performance of utilizing different random seeds. Furthermore, the results contribute to the robustness and standardization of the employed dataset.

G. Limitations

Despite the promising performance of the proposed RFOCNN framework, several limitations can be acknowledged. This study is conducted on only the DKXI dataset, comprising of 1820 knee X-ray images from Indian hospitals, which may limit the generalizability of the findings to other populations and imaging systems. Although classification between normal and OA knees is considered here, assessing the various KL grades, particularly for the early stage of OA, has not been considered, as they are challenging to identify. Moreover, interpretability and comparisons with expert radiologists have not been investigated either, as they require different explorations and assessment by medical specialists.

H. Comparisons

Comprehensive comparisons are performed on various DL models, baseline CNN with default hyperparameters and other hyperparameter optimization methods. All comparison experiments employed the DKXI dataset to ensure fair and consistent evaluation of the proposed RFOCNN as follows:

1) Comparison with DL models

Additional experiments are performed on various DL models with the same dataset in order to compare with the proposed RFOCNN. The compared models include: VGG-19, ResNet-50, XceptionNet and Neural Architecture Search Network-Mobile (NASNet-Mobile). Table IV summarizes the classification performance of the proposed RFOCNN with the compared DL models.

TABLE IV. PERFORMANCE COMPARISON OF THE PROPOSED RFOCNN WITH VARIOUS DL MODELS

Model	Accuracy (%)
ResNet-50	64
XceptionNet	85
NASNet-Mobile	86
VGG-19	87
The proposed RFOCNN	**90**

In this table, it can be seen that ResNet-50 attains the lowest accuracy of 64%. This is due to its residual block, which failed to learn relevant knee features. XceptionNet, NASNet-Mobile and VGG-19 achieve close results of 85%, 86%, and 87%, respectively. They are able to adapt their filters to capture the features of knee regions relating to OA detection. However, they still could not obtain excellent results. It is obvious that the proposed RFOCNN benefits from optimizing the main hyperparameters, as its evaluation accuracy is benchmarked to 90%. Whereas the compared DL models are evaluated using the same training settings as the proposed RFOCNN framework, but without optimal values for the hyperparameters, as each model possesses distinct architectural characteristics and optimization requirements. Such limitations in the compared standard DL models affect their performance, and at the same time approve the importance of reaching and using the optimal values of the hyperparameters.

2) Comparison with CNN models

To evaluate the effectiveness of our RFOCNN approach, comparisons are established with a baseline CNN that has the same architecture but default hyperparameters. This baseline model has reached 86% testing accuracy. In addition, another comparison has been made to assess the contribution of the proposed RFOCNN against a CNN tuned with the random search method. The tuned CNN has attained an accuracy of 89%. Given that the proposed RFOCNN has achieved the highest accuracy of 90%, it can be inferred that its guided optimization strategy of using the RFO is more effective in reaching best values for the CNN hyperparameters than other compared tuning methods.

3) Comparison with other optimization methods

The RFO in the proposed RFOCNN has been compared with two well-known metaheuristic algorithms, namely: Particle Swarm Optimization (PSO) and Equilibrium Optimization (EO). So, EO and PSO are fairly applied to optimize the values of CNN hyperparameters instead of the RFO. The obtained results are very interesting as the EO and PSO based CNNs have achieved testing accuracies of 87% and 88%, respectively. It can be noticed that the proposed RFOCNN outperforms them, as it attains a higher accuracy of 90%. Such comparison approves the effectiveness of the RFO in well optimizing the values of CNN hyperparameters. The improved performance of the RFOCNN framework can be explained by the unique optimization mechanism of the RFO algorithm. Unlike the compared optimization methods that primarily rely on balancing between global exploration and local exploitation, RFO integrates a population replacement approach with its exploration and exploitation stages. The replacement of the worst-performing individuals maintains population diversity and reduces the risk of premature convergence to local optima. Consequently, the RFO identifies more effective CNN hyperparameter configurations, resulting in improved classification accuracy compared to the other optimization methods.

