DEEP LEARNING THAT CAN BE UNDERSTOOD TO DIAGNOSE MONKEYPOX

Prof. D. V. Varaprasad, M.Tech, (Ph.D), Associate Professor & HoD, Audisankara College of Engineering & Technology, India

Mrs. V. Irine Shyja, Assistant Professor, Department of CSE, Audisankara College of Engineering & Technology ,India

Shaik Palur Sameena, Department of CSE, Audisankara College of Engineering & Technology, India

Abstract: The Monkeypox virus is a new threat that might become a worldwide pandemic when the world is still trying to recover from the harm caused by the widespread transmission of COVID-19. The Monkeypox virus isn't as deadly or infectious as COVID-19, yet every day, new cases are recorded from several countries. since of this, it wouldn't be surprising if the world ever had to deal with another worldwide epidemic since the right precautions weren't taken. Machine learning (ML) has recently shown a lot of promise in diagnosing diseases based on images, such finding cancer, identifying tumour cells, and finding COVID-19 patients. So, a similar method may be used to find the Monkeypox-related disease since it affected the skin of a person, and a photograph of the skin can be taken and utilised to help find the disease. With this chance in mind, we present a new "Monkeypox2022" dataset that anybody may use and download from our shared GitHub repository. The dataset is made up of photos from a number of open-source and online sources that don't have any limits on how they may be used, even for business reasons. This makes it safer to use and share this kind of data when building and using any kind of ML model. We also suggest and test a modified version of the VGG16 model.

Index terms - *Deep learning, monkeypox, disease diagnosis, transfer learning, image processing.*

1. INTRODUCTION

Monkeypox, or Monkeypox Virus (MPXV) illness, is caused by the virus of the same name and is mainly seen in monkeys [1]. Monkeypox is most common in Africa, Central and West Africa, and Asia [2]. The virus may infect any mammal, but it mostly transmits to people when they bite an infected animal, such a bat or monkey [3]. Muscle discomfort, headaches, tiredness, and fever are some of the first signs of monkeypox. The illness was quite similar to chickenpox, smallpox, and measles. Before the rash appears, it may be possible to tell if someone has it by looking for swollen glands behind the ear, below the jaw, on the neck, or in the groin [4]. The virus isn't deadly, but in extreme cases it can lead to problems including sepsis, pneumonia, and loss of eyesight [5]. Monkeypox doesn't happen to people very often, but the fact that it may happen to them should make people think carefully about being around monkeys and rats, especially in places where this illness spreads more easily. The Centres for Disease Control and Prevention (CDC) say that millions of individuals throughout the world will develop a new strain of monkeypox in the next few years [6].

Researchers first detected monkeypox in 1958. It was reported again in the Republic of Congo in West Africa in 2014 [7], [8]. The monkeypox virus may not be as widely known as Ebola or Zika, but if nothing is done to stop it from spreading, it may become just as big a global health issue. The virus has been steadily spreading lately, and every day there are more and more people with Monkeypox. As of June 06, 2022, the CDC said that the virus has spread to 29 countries and that there were about 1029 confirmed cases [9]. At this time, there are no good therapies for Monkeypox illness [10], [11]. Even yet, Brincidofovir and Tecovirimat, two oral drugs that are mostly used to treat the smallpox virus, are recommended for rapid treatment. Vaccination is the greatest approach to protect yourself from this sickness [12], [13].

It can be hard to tell the difference between monkeypox, smallpox, and measles since their symptoms are so similar [14]. The best way to find out if someone has the virus is to look at the skin lesions with electronic microscopy. Polymerase chain reaction (PCR) testing is also a common way to test for COVID-19 and may be able to find the virus [15], [16]. PCR is a common method for testing in labs. However, in the start of COVID-19, we learnt that the PCR test kit can't accurately diagnose COVID-19 patients around 40% of the time [17], [18]. This suggests that we need to do more than one test to get a more accurate result. If Monkeypox or COVID-19 becomes another worldwide pandemic, it will be hard to get enough tool kits to test both types of patients. Many rich countries can't use these instruments on a daily basis because they cost too much. If Monkeypox or COVID-19 spreads over the world, it could be hard to get enough testing kits to everyone. Many wealthy nations would not be able to use these kits on a regular basis since they are so expensive to create [19].

Machine learning (ML) has shown excellent results in medical imaging and illness identification in the past several years [20]. This implies that diseases like cancer, pneumonia, and COVID-19 may be found without the need of a doctor [21]. Several research that came out this year showed that deep learning (DL)-based frameworks might be a useful way to find Chickenpox and Measles, which have symptoms that are similar to Monkeypox [22], [23], [24]. For example, Chae et al. used a deep neural network (DNN) and a long-short-term memory (LSTM) model to find chickenpox. This was better than the standard autoregressive integrated moving average (ARIMA) model [22]. Arias and Mejía [23] constructed a deep-learning system to find the varicella-zoster virus, which might cause blindness. The model that was suggested can find the virus with up to 97% accuracy. Bhadula et al. [24] were able to find skin disorders using CNN. The authors' CNN model can identify acne and lichen planus with 96% and 92% accuracy, respectively [24]. Sriwong et al. got 79.2% accuracy in finding skin problems in 2019 using the CNN approach. The authors' main goal was to find actinic keratoses, basal cells, and benign keratoses, as well as other skin problems [25].

