diagnostics

MDPI

Article

An Al-Biruni Earth Radius Optimization-Based Deep Convolutional Neural Network for Classifying Monkeypox Disease

Doaa Sami Khafaga [1,*], Abdelhameed Ibrahim [2], El-Sayed M. El-Kenawy [3,*], Abdelaziz A. Abdelhamid [4,5,*], Faten Khalid Karim [1,*], Seyedali Mirjalili [6,7], Nima Khodadadi [8], Wei Hong Lim [9], Marwa M. Eid [10] and Mohamed E. Ghoneim [11,12]

[1] Department of Computer Sciences, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, Riyadh 11671, Saudi Arabia
[2] Computer Engineering and Control Systems Department, Faculty of Engineering, Mansoura University, Mansoura 35516, Egypt
[3] Department of Communications and Electronics, Delta Higher Institute of Engineering and Technology, Mansoura 35111, Egypt
[4] Department of Computer Science, College of Computing and Information Technology, Shaqra University, Shaqra 11961, Saudi Arabia
[5] Department of Computer Science, Faculty of Computer and Information Sciences, Ain Shams University, Cairo 11566, Egypt
[6] Centre for Artificial Intelligence Research and Optimization, Torrens University Australia, Fortitude Valley, QLD 4006, Australia
[7] Yonsei Frontier Lab, Yonsei University, Seoul 03722, Republic of Korea
[8] Department of Civil and Environmental Engineering, Florida International University, Miami, FL 33199, USA
[9] Faculty of Engineering, Technology and Built Environment, UCSI University, Kuala Lumpur 56000, Malaysia
[10] Faculty of Artificial Intelligence, Delta University for Science and Technology, Mansoura 35712, Egypt
[11] Department of Mathematical Sciences, Faculty of Applied Science, Umm Al-Qura University, Makkah 21955, Saudi Arabia
[12] Faculty of Computers and Artificial Intelligence, Damietta University, Damietta 34511, Egypt
* Correspondence: dskhafga@pnu.edu.sa (D.S.K.); skenawy@ieee.org (E.-S.M.E.-K.); abdelaziz@su.edu.sa (A.A.A.); fkdiaaldin@pnu.edu.sa (F.K.K.)

Abstract: Human skin diseases have become increasingly prevalent in recent decades, with millions of individuals in developed countries experiencing monkeypox. Such conditions often carry less obvious but no less devastating risks, including increased vulnerability to monkeypox, cancer, and low self-esteem. Due to the low visual resolution of monkeypox disease images, medical specialists with high-level tools are typically required for a proper diagnosis. The manual diagnosis of monkeypox disease is subjective, time-consuming, and labor-intensive. Therefore, it is necessary to create a computer-aided approach for the automated diagnosis of monkeypox disease. Most research articles on monkeypox disease relied on convolutional neural networks (CNNs) and using classical loss functions, allowing them to pick up discriminative elements in monkeypox images. To enhance this, a novel framework using Al-Biruni Earth radius (BER) optimization-based stochastic fractal search (BERSFS) is proposed to fine-tune the deep CNN layers for classifying monkeypox disease from images. As a first step in the proposed approach, we use deep CNN-based models to learn the embedding of input images in Euclidean space. In the second step, we use an optimized classification model based on the triplet loss function to calculate the distance between pairs of images in Euclidean space and learn features that may be used to distinguish between different cases, including monkeypox cases. The proposed approach uses images of human skin diseases obtained from an African hospital. The experimental results of the study demonstrate the proposed framework's efficacy, as it outperforms numerous examples of prior research on skin disease problems. On the other hand, statistical experiments with Wilcoxon and analysis of variance (ANOVA) tests are conducted to evaluate the proposed approach in terms of effectiveness and stability. The recorded results confirm the superiority of the proposed method when compared with other optimization algorithms and machine learning models.

check for updates

Citation: Khafaga, D.S.; Ibrahim, A.; El-Kenawy, E.-S.M.; Abdelhamid, A.A.; Karim, F.K.; Mirjalili, S.; Khodadadi, N.; Lim, W.H.; Eid, M.M.; Ghoneim, M.E. An Al-Biruni Earth Radius Optimization-Based Deep Convolutional Neural Network for Classifying Monkeypox Disease. *Diagnostics* **2022**, *12*, 2892. https://doi.org/10.3390/diagnostics12112892

Academic Editors: Sameer Antani, Azhar Zam and Esa Prakasa

Received: 18 October 2022
Accepted: 18 November 2022
Published: 21 November 2022

Keywords: skin disease; monkeypox infection; Al-Biruni Earth radius; deep learning; meta-heuristic; optimization

1. Introduction

Many applications reap the advance of artificial intelligence (AI) and deep learning (DL) methodologies to ease and facilitate the daily lives of humans. Recently, computer systems based on AI reached accuracy up to par with human experts. Applications of AI are spread over a broad spectrum of fields, such as banking transactions, financial affairs, manufacturing, retail, the supply chain, e-commerce, and healthcare [1]. AI has been an important role player in many new business innovations within the technology industry. Its applications include web search engines, video-on-demand recommendations, products of interest, targeted advertising, and self-driving vehicles [2]. The manuscript introduces AI uses in medical science and healthcare services. The most prevalent rule of AI in medical services is accurately detecting any medical issues and attracting the attention of medical teams, enabling them to respond to the threat and thus saving lives.

The application of AI in medical science and health care is spreading rapidly. Common domains of applications in medical image analysis include different fields such as organ localization and abnormality detection [3,4], gene mutation detection [5], and cancer grading [6] and staging [7], and the domains are too numerous to mention and still flourishing. Remarkably, AI methods have recently played a significant role in the COVID-19 pandemic as a tool for diagnosis and severity ranking from medical images [8–10]. This success motivates scientists to recruit AI for the emerging threat of a monkeypox outbreak.

Poxviruses are one of the most known vertebrate infectious viruses, including cowpox, vaccinia, variola (smallpox), and monkeypox viruses [11,12]. Human monkeypox is a zoonotic Orthopoxvirus with symptoms similar to poxviruses, including cowpox, vaccinia, and variola (smallpox). The first case report of human infection by the monkeypox virus was in the Democratic Republic of the Congo in 1970. Monkeypox has been considered the most dangerous Orthopoxvirus for human health since that time. In the past, monkeypox was most frequent in the west African countries. However, it has often been reported in urban areas outside the African continent lately [13].

The multinational monkeypox outbreak has choked the world while recovering from the aftershock of the COVID-19 pandemic. As of the end of June 2022, an abrupt and significant surge in the number of cases close to over 4900 human monkeypox instances has occurred which are non-endemic for the disease, spanning across the Western Hemisphere. Transmission has extended beyond cases linked to an initial exposure in Africa to human-to-human spread in each country. Virologists differentiate two variants of the monkeypox virus: the Central Africa clade and the West Africa clade. There is no proper treatment for the monkeypox virus to date [14,15]. Although the mortality rates of monkeypox are considered low [16], early detection empowers precautions with anyone who seems contagious for patient isolation and contact tracing for effective containment of the disease's spread.

Clinical identification of monkeypox from different pox viruses is difficult. The diagnosis procedure starts with visual observation of skin lesions and the existing history of exposure, followed by testing skin lesions using dermatoscopic images and then confirmation using a polymerase chain reaction (PCR) [17]. This work will be directed to utilizing AI approaches to diagnose monkeypox with a digital skin image classification algorithm. AI techniques based on artificial neural networks (ANNs) could be useful in detecting monkeypox with the help of skin image processing and analysis. Numerous ANN techniques are applied in the diagnosis of skin lesions due to their unique characteristics of learning crucial features from complex data throughout the training process.

Classification models assist physicians in making more accurate diagnoses of skin cancer, skin lesions, and psoriasis [9]. Deep convolutional neural networks (CNNs) [10] have had their capability for general and highly variable tasks across many specific categories revealed [18,19]. Estevan et al. [20] used a deep convolutional neural network (DCNN) to train a model using 129,450 images for binary classification of images into either malignant melanoma or benign tissue. Furthermore, they concluded that the DCNN achieved performance comparable to that of 21 board-certified specialists. In their work, Haenssle et al. [21] compared a CNN's diagnostic performance against a group of 58 physicians, including 30 experts in the field of dermatology. They trained Google's Inception v4 CNN architecture and tested its validity using dermatoscopic images and corresponding diagnoses. The CNN's performance surpassed most dermatologists. Hekler et al. [22] introduced multiclass classification models combining humans and AI to classify images of suspected skin cancer. Combining natural and artificial intelligence achieved superior accuracy compared with the accuracy achieved by artificial or human intelligence alone.

All proposed solutions for lesion detection problems lay on selecting the optimal parameters of the employed model. The selection process is often performed in two phases, namely mathematical modeling and optimization using an optimization method. Optimization is selecting optimal solutions from a pool of competing explanations for the problem under consideration. In general, a multi-dimensional search problem is turned into an optimization problem. In practice, this entails reducing or increasing the quality of the candidate solutions, often represented as a vector in the search space. In recent years, metaheuristic algorithms have been extensively applied among the optimization algorithms for solving classification problems [23]. Their ability to deal with complex, multi-dimensional, and ill-behaved optimization problems and provide satisfying results in a reasonable time has made them the choice of most researchers and practitioners [24,25].

Numerous types of AI have been proposed for use in different applications [26,27]. Combining the attributes of various separate base models creates an ensemble prediction model using the ensemble technique. There are many different methods to put this concept into practice. Resampling the training set is one of the more efficient methods, while others employ other prediction algorithms, adjust different predictive strategy parameters, etc. In order to aggregate the outcomes of each prediction, an ensemble of techniques is utilized [28,29]. Al-Biruni Earth radius (BER) optimization was first proposed recently in [30,31]. When more variables are used in the optimization process, the BER optimization technique performs worse. However, the successful balancing of exploration and exploitation is a considerable benefit. The suggested approach utilizes this benefit by utilizing the BER algorithm. Stochastic fractal search (SFS) [32] has limitations such as a low exploration rate and performance falling when a large number of local optimum solutions are present, despite its simplicity of usage and excellent balance between exploration and exploitation. The BERSFS algorithm is implemented in this study to use the method's benefits and get around its limitations.