V. Conclusion

This study introduced the RFOCNN framework, which optimizes essential CNN hyperparameters by RFO for the case of classifying between normal and OA knees. The optimized hyperparameters related to the CNN layers and training phase include the number of filters, kernel size, kernel stride, activation function, window size, window stride, number of neurons, optimizer and learning rate. DKXI dataset has been utilized for the optimization, training, validation and evaluation of the proposed RFOCNN framework. The findings highlight that optimizing CNN hyperparameters using RFO plays a critical role in attaining high performance.

The proposed RFOCNN framework has achieved a significant testing accuracy of 90%. In addition to a high AUC value of 0.96, indicating that the framework can accurately categorize normal and OA classes. The classification results have reported high F1-Score values of 0.92 for the normal class and 0.87 for the OA class. The framework has also exhibited acceptable achievements by obtaining a recall of 0.83 and a precision of 0.93 for the OA cases, and a recall of 0.95 and a precision of 0.89 for the cases of normal knees. The results of comparisons between the RFOCNN framework and several DL models, baseline CNN, a CNN tuned with the random search method, and CNNs optimized with other metaheuristic algorithms revealed that the RFOCNN outperformed of the compared methods. The proposed framework demonstrates clear potential as an efficient and effective framework for reliable OA diagnosis.

Future studies should consider providing transparent reports for patients. Also, future work should focus on validating the RFOCNN framework using external multi-center datasets from different populations and imaging systems to assess its generalizability. Further studies are suggested for extending the framework to analyze explicit KL grades classification, with particular emphasis on the early stages of OA. It will be valuable to utilize XAI techniques such as Grad-CAM and attention-based visualization methods to investigate the framework's interpretability. It will also be valuable to provide clinically grounded comparisons to radiologists and patient outcomes, as this will enhance the clinical applicability in real-world diagnosis.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

Raid Rafi Omar Al-Nima proposed the RFOCNN framework, analyzed the results, provided guidance throughout the research and assisted in refining the research findings. Abdullah Dawood Sulaiman conducted the experiments and implemented the code, analyzed the results, and wrote the manuscript. Tingting Han approved the analytical methods, provided guidance throughout the research and assisted in refining the research findings, and critically revised the manuscript. All authors had approved the final version.

REFERENCES

[1] M. Doherty, J. W. J. Bijlsma, N. Arden, D. Hunter, and N. Dalbeth, *Oxford Textbook of Osteoarthritis and Crystal Arthropathy*. Oxford University Press, 2016.

[2] World Health Organization. (2026). Osteoarthritis. [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/osteoarthritis

[3] D. Prieto-Alhambra, N. Arden, and D. J. Hunter, *Osteoarthritis: the Facts*. OUP Oxford, 2014.

[4] T. Vos *et al.*, "Global burden of 369 diseases and injuries in 204 countries and territories, 1990–2019: A systematic analysis for the global burden of disease study 2019," *The Lancet*, vol. 396, no. 10258, pp. 1204–1222, Oct. 2020. doi: 10.1016/S0140-6736(20)30925-9

[5] A. Cieza, K. Causey, K. Kamenov, S. W. Hanson, S. Chatterji, and T. Vos, "Global estimates of the need for rehabilitation based on the global burden of disease study 2019: A systematic analysis for the global burden of disease study 2019," *The Lancet*, vol. 396, no. 10267, pp. 2006–2017, 2020. doi: 10.1016/S0140-6736(20)32340-0

[6] M. Bañuls-Mirete, A. F. Lombardi, A. I. B. Posis, A. H. Shadyab, E. Y. Chang, N. E. Lane, and M. Guma, "Effusion-synovitis worsening mediates the association between body mass index and Kellgren-Lawrence progression in obese individuals: Data from the osteoarthritis initiative," *Osteoarthritis and Cartilage*, vol. 30, no. 9, pp. 1278–1286, Sep. 2022. doi: 10.1016/j.joca.2022.05.011

[7] H. M. Lin, P. S. Hsieh, N. C. Chen, C. H. Tsai, W. F. Kuo, Y. L. Lee, and K. C. Hung, "Impact of cognitive behavior therapy on osteoarthritis-associated pain, insomnia, depression, fatigue, and physical function in patients with knee/hip osteoarthritis: A systematic review and meta-analysis of randomized controlled trials," *Frontiers in Medicine*, vol. 9, 1083095, Jan. 2023. doi: 10.3389/fmed.2022.1083095