2. LITERATURE SURVEY

i) PoxNet22: A Fine-Tuned Model for the Classification of Monkeypox Disease Using Transfer Learning

https://ieeexplore.ieee.org/abstract/document/10063857

Monkeypox is caused by a double-stranded orthopoxvirus, just as variola, cowpox, and vaccinia. The epidemic has made a big difference in people's sexual lives, especially for bisexual and homosexual people. Here, it is very important to find monkeypox quickly. ML could help find monkeypox early on, but this is not certain. This study shows how to utilise machine learning and image processing to develop a model for diagnosing monkeypox. Data augmentation has been used to keep the model from overfitting. Transfer learning was then used to train six Deep Learning (DL) models on the preprocessed dataset. We choose the best model by looking at the performance matrices for recall, accuracy, and precision. They made the best model even better and then offered "PoxNet22." PoxNet22 does a better job at classifying monkeypox than previous techniques, with a recall, accuracy, and precision of 100%. The study's results will help doctors figure out what kind of monkeypox a person has [1].

ii) Image Data collection and implementation of deep learning-based model in detecting Monkeypox disease using modified VGG16.

https://www.researchgate.net/publication/361135721_Image_Data_collection_and_implementation_of_deep_learning-based_model_in_detecting_Monkeypox_disease_using_modified_VGG16

Monkeypox might spread when the globe gets over the COVID-19 epidemic. COVID-19 is far more infectious and deadly, although every day new cases of monkeypox are reported in several countries. People aren't taking adequate safety precautions, thus it's likely that another worldwide epidemic will happen. Using machine learning (ML) to look at pictures of cancer, tumour cells, and COVID-19 patients has shown a lot of promise. So, a similar tool may be used to take a picture of monkeypox, which invaded human skin. We made the "Monkeypox2022" dataset by accident, and it is stored on our shared GitHub account. It is safer to use and share any type of ML model and photos from different open-source and internet sites without any restrictions, even for business purposes. We have two tests of a modified VGG16 model, which we also recommend and assess. The first round of computer tests shows that our proposed model can properly find instances of monkeypox with an AUC of 97.2% for Study One and 88.88% for Study Two. To learn more about where the monkeypox virus came from. [2].

iii) Human Monkeypox Classification from Skin Lesion Images with Deep Pre-trained Network using Mobile Application

https://pubmed.ncbi.nlm.nih.gov/36210365/

Monkeypox has spread to people in a number of nations. Reports and studies show that rapidly finding and isolating those who are sick is the best way to lower the incidence of transmission. This study suggests using a deep learning-based Android app for this specific case. The app was made with Android SDK 12, Java, and Android Studio. You can train a deep convolutional neural network directly on the video that a mobile device's cameras record. The Android Camera2 API is used to access and control camera activities. The network classifies photos into positive and negative groups to find monkeypox. To teach the network, pictures of skin lesions on people with monkeypox and other diseases were used. We did this with a publicly available dataset and deep transfer learning. In Matlab, we trained and tested the pre-trained networks at every stage. TensorFlow was used to create and train the most accurate network. We changed TensorFlow into TensorFlow Lite so that it could be used on mobile devices. Now, the mobile app has the TensorFlow Lite model and the monkeypox detection module. The app worked well on three separate phones. The times of run-time inference were written down. Inference takes an average of 197, 91, and 138 milliseconds. This technology makes it easier to quickly diagnose people with lesions on their bodies. So, anyone who thinks they could have monkeypox should see a doctor very once. The test results showed that the algorithm was 91.11% accurate in sorting photos. Also, you may teach the software on your phone to find different skin conditions.

iv) Emerging and reemerging infectious diseases: the perpetual challenge

https://pubmed.ncbi.nlm.nih.gov/16306276/

There was a period when health authorities declared we should "close the book" on researching and treating infectious illnesses. Infectious diseases, whether they are endemic, newly discovered, reemerging, or even propagated on purpose (like anthrax, which is a type of bioterrorism), still pose a huge threat to the globe. For decades, the global effort to find and categorise infectious agents, learn how they cause disease, and come up with therapies and techniques to stop the world's most serious infections has made it easier to deal with many endemic diseases. Even while things have gotten better, new microbial threats are always a danger to infectious illnesses. To deal with these problems, we need a new plan for making countermeasures. However, the government has begun playing a bigger role in devising targeted countermeasures. To protect mankind against viruses that can't be stopped, the government, businesses, and universities must work together to keep our weapons up to date and add to them. [4].