We employ a metaheuristic algorithm for tuning the parameters of the neural network. A novel framework based on Al-Biruni Earth radius (BER) optimization fine-tunes the deep convolutional neural network (DCNN) layers. As a first step in the proposed approach, we use deep CNN-based models to learn the embedding of input images in Euclidean space. In the second step, we use an optimized classification model based on the triplet loss function to calculate the distance between pairs of images in Euclidean space to learn features that may be used to distinguish between different cases, including monkeypox cases. The proposed approach uses images of human facial skin diseases obtained from an African hospital. The experimental results of the study demonstrate the proposed framework's efficacy, outperforming numerous prior works on skin disease problems. On the other hand, statistical experiments with Wilcoxon and analysis of variance (ANOVA) tests are conducted to evaluate the proposed approach in terms of effectiveness and stability. The recorded results confirm the superiority of the proposed method when compared with other optimization algorithms and machine learning models.

Based on the suggested BERSFS classification model, excellent prediction accuracy of monkeypox disease is achieved. The tested dataset is assessed using a classifier built with the specified BERSFS algorithm. The BERSFS algorithm is tested and compared to PSO [33], Grey Wolf Optimizer (GWO) [34], the Whale Optimization Algorithm (WOA) [35], BER [30], and SFS [32]. Comparisons are made between models, including the decision tree (DT) regressor [36], K-nearest neighbors (K-NN) [37], support vector machine (SVM) [38], and the "BERSFS" algorithm. Additionally, several CNN models including AlexNet [39], VGG19Net [40], GoogLeNet [41], and ResNet-50 [42] are tested.

The key contributions of this work can be summed up as follows:

- Offer machine learning techniques for predicting monkeypox disease;
- A new Al-Biruni Earth radius (BER) optimization-based stochastic fractal search (BERSFS) algorithm is suggested;
- To raise the tested dataset prediction accuracy, a BERSFS-based classifier is created.
- A comparison of the results of different algorithms to determine which is the most accurate is performed;
- The Wilcoxon rank-sum and ANOVA tests are used to determine the statistical significance of the BERSFS algorithm;
- It is possible to generalize and test the BERSFS-based classification algorithm for different kinds of datasets.

Here is how this paper is organized. Section 2 contains a literature review. Section 3 describes the BERSFS algorithm. Section 4 shows the results and analysis. In Section 5, the conclusions and future directions are given.

2. Literature Review

Computer-aided methodologies for automatically identifying diseases from images reduce human error and increase detection speed, which helps physicians and radiologists diagnose conditions more easily, thus protecting the patient against medical threats in the early stages. Those methodologies can be categorized into two categories. The first, traditional methodologies, involves methods that apply only image processing techniques. In contrast, the other evolutionary methodologies are methods that apply AI and DL techniques. In the traditional methodologies [43–46], the most important features used to detect and identify skin diseases are their color and texture information. In addition, the nature of the disease and the severity of the condition play a role in determining the size and shape of each individual lesion. The process of selecting features to use in traditional approaches is time-consuming. Furthermore, identifying the features that are relevant to the problem at hand is essential because it has an impact on the accuracy of the classification algorithm. On the other hand, evolutionary techniques are capable of learning traits in an automatic and effective manner. Utilizing one of the many different image segmentation algorithms available, it divides the image into regions based either on the pixels, the edges, or the regions themselves. In addition, in contrast to conventional approaches, feature extraction makes intelligent decisions regarding filter selection [47].

This section discusses the previously proposed evolutionary methodologies used in classifying skin lesions from a medical image. Arifin et al. [48] proposed a two-stage system to detect skin lesions using color images, starting by applying k-means clustering with color gradient techniques to identify infected skin. Then, they used an ANN to classify diseases. The accuracy of the first stage was 95.99%, while the second stage was 94.016% accurate. A prepossessing phase is added before feature extraction, and those features are used to predict the type of disease [49]. This technique's accuracy varies depending on the number of extracted features. These features are input to a feedforward ANN for training and testing. The researchers applied this method to nine types of skin diseases with an accuracy of up to 90%. Soliman and AlEnezi [50] proposed resizing the image to extract features using a pretrained CNN. Then, the features were classified using a multi-class SVM. The system detected three types of skin diseases with an accuracy rate of 100% in a very limited dataset.

Esteva et al. [20] described how an image classifier based on a CNN can reach performance that is comparable to that of 21 dermatologists who have received board certification.

The three steps of disease diagnosis were devised to determine if a particular skin lesion should be classified as malignant, benign, or non-neoplastic. In addition, a nine-way illness partition was carried out in order to place each lesion that was examined into one of the nine categories that were previously described. In their article [51], Sandeep et al. examined the use of DL-based approaches for the detection of various skin lesions. They came up with a CNN to separate skin lesions into the eight different illness categories. When compared with VGGNet, the classification provided by the suggested approach was completed four times more quickly. In comparison, the proposed system would provide an accuracy of up to 78%, whereas VGGNet only achieved a 71% success rate. It was proposed in [52] that a CNN could perform low-cost image analysis for identification of the herpes zoster virus (HZV). When performed on a sample size of 1000 photos, the early detection of the HZV had an accuracy of 89.6 percent. In addition, a transfer learning strategy was utilized for the diagnosis of measles disease, which yielded a detection rate of 81.7%, a specificity rate of 97.1%, and an accuracy rate of 95.2% when utilizing the ResNet-50 model [53].

The majority of the studies that are conducted on skin diseases focus on melanoma, chickenpox, psoriasis, and measles, and very little research takes monkeypox into consideration. Data augmentation was utilized by Ali et al. [54] in order to increase the overall size of the dataset. For the purpose of categorizing monkeypox and other diseases, a threefold cross-validation experiment was prepared, and many pretrained deep learning models were put to use. Additionally, an ensemble model consisting of all three models was generated. The ensemble system achieved an accuracy of 79.26% (±1.05%), whereas VGG16 earned an accuracy of 81.48% (±6.87%) and ResNet50 achieved an accuracy of 82.96% (±4.57%). The best overall accuracy was attained by ResNet50, which was 82.96% (±4.57%). Deep artificial intelligence models offer considerable potential in diagnosing monkeypox from digital skin photographs, according to feasibility research that used state-of-the-art AI deep models to analyze skin images for the purpose of monkeypox detection. The study had precision of 85%. To train deep models, however, one needs training samples that are significantly bigger if one wants to achieve more reliable detection power [55].

Sitaula and Shahi [56] analyzed and contrasted 13 distinct pretrained DL models for the identification of the monkeypox virus. After the models were fine-tuned by the addition of universal custom layers to each of them, the results were compared utilizing four different performance criteria. In order to achieve a higher level of overall performance, the best-performing DL models were selected, and then those models were "ensembled" by using majority voting on the probabilistic outputs that were acquired from those models. In their study [57], Akin et al. compared the performance of several different image classification networks. They found that the MobileNet V2 model achieved the best performance overall, with an accuracy score of 98.25%, a sensitivity score of 96.55%, a specificity score of 100.00%, and an F1 score of 98.25%. Explainable artificial intelligence was created with the help of this model. As a result, the classification model's judgment regarding classification was expounded upon.

Although the models for diagnosing monkeypox have been hopeful, they have increased in difficulty, making classical approaches so ineffective that they cannot solve certain NP-hard problems. Thus, we sought solutions to classic optimization methods' issues. Metaheuristics are widely used in continuous optimization due to their robust and reliable performance, global search, lack of prerequisites for a continuously differentiable or differentiable objective function, simple implementation, implicit parallelism, and lack of specific information about the problem to be solved. This work offers machine learning techniques for effectively predicting monkeypox diseases compared with the classical approaches. A new Al-Biruni Earth radius (BER) optimization-based stochastic fractal search (BERSFS) algorithm is presented to raise the tested dataset's prediction accuracy.

3. Materials and Methods

3.1. Convolutional Neural Network (CNN)

Convolutional neural networks are effective deep learning approaches. Although they can be used for text, signals, and other continuous data types, image inputs are where they shine the brightest when it comes to their applicability. Instead of being connected to one another, as neurons in traditional neural networks are, neurons in a convolutional layer are connected to the sub-regions of the layer that came before it. Neurons in a CNN create outputs that are spatially correlated because the sub-regions they represent may overlap. In contrast, neurons in fully connected neural networks provide outputs that are independent of one another. In addition, the number of parameters in a typical neural network can rapidly increase in response to an increase in the amount of inputs. A CNN, on the other hand, has a lower total number of network parameters since it has fewer connections, shared weights, and downsampling [29].

A CNN comprises several layers: the convolutional, max-pooling, and fully connected layers. The CNN is made up of two phases: a phase for the extraction of features followed by a phase for the classification of binary options. The proposed CNN features three different convolutional subsections. Every segment is comprised of multiple fundamental units. The fundamental unit is made up of three layers: the convolution layer, the ReLu activation layer, and the batch normalization layer. After each block of convolution, a pooling layer is formed. The sections that use convolution are used as feature extractors. The classification process is carried out by the three most recent layers: the average pooling layer, the fully connected layer, and the softmax layer. The number of basic blocks is a tunable hyperparameter that, when set, influences the architecture of the CNN [58]. The hyperparameters of the CNN are optimized with the help of Al-Biruni Earth radius (BER) optimization-based stochastic fractal search (BERSFS) algorithm, which is used to locate the most effective design.

3.2. Al-Biruni Earth Radius (BER) Algorithm

The Al-Biruni Earth radius (BER) optimizes the situation by segmenting the population into groups that are focused on exploration and exploitation. A dynamic process of switching up the composition of agent subgroups is used to strike a healthy balance between exploitative and exploratory activities. The population is split between exploration (which makes up 70%) and exploitation (30%). In order to raise the overall global average fitness level of both the exploration and exploitation groups, the number of agents in each of those groups has been increased. The exploration team uses mathematics to search for potentially fruitful areas in the immediate vicinity of their current location. This is accomplished by searching through the available possibilities over and over again for one that has a better degree of fitness [30].

The goal of optimization algorithms is to determine the best solution given a set of constraints. When using the BER, each member of the population is represented by a vector $\mathbf{S}$. This vector takes the form $\mathbf{S} = S_1, S_2, \ldots, S_d \in R$, where S_d is the size of the search space and d is the optimization parameter or feature. It has been recommended that the fitness function F be used in order to evaluate the performance of an individual up to a specific point. During these phases of optimization, populations are searched in search of a fitness-optimal vector $\mathbf{S}^*$. The process begins with selecting a sample of the population at random (solutions). In order for the BER to optimize, it first requires the fitness function, as well as the lower and upper limits for each solution, the dimension, and the size of the population. Algorithm 1 provides a visual representation of the BER optimization algorithm.