[8] H. Long, Q. Liu, H. Yin, K. Wang, N. Diao, Y. Zhang, J. Lin, and A. Guo, "Prevalence Trends of Site-Specific Osteoarthritis From 1990 to 2019: Findings From the Global Burden of Disease Study 2019," *Arthritis & Rheumatology*, vol. 74, no. 7, pp. 1172–1183, Jul. 2022. doi: 10.1002/art.42089

[9] N. Arden, F. Blanco, C. Cooper, A. Guermazi, D. Hayashi, D. Hunter, M. K. Javaid, F. Rannou, F. Roemer, and J.-Y. Reginster. (2018). *Atlas of Osteoarthritis*, 2nd ed. London,UK, Springer Healthcare Ltd. [Online]. Available: https://www.esceo.org/sites/esceo/files/pdf/OA%20ESCEO%20Atlas%20of%20Osteoarthritis%202nd%20Edition%202018.pdf

[10] E. Vaienti, G. Scita, F. Ceccarelli, and F. Pogliacomi, "Understanding the human knee and its relationship to total knee replacement," *Acta Biomedica*, vol. 88, pp. 6–16, 2017. doi: 10.23750/abm.v88i2-S.6507

[11] S. M. Ahmed and R. J. Mstafa, "Identifying Severity Grading of Knee Osteoarthritis from X-ray Images Using an Efficient Mixture of Deep Learning and Machine Learning Models," *Diagnostics*, vol. 12, no. 12, 2939, 2022. doi: 10.3390/diagnostics12122939

[12] M. W. Brejnebøl *et al.*, "Interobserver Agreement and Performance of Concurrent AI Assistance for Radiographic Evaluation of Knee Osteoarthritis," *Radiology*, vol. 312, no. 1, e233341, Jul. 2024. doi: 10.1148/radiol.233341

[13] H. A. Ahmed and E. A. Mohammed, "Using Artificial Intelligence to classify osteoarthritis in the knee joint: Review," *NTU Journal of Engineering and Technology*, vol. 1, no. 3, pp. 31–40, 2022. doi: 10.56286/ntujet.v1i3.155

[14] J. Thomson, T. O'Neill, D. Felson, and T. Cootes, "Automated shape and texture analysis for detection of osteoarthritis from radiographs of the knee," in *Proc. International Conference on Medical Image Computing and Computer-Assisted Intervention*, Springer, 2015, pp. 127–134. doi: 10.1007/978-3-319-24571-3_16

[15] J. Antony, K. McGuinness, N. E. O'Connor, and K. Moran, "Quantifying radiographic knee osteoarthritis severity using deep convolutional neural networks," in *Proc. 2016 23rd International Conference on Pattern Recognition (ICPR)*, Cancun, Mexico: IEEE, Dec. 2016, pp. 1195–1200. doi: 10.1109/ICPR.2016.7899799

[16] A. Tiulpin, J. Thevenot, E. Rahtu, P. Lehenkari, and S. Saarakkala, "Automatic knee osteoarthritis diagnosis from plain radiographs: A deep learning-based approach," *Scientific Reports*, vol. 8, no. 1, 1727, 2018. doi: 10.1038/s41598-018-20132-7

[17] J. Lim, J. Kim, and S. Cheon, "A deep neural network-based method for early detection of osteoarthritis using statistical data," *International Journal of Environmental Research and Public Health*, vol. 16, no. 7, 1281, 2019. doi: 10.3390/ijerph16071281

[18] K. A. Thomas, Ł. Kidziński, E. Halilaj, S. L. Fleming, G. R. Venkataraman, E. H. G. Oei, G. E. Gold, and S. L. Delp, "Automated classification of radiographic knee osteoarthritis severity using deep neural networks," *Radiology: Artificial Intelligence*, vol. 2, no. 2, e190065, 2020. doi: 10.1148/ryai.2020190065