Algorithm 1 AL-Biruni Earth radius (BER) algorithm

```
**Initialize** BER population $\mathbf{S}_i (i = 1, 2, \ldots, d)$ with size $d$, iterations $T_{max}$, fitness function $F_n$, $t = 1$, BER parameters
**Calculate** fitness function $F_n$ for each $\mathbf{S}_i$
**Find** best solution as $\mathbf{S}^*$
**while** $t \leq T_{max}$ **do**
    **for** $(i = 1 : i < n_1 + 1)$ **do**
        **Update** $\mathbf{r}_1 = h_1 \frac{cos(x)}{1-cos(x)}$, $\mathbf{r}_2 = h_2 \frac{cos(x)}{1-cos(x)}$
        **Update** positions to move toward best solution as in Equation (1)
    **end for**
    **for** $(i = 1 : i < n_2 + 1)$ **do**
        **Update** $\mathbf{r} = h \frac{cos(x)}{1-cos(x)}$, $\mathbf{r}_3 = h_3 \frac{cos(x)}{1-cos(x)}$
        **Update** positions Elitism of best solution as in Equation (2)
        **Update** positions Investigating area around best solution as in Equation (3)
        **Compare** $\mathbf{S}(t+1)$ and $\mathbf{S'}(t+1)$ to select best solution $\mathbf{S}^*$
        **if** The best fitness has remained the same over the past two editions. **then**
            **Mutate** solution as in Equation (4)
        **end if**
    **end for**
    **Update** the fitness function $F_n$ for each $\mathbf{S}_i$
    **Find** best solution as $\mathbf{S}^*$
    **Update** BER parameters, $t = t + 1$
**end while**
**Return** $\mathbf{S}^*$
```

3.2.1. Exploration Operation

As will be explained in further detail below, exploration is the process that is responsible for both identifying interesting regions of the search space and avoiding local optimum stagnation through forward progress toward the optimal solution. Exploration is also responsible for determining whether or not a solution is optimal.

In order to get closer to the optimal solution, the lone explorer in the group will use this method to search for promising new regions to investigate in the near area surrounding where they are now located. To accomplish this, one must search through the numerous available options in the surrounding area in an attempt to choose one that is superior (in terms of physical health) to the others. In order to accomplish this, the research conducted by the BER makes use of the following equations:

$$\mathbf{S}(t+1) = \mathbf{S}(t) + \mathbf{D}(2\mathbf{r}_2 - 1), \mathbf{D} = \mathbf{r}_1(\mathbf{S}(t) - 1) \tag{1}$$

where $\mathbf{S}(t)$ is the solution vector at iteration t and $\mathbf{D}$ is the diameter of the circle in which the search agent will explore for promising areas, while $0 < x \leq 180$ and h is a number that is arbitrarily chosen from the range $[0, 2]$. Both $\mathbf{r}_1$ and $\mathbf{r}_2$ are examples of coefficient vectors, and their values may be calculated using the formula: $\mathbf{r} = h \frac{cos(x)}{1-cos(x)}$.

3.2.2. Exploitation Operation

The group in charge of exploiting opportunities has the duty to improve upon the solutions that are currently in place. At the conclusion of each iteration, the BER identifies the participants who have achieved the highest levels of physical fitness and awards them accordingly. The purpose of exploitation is accomplished by the BER through the employment of two unique approaches, both of which are explained here.

Moving in the direction of the optimal solution using the following equation, we can make progress in the direction of the optimal solution:

$$\mathbf{S}(t+1) = \mathbf{r}^2(\mathbf{S}(t) + \mathbf{D}), \mathbf{D} = \mathbf{r}_3(\mathbf{L}(t) - \mathbf{S}(t)) \tag{2}$$

where $\mathbf{r}_3$ is a random vector that is created using the formula $\mathbf{r} = h\frac{cos(x)}{1-cos(x)}$, which regulates the movement steps toward the best solution, $\mathbf{S}(t)$ is the solution vector at iteration t, $\mathbf{L}(t)$ is the best solution vector, and $\mathbf{D}$ refers to the distance vector.

For investigating the region surrounding the optimal solution, the territory surrounding the correct response is the most interesting potential option (leader). Because of this, some people try to find ways to enhance things by investigating possibilities that are somewhat near to the best option. In order to carry out the process described above, the BER makes use of the equation shown below:

$$\mathbf{S}'(t+1) = \mathbf{r}(\mathbf{S}^*(t) + \mathbf{k}), \mathbf{k} = 1 + \frac{2 \times t^2}{Max_{iter}^2} \tag{3}$$

where $\mathbf{S}^*(t)$ is the best solution. The best solution for $\mathbf{S}^*$ can be selected by comparing $\mathbf{S}(t+1)$ and $\mathbf{S}'(t+1)$. The solution will be modified according to the following equation if the best fitness has not changed throughout the course of the previous two iterations:

$$\mathbf{S}(t+1) = \mathbf{k} * z^2 - h\frac{cos(x)}{1-cos(x)} \tag{4}$$

where z is a random number in the range $[0,1]$.

3.2.3. Selection of the Best Solution

In order to ensure that the solutions have a high level of quality, the BER chooses the one that is going to be the most effective for usage in the subsequent cycle. However, due to the enhanced efficiency of the elitism technique, multi-modal functions might converge too soon. By taking a mutational approach and examining all around the members of the exploration group, the BER is able to deliver exceptional capabilities for the exploration process. The BER is able to delay convergence thanks to its powerful exploring capabilities. The BER pseudo-code can be viewed by referring to Algorithm 1 and looking at the section titles. To begin, we will provide the BER with basic information, such as the population size, mutation rate, and number of iterations. After that, the BER splits the participants up into two groups: those who undertake work that is exploratory and those who perform work that is exploitative. The BER method will automatically make adjustments to the size of each group as it works its way through the iterative process of finding the best answer. Each team completes its tasks using two different approaches. In order to ensure a diverse and in-depth inquiry, the BER will rearrange the order of the responses whenever it performs an iteration. For instance, a solution that was part of the exploration group in the previous iteration may migrate to the exploitation group in the iteration after that. Because of the exclusive nature of the BER's selection procedure, the leader will not be changed while the process is ongoing.

3.3. Stochastic Fractal Search (SFS) Algorithm

It is possible for random fractals to motivate a metaheuristic algorithm by making use of the time and accuracy of the conventional fractal technique. It is possible for a particle to have electrical potential energy, where each particle has the ability to diffuse. Other random particles can be created where the energy of the original particle is distributed among the new particles, and in each generation, the best particles are retained while the remaining particles are discarded. These three guidelines are the foundation of the fundamental fractal search (FS) procedure, which is utilized to locate a remedy for a specific issue that has been presented. The stochastic fractal search, also known as SFS, is a type of algorithm that was designed with the fractal paradigm as its foundation [59,60]. By utilizing three different update mechanisms—one for diffusion, one for the first update, and one for the second update—SFS is able to go around the limitations of FS [61,62]. The diffusion phase of the SFS algorithm consists of a series of Gaussian walks that are carried out around the best possible solution (best particle).

In the DLA growth process, the Gaussian distribution approach is used for random walk. This is so that new particles can be generated based on the diffusion mechanism of the SFS. A list of walks that were formed throughout the process of diffusion in accordance with the optimal solution $\mathbf{S}^*(t)$ is created. The expression can be calculated as follows:

$$\mathbf{S}'^*(t+1) = Gaussian(\mu_{\mathbf{S}^*}, \sigma) + (\eta \times \mathbf{S}^*(t) - \eta' \times \mathbf{P}_i) \tag{5}$$

where the updated best solution is represented by the symbol $\mathbf{S}'^*(t+1)$. The parameters of η and η' are random numbers $\in [0,1]$. The value represented by $\mathbf{P}_i$ is the location of the *i*th point in the group that surrounds the point. Since there are fewer generations around the optimal solution, the value of $\mu_{\mathbf{S}^*}$ is equivalent to $|\mathbf{S}^*|$, and the value of σ is equivalent to $|\mathbf{P}_i - \mathbf{S}^*|$. This boosts the potential of exploration in the proposed BER, which is based on the diffusion process of the SFS algorithm, so that it can find the optimal solution more quickly. Algorithm 2 shows the SFS algorithm step by step.

Algorithm 2 Stochastic fractal search (SFS) algorithm

```
Initialize SFS population $\mathbf{S}_i (i = 1, 2, \ldots, d)$ with size $d$, iterations $T_{max}$, fitness function $F_n$, $t = 1$, SFS parameters
Calculate fitness function $F_n$ for each $\mathbf{S}_i$
Find best solution as $\mathbf{S}^*$
while $t \leq T_{max}$ do
    for $(i = 1 : i < n + 1)$ do
        Calculate updated best solution as in Equation (5)
    end for
    Update the fitness function $F_n$ for each $\mathbf{S}_i$
    Find best solution as $\mathbf{S}^*$
    Update SFS parameters, $t = t + 1$
end while
Return $\mathbf{S}^*$
```

3.4. Proposed BERSFS Algorithm

A step-by-step explanation of the proposed AL-Biruni Earth radius stochastic fractal search (BERSFS) algorithm can be found in Algorithm 3. The BERSFS algorithm mitigates the drawbacks of the BER and SFS algorithms while capitalizing on the benefits of both in order to produce the optimal answer for the whole world. The first step in the method is to find the starting positions of d preset agents using the notation $\mathbf{S}_i (i = 1, 2, \ldots, d)$. This also defines the parameters for the BER algorithm and the SFS algorithm, as well as the maximum number of allowable iterations for the execution process, which is denoted by T_{max}. A value that is completely unpredictable and falls anywhere between 0 and 1 is referred to as $rand_{BERSFS}$. The BER equations will be used by the BERSFS algorithm to determine how the positions of the agents should be updated if $rand_{BERSFS} > 0.5$. If $rand_{BERSFS} \leq 0.5$, then the SFS equations will be used by the BERSFS algorithm to guide the process of updating the positions of the agents.

The following is an expression of the computational complexity that the BERSFS algorithm presents in this paper. For iterations T_{max} and n number of agents, one definition of the complexity is as follows:

- Initialize the parameters of the BERSFS algorithm: $O(1)$;
- Calculate F_n for each agent $\mathbf{S}_i$: $O(n)$;
- Obtain the best agent $\mathbf{S}^*$: $O\ (n)$;
- Update positions to head toward the best solution: $O(T_{max} \times n)$;
- Update positions for the elitism of the best solution: $O(T_{max} \times n)$;
- Update the positions for investigating the area around the best solution: $O(T_{max} \times n)$;
- Mutate the solution: $O(T_{max})$;
- Calculate the updated best solution: $O(T_{max} \times n)$;
- Calculate F_n for each agent $\mathbf{S}_i$: $O(T_{max})$;

- Update the BERSFS parameters: $O(T_{max})$;
- Obtain the best agent $\mathbf{S}^*$: $O(T_{max})$;
- Obtain the best agent $\mathbf{S}^*$: $O(1)$.

Algorithm 3 Proposed BERSFS algorithm

```
Initialize BERSFS population $\mathbf{S}_i (i = 1, 2, \ldots, d)$ with size $d$, iterations $T_{max}$, fitness function $F_n$, $t = 1$, BERSFS parameters
Calculate fitness function $F_n$ for each $\mathbf{S}_i$
Find best solution as $\mathbf{S}^*$
while $t \leq T_{max}$ do
    if ($rand_{BERSFS} > 0.5$) then
        for ($i = 1 : i < n_1 + 1$) do
            Update $\mathbf{r}_1 = h_1 \frac{cos(x)}{1-cos(x)}$, $\mathbf{r}_2 = h_2 \frac{cos(x)}{1-cos(x)}$
            Calculate $\mathbf{D} = \mathbf{r}_1(\mathbf{S}(t) - 1)$
            Update positions to head toward best solution as
            $\mathbf{S}(t+1) = \mathbf{S}(t) + \mathbf{D}(2\mathbf{r}_2 - 1)$
        end for
        for ($i = 1 : i < n_2 + 1$) do
            Update $\mathbf{r} = h \frac{cos(x)}{1-cos(x)}$, $\mathbf{r}_3 = h_3 \frac{cos(x)}{1-cos(x)}$
            Calculate $\mathbf{D} = \mathbf{r}_3(\mathbf{L}(t) - \mathbf{S}(t))$
            Update positions Elitism of best solution as
            $\mathbf{S}(t+1) = \mathbf{r}^2(\mathbf{S}(t) + \mathbf{D})$
            Calculate $\mathbf{k} = 1 + \frac{2 \times t^2}{Max_{iter}^2}$
            Update positions Investigating area around best solution as
            $\mathbf{S'}(t+1) = \mathbf{r}(\mathbf{S}^*(t) + \mathbf{k})$
            Compare $\mathbf{S}(t+1)$ and $\mathbf{S'}(t+1)$ to select best solution $\mathbf{S}^*$
            if best fitness is not changed for last two iterations then
                Mutate solution as $\mathbf{S}(t+1) = \mathbf{k} * z^2 - h \frac{cos(x)}{1-cos(x)}$
            end if
        end for
    else
        for ($i = 1 : i < n + 1$) do
            Calculate updated best solution as
            $\mathbf{S'}^*(t+1) = Gaussian(\mu_{\mathbf{S}^*}, \sigma) + (\eta \times \mathbf{S}^*(t) - \eta' \times \mathbf{P}_i)$
        end for
    end if
    Update the fitness function $F_n$ for each $\mathbf{S}_i$
    Find best solution as $\mathbf{S}^*$
    Update BERSFS parameters, $t = t + 1$
end while
Return $\mathbf{S}^*$
```

According to the preceding investigation of the BERSFS algorithm, the level of computational complexity was determined to be $O(T_{max} \times n)$.

4. Experimental Results

This section devotes its entirety to providing an in-depth analysis of the investigation's results. There were three distinct environments in which the studies were conducted. In the first scenario, the BERSFS algorithm's capabilities for the dataset under testing were covered in comparison with those of basic models. In the second situation, however, the algorithm's classification capabilities were demonstrated in comparison with those of deep learning models. In the final example, the classification skills of the BERSFS algorithm were contrasted with those of state-of-the-art optimization techniques that are based on CNNs. Table 1 shows the parameters for the BERSFS algorithm's configuration. The parameters of the experimental CNN configuration are presented in Table 2. These are the values that will be used in this experiment.

Table 1. Parameters for BERSFS algorithm configuration.

Parameter	Value
Number of Agents	10
Number of Iterations	80
Number of Repetitions	20
η	$\in[0, 1]$
η'	$\in[0, 1]$
Mutation probability	0.5
Exploration percentage	70
K (decreases from 2 to 0)	1

Table 2. Parameters of the experimental CNN configuration.

Parameter	Value
CNN training options (Default)	
Momentum Learn	0.9000
RateDropFactor	0.1000
L2Regularization	1.0000×10^{-4}
LearnRateDropPeriod	10
GradientThreshold	Inf
GradientThresholdMethod	l2norm
ValidationData	imds
VerboseFrequency	50
ValidationPatience	Inf
ValidationFrequency	50
ResetInputNormalization	1
CNN training options (Custom)	
InitialLearnRate	0.001
ExecutionEnvironment	gpu
BatchSize	32
MaxEpochs	40
Verbose	0
Shuffle	every-epoch
LearnRateSchedule	piecwise
Optimizer	BERSFS

4.1. Dataset Description

An openly accessible dataset on Kaggle [63] served as the foundation for the experiments that were carried out as part of this body of work. Concern has been raised among medical professionals all over the world in response to the most recent monkeypox outbreak. In order to accomplish this goal, a novel skin image-based dataset dedicated to the diagnosis of monkeypox disease was developed. There were a total of 293 photographs of normal cases, 279 images of monkeypox cases, 107 images of chickenpox patients, and 91 images of measles cases included in this collection. Sources based on the internet were used to compile all of the image classes. The Department of Computer Science and Engineering at Islamic University, which is located in Kushtia-7003, Bangladesh, was responsible for the development of the full dataset. Figure 1 displays several example images taken from the dataset. During the preprocessing phase, each of the images that were included in the dataset for each of the four separate types—monkeypox, chickenpox, measles, and normal—were scaled and normalized to have dimensions of 227 × 227. This step took place before any of the other processing steps. This was performed before the images were incorporated into the model being proposed.

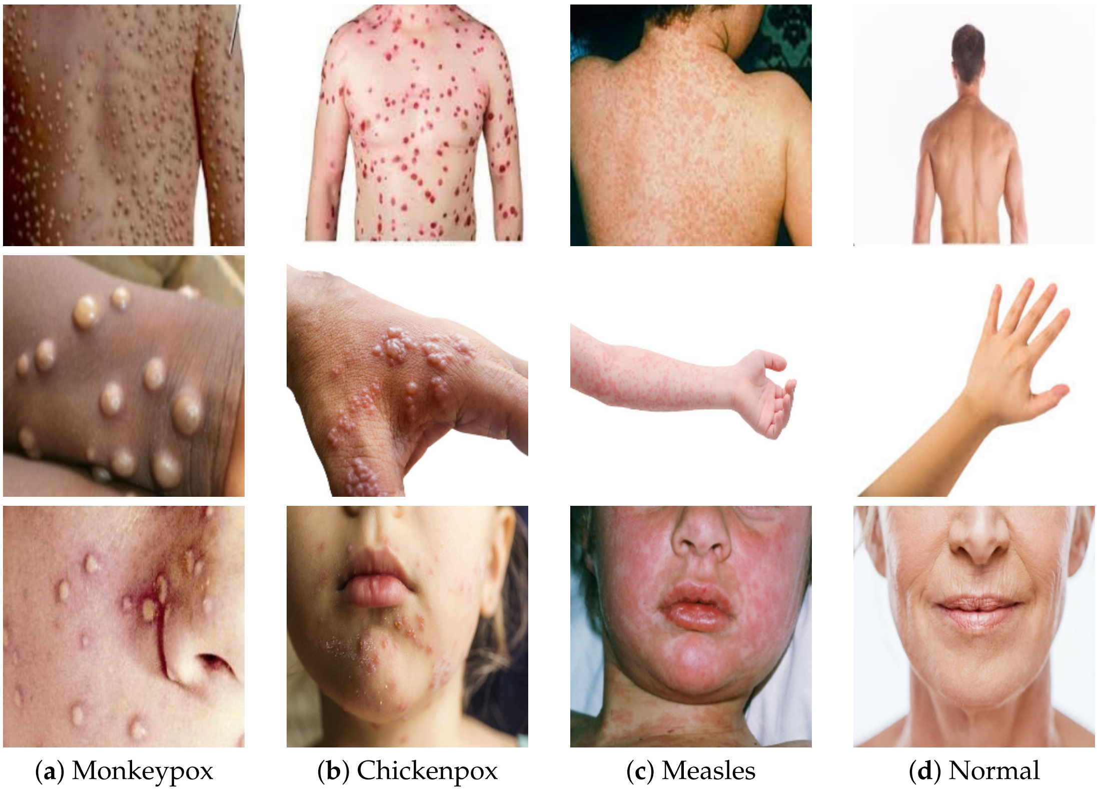

Figure 1. Samples from the Monkeypox Skin Images Dataset (MSID) [63]: (**a**) monkeypox cases, (**b**) chickenpox cases, (**c**) measles cases, and (**d**) normal cases.

4.2. Performance Metrics

The models can be evaluated with a number of different performance indicators, such as accuracy, precision, F1 score, recall, specificity, and sensitivity. The number of cases that were correctly diagnosed as being defective is represented by the letter "TP", which stands for "true positive". The number of cases that were wrongly recognized as defective is referred to as the "false positive", and its abbreviation is "FP". In addition, "false negative" (abbreviated as "FN") refers to the number of instances that were wrongly categorized as non-defective and shows the total number of these occurrences. The abbreviation "TN" stands for "true negative", which refers to the number of instances that were accurately identified as being free of defects. The definitions of the measures can be found in Table 3 .

Table 3. Performance metrics used in CNN.

No. Calculation	Metrics
Accuracy	$\frac{TP+TN}{TP+TN+FP+FN}$
Sensitivity	$\frac{TP}{TP+FN}$
Specificity	$\frac{TN}{TN+FP}$
Precision (PPV)	$\frac{TP}{TP+FP}$
Negative Predictive Value (NPV)	$\frac{TN}{TN+FN}$
F1 Score	$2 \times \frac{PPV \times TPR}{PPV+TPR}$

4.3. Comparison with Basic Models

In the trials, the classification results of the presented BERSFS algorithm, which is based on the CNN classifier, are discussed. There were 20 runs, and 80 iterations with

10 agents were used, as shown in Table 1. The fundamental classifiers of the decision tree (DT) Regressor [36], K-nearest neighbors (K-NN) [37], and support vector machine (SVM) [38] models were initially evaluated and contrasted with the suggested BERSFS algorithm-based CNN technique before being applied to the dataset that would be examined.

The effectiveness of the proposed method, which is referred to as BERSFS-CNN, was evaluated in comparison to the widely utilized classifiers CNN, SVM-Linear, k-NN, and DT. The results of this comparison are presented in Table 4. As a result of the BERSFS approach improving the performance of the CNN after tweaking its hyperparameters, the BERSFS-CNN model provided the best results when employed as a classifier, with an accuracy of 0.9883. The simple CNN model achieved the second-best accuracy with a score of 0.9337, followed by the SVM-Linear model with a score of 0.9213 and then the K-NN model with a score of 0.8777. The DT model achieved the lowest accuracy at 0.8510. This was as a result of the fact that the performance of the CNN was enhanced by the algorithm that was proposed.