[19] C. Guida, M. Zhang, and J. Shan, "Knee osteoarthritis classification using 3D CNN and MRI," *Applied Sciences*, vol. 11, no. 11, 5196, 2021. doi: 10.3390/app11115196

[20] C. Chen, P. Liu, Y. Feng, D. Ye, C.-C. Fu, L. Ye, Y. Song, D. Liu, G. Zheng, and C. Zhang, "Diagnostic performance for severity grading of hip osteoarthritis and osteonecrosis of femoral head on radiographs: Deep learning model vs. board-certified orthopaedic surgeons," *Osteoarthritis Imaging*, vol. 3, no. 2, 100092, 2023. doi: 10.1016/j.ostima.2023.100092

[21] Z. Wang, A. Chetouani, and R. Jennane, "Transformer with selective shuffled position embedding for early detection of knee osteoarthritis," in *Proc. 20th International Conference on Content-based Multimedia Indexing (CBMI 2023)*, Orléans, France, New York, NY, USA,: ACM, Sep. 2023, pp. 261–264. doi: 10.1145/3617233.3617250

[22] S. W. Pi, B. D. Lee, M. S. Lee, and H. J. Lee, "Ensemble deep-learning networks for automated osteoarthritis grading in knee X-ray images," *Scientific Reports*, vol. 13, no. 1, pp. 1–17, 2023. doi: 10.1038/s41598-023-50210-4

[23] X. Y. Teh, P. S. Q. Yeoh, T. Wang, X. Wu, K. Hasikin, and K. W. Lai, "Knee osteoarthritis diagnosis with unimodal and multi-modal neural networks: Data from the osteoarthritis initiative," *IEEE Access*, vol. 12, pp. 146698–146717, 2024. doi: 10.1109/ACCESS.2024.3472654

[24] R. Ahmed and A. S. Imran, "Knee Osteoarthritis Analysis Using Deep Learning and XAI on X-Rays," *IEEE Access*, vol. 12, pp. 68870–68879, 2024. doi: 10.1109/ACCESS.2024.3400987

[25] T. Mehdi, Y. Nasser, S. Aloui, and R. Jennane, "Biomedical signal processing and control deep learning based approach combining shape and texture features for knee osteoarthritis prediction from X-ray images," *Biomedical Signal Processing and Control*, vol. 101, 107172, 2025. doi: 10.1016/j.bspc.2024.107172

[26] P. Panwar, S. Chaurasia, J. Gangrade, and A. Bilandi, "Early diagnosis of knee osteoarthritis severity using vision transformer," *BMC Musculoskeletal Disorders*, vol. 26, no. 1, 884, 2025. doi: 10.1186/s12891-025-09137-2

[27] S. Gornale and P. Patravali. (2026). Digital knee x-ray images. *Mendeley Data*. [Online]. Available: https://data.mendeley.com/datasets/t9ndx37v5h/1

[28] J. Amara, P. Kaur, M. Owonibi, and B. Bouaziz, "Convolutional neural network based chart image classification," *Computer Science Research Notes*, vol. 2703, pp. 83–88, 2017.

[29] H. M. Ahmed and M. Y. Kashmola, "Performance Improvement of Convolutional Neural Network Architectures for Skin Disease Detection," *International Journal of Computing and Digital Systems*, vol. 13, no. 1, pp. 657–669, 2023. doi: 10.12785/ijcds/130152

[30] A. M. Ibrahim, A. K. Eesee, and R. R. O. Al-Nima, "Deep fingerprint classification network," *Telkomnika (Telecommunication Computing Electronics and Control)*, vol. 19, no. 3, pp. 893–901, 2021. doi: 10.12928/TELKOMNIKA.v19i3.18771

[31] M. A. Wani, F. A. Bhat, S. Afzal, and A. I. Khan, "Basics of supervised deep learning," in *Advances in Deep Learning*, Springer, 2019, pp. 13–29.