Table 4. Results of comparisons between proposed BERSFS-CNN and basic models.

	Accuracy	**Sensitivity (TRP)**	**Specificity (TNP)**	***p* Value (PPV)**	**N Value (NPV)**	**F1 Score**
BERSFS-CNN	0.9883	0.8571	0.9921	0.7595	0.9959	0.8054
CNN	0.9337	0.7500	0.9693	0.8257	0.9524	0.7860
SVM-Linear	0.9213	0.8571	0.9231	0.2308	0.9959	0.3636
K-NN	0.8777	0.8000	0.9132	0.8081	0.9091	0.8040
DT	0.8510	0.7273	0.9132	0.8081	0.8696	0.7656

The results of the Wilcoxon signed-rank test and the analysis of variance (based on 10 runs and 80 iterations for 10 agents) results are included in Tables 5 and 6, respectively. This statistical testing, which had a p value that is less than 0.05, revealed that there was a significant difference between the outcomes of using the suggested BERSFS-CNN classifier and the outcomes of using alternative methods. Figure 2 shows the homoscedasticity graphs, heat maps, residual plots, and QQ plots of the BERSFS-CNN and other basic models. The figure confirms the performance of the proposed model.

Table 5. Results of comparisons between proposed classifiers and those obtained from ANOVA.

	SS	**DF**	**MS**	**F (DFn, DFd)**	***p* Value**
Treatment (between columns)	0.1129	4	0.0282	F (4, 45) = 1258	$p < 0.0001$
Residual (within columns)	0.0010	45	2.24×10^{-5}	-	-
Total	0.1139	49	-	-	-

Table 6. Test results for the Wilcoxon signed-rank test of the BERSFS-CNN model and basic classifiers.

	BERSFS-CNN	**CNN**	**SVM-Linear**	**K-NN**	**DT**
Theoretical median	0	0	0	0	0
Actual median	0.9883	0.9337	0.9213	0.8777	0.8511
Number of values	10	10	10	10	10
Wilcoxon signed-rank test					
Sum of signed ranks (W)	55	55	55	55	55
Sum of positive ranks	55	55	55	55	55
Sum of negative ranks	0	0	0	0	0
p value (two-tailed)	0.002	0.002	0.002	0.002	0.002
Exact or estimate?	Exact	Exact	Exact	Exact	Exact
Significant (alpha = 0.05)?	Yes	Yes	Yes	Yes	Yes
How big is the discrepancy?					
Discrepancy	0.9883	0.9337	0.9213	0.8777	0.8511

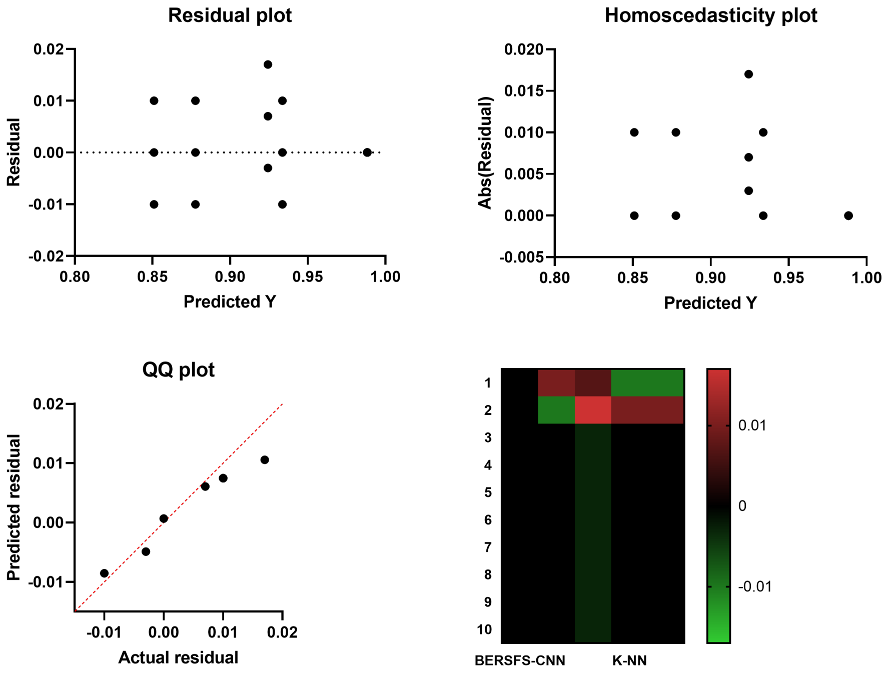

Figure 2. The homoscedasticity graphs, heat maps, residual plots, and QQ plots of the BERSFS and other basic models.

4.4. Comparison with Deep Learning Models

The deep learning models of AlexNet [39], VGG19Net [40], GoogLeNet [41], and ResNet-50 [42] were used as benchmarks for the BERSFS-CNN algorithm that was developed. The performance metrics used in this part are listed in Table 3. The effectiveness of the proposed method, which is referred to as BERSFS-CNN, was evaluated in comparison to the widely utilized deep learning models. The results of this comparison are presented in Table 7. When put to use as a classifier, the BERSFS-CNN model produced the highest-quality results, achieving an accuracy of 0.9883. This was due to the fact that the BERSFS technique resulted in an improvement in the CNN's performance after the hyperparameters were tuned. With a score of 0.9459, the AlexNet model achieved the second-best accuracy, followed by the GoogLeNet model with a score of 0.9351 and the VGG19Net model with a score of 0.9280. The ResNet-50 model achieved the lowest level of accuracy possible at 0.9208. This was due to the improved performance of the CNN, which was the driving force for the decision to make the shift.

Table 7. Results of comparisons between proposed BERSFS-CNN and deep learning models.

	Accuracy	Sensitivity (TRP)	Specificity (TNP)	*p* Value (PPV)	N Value (NPV)	F1 Score
BERSFS-CNN	0.9883	0.8571	0.9921	0.7595	0.9959	0.8054
AlexNet	0.9459	0.7143	0.9524	0.2941	0.9917	0.4167
GoogLeNet	0.9351	0.7143	0.9412	0.2500	0.9917	0.3704
VGG19Net	0.9280	0.7143	0.9339	0.2273	0.9917	0.3448
ResNet-50	0.9208	0.6667	0.9266	0.1739	0.9917	0.2759

Tables 8 and 9 present the findings of the Wilcoxon signed-rank test and the analysis of variance, respectively. These findings were based on 10 runs of the simulation and 80 iterations for 10 different agents. This statistical test, which had a *p* value that is less than 0.05, demonstrated that there was a significant difference between the results of using the suggested BERSFS-CNN classifier and the results of applying alternative approaches. The *p* value for this test was less than 0.05. Figure 3 depicts the homoscedasticity graphs, heat maps, residual plots, and QQ plots of the BERSFS-CNN, in addition to those of the other deep learning models. Heat maps provide a visual representation of the temperature of an area. The effectiveness of the model that was suggested can be shown to be proven by looking at the figure.

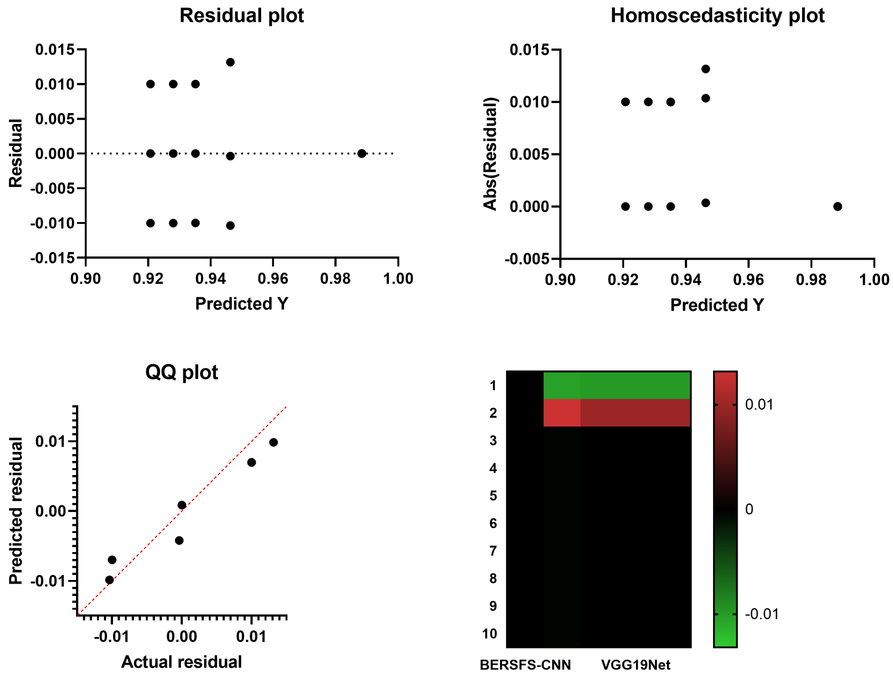

Figure 3. The homoscedasticity graphs, heat maps, residual plots, and QQ plots of the BERSFS-CNN and other deep learning models.

Table 8. Results of comparisons between proposed classifiers and those obtained from ANOVA.

	SS	DF	MS	F (DFn, DFd)	p Value
Treatment (between columns)	0.0284	4	0.0071	F (4, 45) = 363.3	$p < 0.0001$
Residual (within columns)	0.0009	45	0.1959×10^{-4}	-	-
Total	0.0293	49	-	-	-

Table 9. Results for the Wilcoxon signed-rank test of the BERSFS-CNN model and deep learning models.

	BERSFS-CNN	AlexNet	GoogLeNet	VGG19Net	ResNet-50
Theoretical median	0	0	0	0	0
Actual median	0.9883	0.9459	0.9351	0.928	0.9208
Number of values	10	10	10	10	10
Wilcoxon signed-rank test					
Sum of signed ranks (W)	55	55	55	55	55
Sum of positive ranks	55	55	55	55	55
Sum of negative ranks	0	0	0	0	0
p value (two-tailed)	0.002	0.002	0.002	0.002	0.002
Exact or estimate?	Exact	Exact	Exact	Exact	Exact
Significant (alpha = 0.05)?	Yes	Yes	Yes	Yes	Yes
How big is the discrepancy?					
Discrepancy	0.9883	0.9459	0.9351	0.928	0.9208

4.5. Comparison with Optimization-Based Models

The classification results of the suggested algorithm are compared with the results of other CNN-based models, including the BER-CNN [30], SFS-CNN [32], WOA-CNN [35], GWO-CNN [34], and PSO-CNN [33], in order to demonstrate the effectiveness of the provided method. Table 10 shows the parameter settings for the methods being compared. The performance metrics used in this part are listed in Table 3.