[32] A. R. H. Al-Jader, N. S. M. Sheet, M. N. Farhan, and R. R. O. Al-Nima, "Normalized Deep Learning Model for Classifying White Blood Cell," *Journal of Image and Graphics*, vol. 14, no. 2, pp. 259–266, 2026. doi: 10.18178/joig.14.2.259-266

[33] S. Khan, H. Rahmani, S. A. A. Shah, and M. Bennamoun, "A Guide to Convolutional Neural Networks for Computer Vision," in *Synthesis Lectures on Computer Vision*, Morgan & Claypool, 2018. doi: 10.2200/S00822ED1V01Y201712COV015

[34] F. Bieder, R. Sandkühler, and P. C. Cattin, "Comparison of methods generalizing max-and average-pooling," arXiv preprint arXiv:2103.01746, 2021.

[35] S. O. Ali, R. R. O. Al-Nima, and E. A. Mohammed, "Individual

recognition with deep earprint learning," in *Proc. International Conference on Communication and Information Technology, ICICT 2021*, Basra, 2021, pp. 304–309. doi: 10.1109/ICICT52195.2021.9568410

[36] Y. Shen, Y. Yin, C. Zhao, B. Li, J. Wang, G. Li, and Z. Zhang, "Image recognition method based on an improved convolutional neural network to detect impurities in wheat," *IEEE Access*, vol. 7, pp. 162206–162218, 2019. doi: 10.1109/ACCESS.2019.2946589

[37] M. L. Chan, I. S. Heng, and C. Messenger, "Detection and classification of supernova gravitational wave signals: A deep learning approach," *Physical Review D*, vol. 102, no. 4, 043022, Aug. 2020. doi: 10.1103/PhysRevD.102.043022

[38] D. Połap and M. Woźniak, "Red fox optimization algorithm," *Expert Systems with Applications*, vol. 166, 114107, Mar. 2021. doi: 10.1016/j.eswa.2020.114107

[39] R. Natarajan, G. Megharaj, A. Marchewka, P. B. Divakarachari, and M. R. Hans, "Energy and distance based multi-objective red fox optimization algorithm in wireless sensor network," *Sensors*, vol. 22, no. 10, 3761, 2022. doi: 10.3390/s22103761

[40] A. Jaszcz, D. Połap, and R. Damaševičius, "Lung X-Ray image segmentation using heuristic red fox optimization algorithm," *Scientific Programming*, vol. 2022, no. 1, 4494139, 2022. doi: 10.1155/2022/4494139

[41] E. Khorami, F. Mahdi Babaei, and A. Azadeh, "Optimal diagnosis of COVID-19 based on convolutional neural network and red fox optimization algorithm," *Computational Intelligence and Neuroscience*, vol. 2021, no. 1, 4454507, Jan. 2021. doi: 10.1155/2021/4454507

[42] N. Sadiq Jaafar Al-Bayati and R. Rafi Omar Al-Nima, "Classification of alzheimer's disease using a hybrid technique integration between cnn and optuna optimization," *NTU Journal of Engineering and Technology*, vol. 5, no. 1, pp. 159–169, Mar. 2026. doi: 10.56286/49fsy824

[43] A. K. Abdullah, R. Rafi Omar Al-Nima, and K. G. Majeed, "Hybrid framework of whale optimization and convelution neural network for osteoporosis diagnosis," in *Proc. 2025 3rd International Conference on Cyber Resilience (ICCR)*, IEEE, Jul. 2025, pp. 1–8. doi: 10.1109/iccr67387.2025.11292020

[44] M. Azad and T. Rahman Anik, "Automated diagnosis of knee osteoarthritis: A stacked ensemble deep learning approach with explainable AI techniques," *IEEE Access*, vol. 13, pp. 194862–194883, 2025. doi: 10.1109/ACCESS.2025.3631513

[45] G. W. Stachowiak, M. Wolski, T. Woloszynski, and P. Podsiadlo, "Detection and prediction of osteoarthritis in knee and hand joints based on the X-ray image analysis," *Biosurface and Biotribology*, vol. 2, no. 4, pp. 162–172, 2016. doi: 10.1016/j.bsbt.2016.11.004

Copyright © 2026 by the authors. This is an open access article distributed under the Creative Commons Attribution License which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited (CC BY 4.0).