Table 10. Parameter settings for the methods being compared.

Algorithm	Parameter (s)	Value (s)
BER	Mutation probability	0.5
	Exploration percentage	70
	K (decreases from 2 to 0)	1
SFS	η	$\in$[0, 1]
	η'	$\in$[0, 1]
PSO	Acceleration constants	[2, 2]
	Inertia W_{max}, W_{min}	[0.6, 0.9]
	Particles	10
	Iterations	80
GWO	a	2 to 0
	Iterations	80
	Wolves	10
WOA	r	[0, 1]
	Iterations	80
	Whales	10
	a	2 to 0

Table 11 displays the classification outcomes of the proposed and contrasted techniques that are based on optimizing the CNN model's parameters. The BERSFS-CNN model generated results with an accuracy value that could reach a maximum of 0.9883. The results reveal that the model was superior to cutting-edge classifier models that were constructed using the CNN technique. The BER-CNN-based model produced the second-best classification results with an accuracy value of 0.9759, followed by the SFS-CNN-based model with a score of 0.972, the PSO-CNN-based model with a score of (0.968), the GWO-CNN-based model with a score of 0.9636, and the WOA-CNN-based model with a score of 0.9598, which produced the results that were the least accurate.

Table 11. Classification outcomes of the proposed and contrasted techniques that were based on optimizing the CNN model's parameters.

	Accuracy	Sensitivity (TRP)	Specificity (TNP)	p Value (PPV)	N Value (NPV)	F1 Score
BERSFS-CNN	0.9883	0.8571	0.9921	0.7595	0.9959	0.8054
BER-CNN	0.9759	0.7500	0.9796	0.3750	0.9959	0.5000
SFS-CNN	0.9720	0.6000	0.9796	0.3750	0.9917	0.4615
PSO-CNN	0.9680	0.4000	0.9796	0.2857	0.9877	0.3333
GWO-CNN	0.9636	0.4000	0.9767	0.2857	0.9859	0.3333
WOA-CNN	0.9598	0.7674	0.9655	0.3976	0.9929	0.5238

Tables 12–14 contain statistical descriptions of the provided BERSFS-CNN classifier and the comparative classifiers, respectively, as well as the results of the ANOVA and Wilcoxon signed-rank tests based on 10 runs and 80 iterations for 10 agents. This statistical testing indicates the significant difference between the suggested BERSFS-CNN classifier's outcomes and those of other algorithms, with a p value of less than 0.05. In addition to the homoscedasticity graphs, heat maps, residual plots, and QQ plots of the other fundamental models that are displayed in Figure 4, the BERSFS-CNN also contains its very own versions of each of these sorts of plots. When one looks at the graphic Figure 4, it is very clear that the usefulness of the model that was provided is proven.

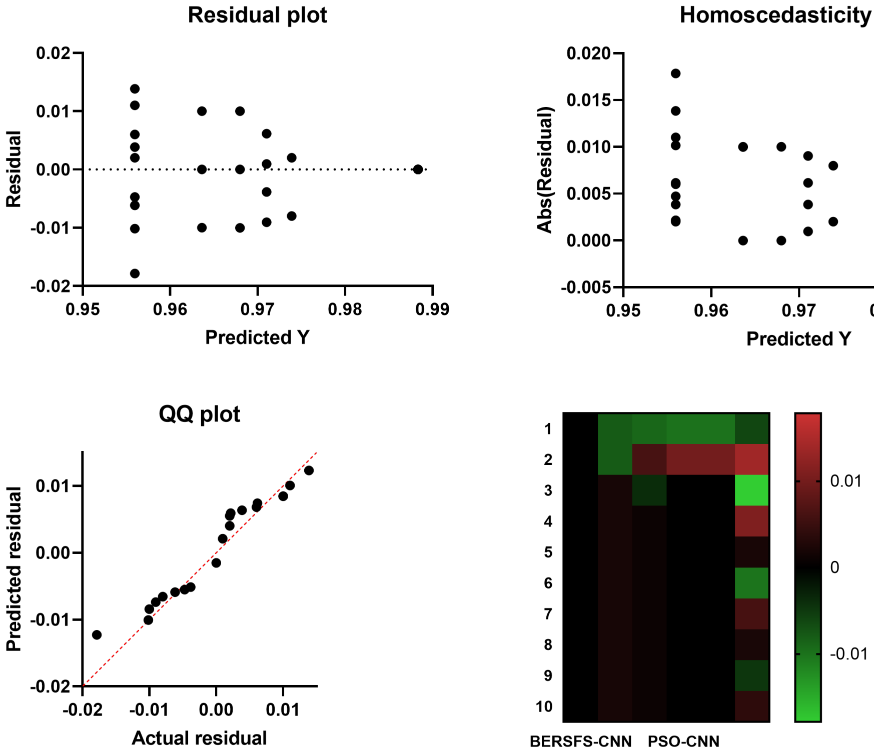

Figure 4. The homoscedasticity graphs, heat maps, residual plots, and QQ plots of the BERSFS-CNN and other optimization-based models.

Table 12. Results of comparisons between proposed classifiers and those obtained from ANOVA.

	SS	DF	MS	F (DFn, DFd)	*p* Value
Treatment (between columns)	0.0059	5	0.0012	F (5, 54) = 41.27	$p < 0.0001$
Residual (within columns)	0.0016	54	2.88×10^{-5}	-	-
Total	0.0075	59	-	-	-

Table 13. Test results for the Wilcoxon signed-rank test of the BERSFS-CNN model and optimization-based models.

	BERSFS-CNN	BER-CNN	SFS-CNN	PSO-CNN	GWO-CNN	WOA-CNN
Theoretical median	0	0	0	0	0	0
Actual median	0.9883	0.9759	0.972	0.968	0.9636	0.9581
Number of values	10	10	10	10	10	10
Wilcoxon signed-rank test						
Sum of signed ranks (W)	55	55	55	55	55	55
Sum of positive ranks	55	55	55	55	55	55
Sum of negative ranks	0	0	0	0	0	0
p value (two-tailed)	0.002	0.002	0.002	0.002	0.002	0.002
Exact or estimate?	Exact	Exact	Exact	Exact	Exact	Exact
Significant (alpha = 0.05)?	Yes	Yes	Yes	Yes	Yes	Yes
How big is the discrepancy?						
Discrepancy	0.9883	0.9759	0.972	0.968	0.9636	0.9581

Table 14. Statistical descriptions of the provided BERSFS-CNN classifier and the comparative classifiers.

	BERSFS-CNN	BER-CNN	SFS-CNN	PSO-CNN	GWO-CNN	WOA-CNN
Number of values	10	10	10	10	10	10
Minimum	0.9883	0.9659	0.962	0.9580	0.9536	0.9381
25th percentile	0.9883	0.9734	0.9708	0.9680	0.9636	0.9488
Median	0.9883	0.9759	0.972	0.9680	0.9636	0.9581
75th percentile	0.9883	0.9759	0.972	0.9680	0.9636	0.9632
Maximum	0.9883	0.9759	0.9772	0.9780	0.9736	0.9698
Range	0	0.0100	0.0152	0.0200	0.0200	0.0317
Mean	0.9883	0.9739	0.9710	0.9680	0.9636	0.9560
Std. deviation	0	0.0042	0.0036	0.0047	0.0047	0.0097
Std. error of mean	0	0.0013	0.0013	0.0015	0.0015	0.0031
Geometric mean	0.9883	0.9739	0.9710	0.9680	0.9636	0.9559
Geometric SD factor	1	1.0040	1.0040	1.0050	1.0050	1.0100
Sum	9.8830	9.7390	9.7100	9.6800	9.6360	9.5600

The box plot in Figure 5 illustrates the accuracy of the proposed BERSFS-CNN and compared approaches. The BERSFS-CNN-based optimization algorithm's highest accuracy results are shown in the graph below. The accuracy histogram for the algorithms presented and compared is displayed in Figure 6, and this is based on the number of values with the bin center range 0.938–0.968, which attests to the stability of the suggested approach. Figures 7 and 8 show the ROC curves of the proposed BERSFS-CNN algorithm versus the BER-CNN and WOA-CNN algorithms, respectively. The ROC curves confirm the stability of the proposed BERSFS-CNN algorithm.

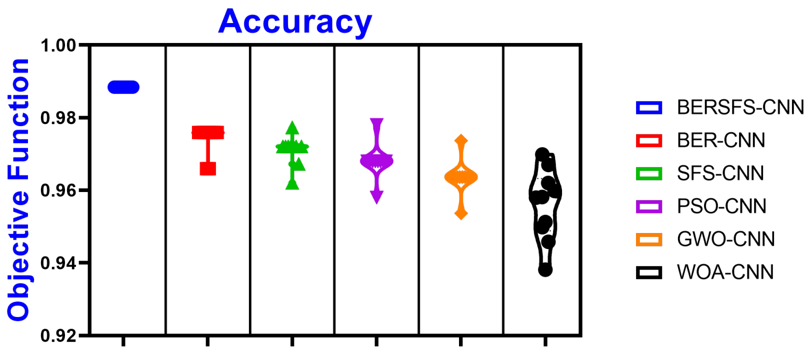

Figure 5. The box plot of the accuracy of the proposed BERSFS-CNN and comparison approaches.

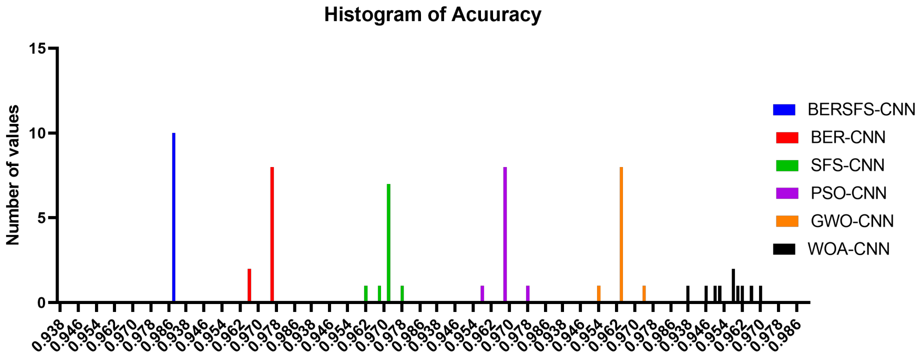

Figure 6. The accuracy histogram for the algorithms presented and compared with number of values in the bin center range 0.938–0.968.

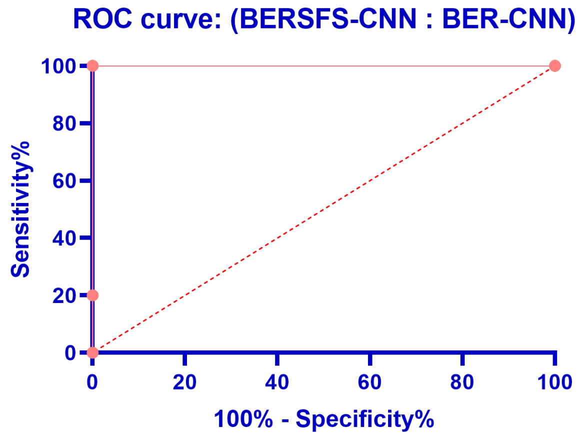

Figure 7. ROC curve of the proposed BERSFS-CNN algorithm versus the BER-CNN algorithm.

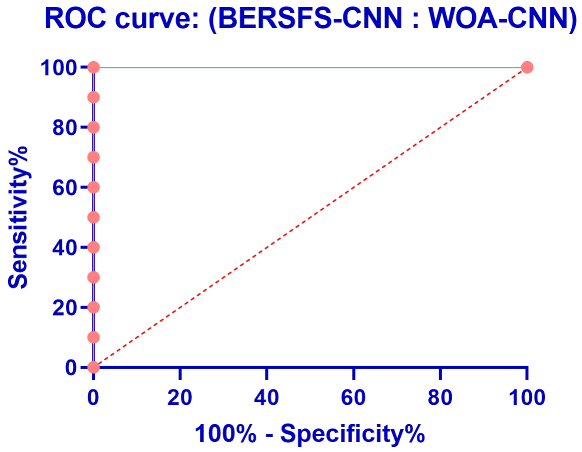

Figure 8. ROC curve of the proposed BERSFS-CNN algorithm versus the WOA-CNN algorithm.

5. Conclusions

To fine-tune the deep convolutional neural network (DCNN) layers, we offer a new framework that uses Al-Biruni Earth radius (BER) optimization-based stochastic fractal search (BERSFS). In the first stage of the proposed method, deep CNN-based models are utilized in order to acquire knowledge regarding the embedding of input images into Euclidean space. In the second stage, we calculate the distance between pairs of images in Euclidean space using an improved classification model that is based on the triplet loss function. This allows us to learn features that may be used to differentiate between different cases, including monkeypox cases. The method that has been proposed makes use of photographs of human facial skin illnesses that were collected from a hospital in Africa. The experimental findings of the study provide evidence that the proposed framework is effective, outperforming a large number of other studies that were conducted previously on skin disease issues. On the other hand, statistical experiments using the Wilcoxon and analysis of variance (ANOVA) tests were carried out in order to assess the proposed method with regard to its efficiency and consistency. When compared with other optimization algorithms and machine learning models, the findings that were recorded provided conclusive evidence that the method that was provided was superior.

Author Contributions: Conceptualization, M.E.G.; methodology, A.I. and A.A.A.; software, E.-S.M.E.-K. and M.M.E.; validation, S.M. and W.H.L.; formal analysis, A.A.A.; investigation, E.-S.M.E.-K., N.K. and M.M.E.; resources, N.K., W.H.L. and M.E.G.; data curation, E.-S.M.E.-K. and M.M.E.; writing—original draft, A.I. and M.E.G.; writing—review & editing, A.I. and S.M.; visualization, A.A.A. and N.K.; supervision, F.K.K. and S.M.; project administration, D.S.K. and F.K.K.; funding acquisition, D.S.K. and F.K.K. All authors have read and agreed to the published version of the manuscript.

Funding: This work received funding from Princess Nourah bint Abdulrahman University Researchers Supporting Project number PNURSP2022R300 of Princess Nourah bint Abdulrahman University in Riyadh, Saudi Arabia.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The authors would like to thank the Deanship of Scientific Research at Shaqra University for supporting this work.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

1. Somogyi, Z. *The Application of Artificial Intelligence*; Springer International Publishing: Berlin/Heidelberg, Germany, 2021. [CrossRef]
2. Chakraborty, U. *Artificial Intelligence for All: Transforming Every Aspect of Our Life*; BPB Publications: Noida, India, 2020.
3. Zhou, X.; Wang, S.; Chen, H.; Hara, T.; Yokoyama, R.; Kanematsu, M.; Fujita, H. Automatic localization of solid organs on 3D CT images by a collaborative majority voting decision based on ensemble learning. *Comput. Med. Imaging Graph.* **2012**, *36*, 304–313. [CrossRef] [PubMed]
4. Hussain, M.A.; Hamarneh, G.; Garbi, R. Cascaded Regression Neural Nets for Kidney Localization and Segmentation-free Volume Estimation. *IEEE Trans. Med. Imaging* **2021**, *40*, 1555–1567. [CrossRef] [PubMed]
5. Chen, Z.; Li, X.; Yang, M.; Zhang, H.; Xu, X.S. Optimize Deep Learning Models for Prediction of Gene Mutations Using Unsupervised Clustering. *arXiv* **2022**, arXiv:2204.01593.
6. Habibi, K.; Tirdad, K.; Dela Cruz, A.; Wenger, K.; Mari, A.; Basheer, M.; Kuk, C.; van Rhijn, B.W.G.; Zlotta, A.R.; van der Kwast, T.H.; et al. ABC: Artificial Intelligence for Bladder Cancer grading system. *Mach. Learn. Appl.* **2022**, *9*, 100387. [CrossRef]
7. Hunter, B.; Hindocha, S.; Lee, R.W. The Role of Artificial Intelligence in Early Cancer Diagnosis. *Cancers* **2022**, *14*, 1524. [CrossRef]
8. El-Kenawy, E.S.M.; Mirjalili, S.; Ibrahim, A.; Alrahmawy, M.; El-Said, M.; Zaki, R.M.; Eid, M.M. Advanced Meta-Heuristics, Convolutional Neural Networks, and Feature Selectors for Efficient COVID-19 X-Ray Chest Image Classification. *IEEE Access* **2021**, *9*, 36019–36037. [CrossRef]
9. Basu, K.; Sinha, R.; Ong, A.; Basu, T. Artificial Intelligence: How is It Changing Medical Sciences and Its Future? *Indian J. Dermatol.* **2020**, *65*, 365–370. [CrossRef]
10. The Handbook of Brain Theory and Neural Networks. Available online: https://mitpress.mit.edu/9780262511025/the-handbook-of-brain-theory-and-neural-networks/ (accessed on 2 October 2022).
11. Fenner, F. Adventures with poxviruses of vertebrates. *Fems Microbiol. Rev.* **2000**, *24*, 123–133. [CrossRef]
12. Poxviruses—An Overview | ScienceDirect Topics. Available online: https://www.sciencedirect.com/topics/neuroscience/poxviruses (accessed on 2 October 2022).
13. Monkeypox Virus Infection in Humans across 16 Countries—April–June 2022 | NEJM. Available online: https://www.nejm.org/doi/full/10.1056/NEJMoa2207323 (accessed on 2 October 2022).
14. Memariani, M.; Memariani, H. Multinational monkeypox outbreak: What do we know and what should we do? *Ir. J. Med. Sci.* **2022**, *2022*, 1–2. [CrossRef]
15. Multi-Country Monkeypox Outbreak in Non-Endemic Countries. Available online: https://www.who.int/emergencies/disease-outbreak-news/item/2022-DON385 (accessed on 2 October 2022).
16. Gong, Q.; Wang, C.; Chuai, X.; Chiu, S. Monkeypox virus: A re-emergent threat to humans. *Virol. Sin.* **2022**, *37*, 477–482. [CrossRef]
17. Laboratory Testing for the Monkeypox Virus. Available online: https://apps.who.int/iris/bitstream/handle/10665/354488/WHO-MPX-Laboratory-2022.1-eng.pdf?sequence=1&isAllowed=y (accessed on 2 October 2022).
18. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet Classification with Deep Convolutional Neural Networks. In *Advances in Neural Information Processing Systems*; Curran Associates, Inc.: Lake Tahoe, NV, USA, 2012; Volume 25.
19. Ioffe, S.; Szegedy, C. Batch ormalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In Proceedings of the 32nd International Conference on Machine Learning, Vienna, Austria, 29–30 October 2015; pp. 448–456.
20. Esteva, A.; Kuprel, B.; Novoa, R.A.; Ko, J.; Swetter, S.M.; Blau, H.M.; Thrun, S. Dermatologist–level classification of skin cancer with deep neural networks. *Nature* **2017**, *542*, 115–118. [CrossRef]
21. Haenssle, H.A.; Fink, C.; Schneiderbauer, R.; Toberer, F.; Buhl, T.; Blum, A.; Kalloo, A.; Hassen, A.B.H.; Thomas, L.; Enk, A.; et al. Man against machine: Diagnostic performance of a deep learning convolutional neural network for dermoscopic melanoma recognition in comparison to 58 dermatologists. *Ann. Oncol.* **2018**, *29*, 1836–1842. [CrossRef]

22. Hekler, A.; Utikal, J.S.; Enk, A.H.; Hauschild, A.; Weichenthal, M.; Maron, R.C.; Berking, C.; Haferkamp, S.; Klode, J.; Schadendorf, D.; et al. Superior skin cancer classification by the combination of human and artificial intelligence. *Eur. J. Cancer* **2019**, *120*, 114–121. [CrossRef]
23. Akinola, O.O.; Ezugwu, A.E.; Agushaka, J.O.; Zitar, R.A.; Abualigah, L. Multiclass feature selection with metaheuristic optimization algorithms: A review. *Neural Comput. Appl.* **2022**, *34*, 19751–19790. [CrossRef]
24. Eid, M.M.; El-kenawy, E.S.M.; Ibrahim, A. A binary Sine Cosine-Modified Whale Optimization Algorithm for Feature Selection. In Proceedings of the 2021 National Computing Colleges Conference (NCCC), Taif, Saudi Arabia, 27–28 March 2021. [CrossRef]
25. Ibrahim, A.; Mirjalili, S.; El-Said, M.; Ghoneim, S.S.M.; Al-Harthi, M.M.; Ibrahim, T.F.; El-Kenawy, E.S.M. Wind Speed Ensemble Forecasting Based on Deep Learning Using Adaptive Dynamic Optimization Algorithm. *IEEE Access* **2021**, *9*, 125787–125804. [CrossRef]
26. Trojovská, E.; Dehghani, M. A new human-based metahurestic optimization method based on mimicking cooking training. *Sci. Rep.* **2022**, *12*, 14861. [CrossRef]
27. El-Kenawy, E.S.M.; Mirjalili, S.; Abdelhamid, A.A.; Ibrahim, A.; Khodadadi, N.; Eid, M.M. Meta-Heuristic Optimization and Keystroke Dynamics for Authentication of Smartphone Users. *Mathematics* **2022**, *10*, 2912. [CrossRef]
28. Alhussan, A.A.; Khafaga, D.S.; El-Kenawy, E.S.M.; Ibrahim, A.; Eid, M.M.; Abdelhamid, A.A. Pothole and Plain Road Classification Using Adaptive Mutation Dipper Throated Optimization and Transfer Learning for Self Driving Cars. *IEEE Access* **2022**, *10*, 84188–84211. [CrossRef]
29. Abdelhamid, A.A.; El-Kenawy, E.S.M.; Alotaibi, B.; Amer, G.M.; Abdelkader, M.Y.; Ibrahim, A.; Eid, M.M. Robust Speech Emotion Recognition Using CNN+LSTM Based on Stochastic Fractal Search Optimization Algorithm. *IEEE Access* **2022**, *10*, 49265–49284. [CrossRef]
30. El-kenawy, E.S.M.; Abdelhamid, A.A.; Ibrahim, A.; Mirjalili, S.; Khodadad, N.; Al Duailij, M.A.; Alhussan, A.A.; Khafaga, D.S. Al-Biruni Earth Radius (BER) Metaheuristic Search Optimization Algorithm. *Comput. Syst. Sci. Eng.* **2023**, *45*, 1917–1934. [CrossRef]
31. Abdelhamid, A.A.; El-Kenawy, E.S.M.; Khodadadi, N.; Mirjalili, S.; Khafaga, D.S.; Alharbi, A.H.; Ibrahim, A.; Eid, M.M.; Saber, M. Classification of Monkeypox Images Based on Transfer Learning and the Al-Biruni Earth Radius Optimization Algorithm. *Mathematics* **2022**, *10*, 3614. [CrossRef]
32. Salimi, H. Stochastic Fractal Search: A powerful metaheuristic algorithm. *Knowl.-Based Syst.* **2015**, *75*, 1–18. [CrossRef]
33. Bello, R.; Gomez, Y.; Nowe, A.; Garcia, M.M. Two-Step Particle Swarm Optimization to Solve the Feature Selection Problem. In Proceedings of the Seventh International Conference on Intelligent Systems Design and Applications (ISDA 2007), Rio de Janeiro, Brazil, 20–24 October 2007; pp. 691–696. [CrossRef]
34. Al-Tashi, Q.; Abdul Kadir, S.J.; Rais, H.M.; Mirjalili, S.; Alhussian, H. Binary Optimization Using Hybrid Grey Wolf Optimization for Feature Selection. *IEEE Access* **2019**, *7*, 39496–39508. [CrossRef]
35. Mirjalili, S.; Lewis, A. The Whale Optimization Algorithm. *Adv. Eng. Softw.* **2016**, *95*, 51–67. [CrossRef]
36. Breiman, L. Random Forests. *Mach. Learn.* **2001**, *45*, 5–32. [CrossRef]
37. Jang, S.; Jang, Y.E.; Kim, Y.J.; Yu, H. Input initialization for inversion of neural networks using k-nearest neighbor approach. *Inf. Sci.* **2020**, *519*, 229–242. [CrossRef]
38. Tharwat, A. Parameter investigation of support vector machine classifier with kernel functions. *Knowl. Inf. Syst.* **2019**, *61*, 1269–1302. [CrossRef]
39. Han, J.; Zhang, D.; Cheng, G.; Liu, N.; Xu, D. Advanced Deep-Learning Techniques for Salient and Category-Specific Object Detection: A Survey. *IEEE Signal Process. Mag.* **2018**, *35*, 84–100. [CrossRef]
40. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv* **2014**, arXiv:1409.1556.
41. Al-Dhamari, A.; Sudirman, R.; Mahmood, N.H. Transfer Deep Learning Along with Binary Support Vector Machine for Abnormal Behavior Detection. *IEEE Access* **2020**, *8*, 61085–61095. [CrossRef]
42. Yu, S.; Xie, L.; Liu, L.; Xia, D. Learning Long-Term Temporal Features with Deep Neural Networks for Human Action Recognition. *IEEE Access* **2020**, *8*, 1840–1850. [CrossRef]
43. Amarathunga, A.a.L.C.; Ellawala, E.P.W.C.; Abeysekara, G.N.; Amalraj, C.R.J. Expert System For Diagnosis Of Skin Diseases. *Int. J. Sci. Technol. Res.* **2015**, *4*, 174–178.
44. Wei, L.s.; Gan, Q.; Ji, T. Skin Disease Recognition Method Based on Image Color and Texture Features. *Comput. Math. Methods Med.* **2018**, *2018*, 8145713. [CrossRef]
45. Chatterjee, S.; Dey, D.; Munshi, S.; Gorai, S. Extraction of features from cross correlation in space and frequency domains for classification of skin lesions. *Biomed. Signal Process. Control.* **2019**, *53*, 101581. [CrossRef]
46. *Diagnosis of Skin Lesions Based on Dermoscopic Images Using Image Processing Techniques*; IntechOpen: London, UK, 2022.
47. Narayanan, S.J.; Jaiswal, P.R.; Chowdhury, A.; Joseph, A.M.; Ambar, S. A Computational Intelligence Approach for Skin Disease Identification Using Machine/Deep Learning Algorithms. In *Computational Intelligence and Healthcare Informatics*; John Wiley & Sons, Ltd.: Hoboken, NJ, USA, 2021; Chapter 15, pp. 269–295. [CrossRef]
48. Shamsul Arifin, M.; Golam Kibria, M.; Firoze, A.; Ashraful Amini, M.; Yan, H. Dermatological disease diagnosis using color-skin images. In Proceedings of the 2012 International Conference on Machine Learning and Cybernetics, Xi'an, China, 15–17 July 2012; Volume 5, pp. 1675–1680. [CrossRef]

49. Yasir, R.; Rahman, M.A.; Ahmed, N. Dermatological disease detection using image processing and artificial neural network. In Proceedings of the 8th International Conference on Electrical and Computer Engineering, Dhaka, Bangladesh, 20–22 December 2014; pp. 687–690. [CrossRef]
50. ALKolifi ALEnezi, N.S. A Method Of Skin Disease Detection Using Image Processing and Machine Learning. *Procedia Comput. Sci.* **2019**, *163*, 85–92. [CrossRef]
51. Sandeep, R.; Vishal, K.P.; Shamanth, M.S.; Chethan, K. Diagnosis of Visible Diseases Using CNNs. In *Proceedings of the International Conference on Communication and Artificial Intelligence*; Goyal, V., Gupta, M., Mirjalili, S., Trivedi, A., Eds.; Lecture Notes in Networks and Systems; Springer Nature: Berlin/Heidelberg, Germany, 2022; pp. 459–468.
52. Mejía Lara, J.V.; Arias Velásquez, R.M. Low-cost image analysis with convolutional neural network for herpes zoster. *Biomed. Signal Process. Control* **2022**, *71*, 103250. [CrossRef]
53. Glock, K.; Napier, C.; Gary, T.; Gupta, V.; Gigante, J.; Schaffner, W.; Wang, Q. Measles Rash Identification Using Transfer Learning and Deep Convolutional Neural Networks. In Proceedings of the 2021 IEEE International Conference on Big Data (Big Data), Orlando, FL, USA, 15–18 December 2021; pp. 3905–3910. [CrossRef]
54. Ali, S.N.; Ahmed, M.T.; Paul, J.; Jahan, T.; Sani, S.M.S.; Noor, N.; Hasan, T. Monkeypox Skin Lesion Detection Using Deep Learning Models: A Feasibility Study. *arXiv* **2022**, arXiv:2207.03342.
55. Islam, T.; Hussain, M.A.; Chowdhury, F.U.H.; Islam, B.M.R. Can Artificial Intelligence Detect Monkeypox from Digital Skin Images? *bioRxiv* **2022**. [CrossRef]
56. Sitaula, C.; Shahi, T.B. Monkeypox virus detection using pre-trained deep learning-based approaches. *arXiv* **2022**, arXiv:2209.04444.
57. Akin, K.D.; Gurkan, C.; Budak, A.; Karataş, H. Classification of Monkeypox Skin Lesion using the Explainable Artificial Intelligence Assisted Convolutional Neural Networks. *Avrupa Bilim ve Teknoloji Dergisi* **2022**, *40*, 106–110. [CrossRef]
58. Bacanin, N.; Bezdan, T.; Tuba, E.; Strumberger, I.; Tuba, M. Optimizing Convolutional Neural Network Hyperparameters by Enhanced Swarm Intelligence Metaheuristics. *Algorithms* **2020**, *13*, 67. [CrossRef]
59. El-Kenawy, E.S.M.; Eid, M.M.; Saber, M.; Ibrahim, A. MbGWO-SFS: Modified Binary Grey Wolf Optimizer Based on Stochastic Fractal Search for Feature Selection. *IEEE Access* **2020**, *8*, 107635–107649. [CrossRef]
60. El-kenawy, E.S.M.; Albalawi, F.; Ward, S.A.; Ghoneim, S.S.M.; Eid, M.M.; Abdelhamid, A.A.; Bailek, N.; Ibrahim, A. Feature Selection and Classification of Transformer Faults Based on Novel Meta-Heuristic Algorithm. *Mathematics* **2022**, *10*, 3144. [CrossRef]
61. El-Kenawy, E.S.M.; Mirjalili, S.; Alassery, F.; Zhang, Y.D.; Eid, M.M.; El-Mashad, S.Y.; Aloyaydi, B.A.; Ibrahim, A.; Abdelhamid, A.A. Novel Meta-Heuristic Algorithm for Feature Selection, Unconstrained Functions and Engineering Problems. *IEEE Access* **2022**, *10*, 40536–40555. [CrossRef]
62. Khafaga, D.S.; Alhussan, A.A.; El-Kenawy, E.S.M.; Ibrahim, A.; Eid, M.M.; Abdelhamid, A.A. Solving Optimization Problems of Metamaterial and Double T-Shape Antennas Using Advanced Meta-Heuristics Algorithms. *IEEE Access* **2022**, *10*, 74449–74471. [CrossRef]
63. Monkeypox Skin Images Dataset (MSID). Available online: https://www.kaggle.com/datasets/dipuiucse/monkeypoxskinimagedataset (accessed on 2 October 2022).