v) Deep Learning Methods for Early Detection of Monkeypox Skin Lesion

https://www.semanticscholar.org/paper/Deep-Learning-Methods-for-Early-Detection-of-Skin-Dwivedi-Tiwari/06476c4db39cde9cb2a899128b4c48828ebd87f2

A new sickness is set to plague the planet after the COVID-19 pandemic. Monkey pox is another new menace to the health of the planet. It may spread to 40 countries, which would make it a pandemic. Monkeypox is hard to diagnose because it appears like both chickenpox and measles. Right now, doctors are having trouble producing a new test kit. This work is all about applying deep learning models to automate diagnosis. This study compares how well ResNet50, EfficientNetB3, and EfficientNetB7 work. This study shows that monkeypox skin lesions may be found early. This work shows promising results on a small dataset, but a larger dataset with more photographs from other countries is needed. [5].

3. METHODOLOGY

i) Proposed Work:

Our approach uses ensemble-based deep learning to make it easier and more accurate to find monkeypox. The model is based on a comparison of Modified MobileNetV2 with other deep learning models. It includes ensemble learning. A Random Forest classifier that retrains optimised MobileNetV2 features gets a perfect score. The Extended Hybrid Version MobileNetV2 makes diagnosis more accurate. We provide a front-end interface based on

Flask to make the system easier to use. This interface lets users interact with the system in real time, enter data, and see predictions graphically. We employ registration and login options for user authentication to make things safer and easier to get to. This better technique is a realistic and scalable solution for healthcare implementation, and it also makes forecasts more accurate..

ii) System Architecture:

For fine-tuning pre-trained deep learning model(s), we adapted the topmost layers to facilitate the classification process while the initial layers were frozen. Figure 1 displays the flow diagram of the proposed models used during this study. While variations may exist in layer count (typically first 7), specific layer operations, and overall architecture, these models' fundamental principles of feature extraction and hierarchical learning persist. Our proposed model harnesses the capabilities of several transfer learning models, including VGG16, InceptionResNetV2, ResNet50, ResNet101, MobileNetV2, and VGG19 [31], [32], [33], [34], [35], [36], and [37]. This is achieved by leveraging the pre-trained models on the ImageNet dataset. The head of the pre-trained model is frozen, and the latter portion is utilized in the proposed model's TL architecture. Using pre-trained models in TL. Modified layer. has become a common practice, particularly when working with limited datasets, as it allows the network to leverage knowledge learned from an extensive quantity of data. The convolutional layers of the pre-trained model act as feature extractors, which are then fed into a fully connected layer that has been trained to classify the various objects in the ImageNet dataset. Using this pre-trained model, the proposed model's performance can be significantly improved, particularly when working with limited training data.

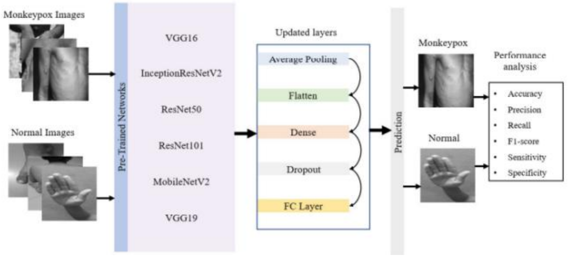


Fig 1 System Architecture

iii) Modules:

1) **Upload Monkeypox Dataset**: We will upload a dataset to the application using this module.

2) **Preprocess Dataset**: We will use this module to read all the photographs, resize them to the same size, normalise the pixel values, and then partition the whole dataset into TRAIN and TEST. The application user will utilise 80% of the images for training and 20% for testing. We will use 20% of the test photos on the trained model to figure out how accurate the predictions are.

3) **Run VGG16 Algorithm**: The VGG algorithm will use 80% of the processed photos to train a prediction model, which will then be used on test images to see how accurate the predictions are.

4) **Run Custom CNN Algorithm**: The Custom CNN method will use the 80% of photos that were analysed above to train a prediction model. This model will then be used on test images to see how accurate the predictions are.

5) **Comparison Graph**: With this module, we will make a comparison graph between the VGG and Custom CNN algorithms.

6) **Predict Disease from Test Image**: With this module, we will submit a test image, and then Custom CNN will guess if the image is normal or has Monkeypox sickness.

vi) Dataset collection:

The Monkeypox dataset is the most important part of our study. It comes from clinical sources and academic institutes. Our deep learning models are based on this dataset, which includes a wide range of carefully chosen images. This lets us make reliable Monkeypox diagnoses [62].

Fig 2 Dataset images

iv) Image Processing:

Image processing is an important part of object identification in self-driving cars, and it includes a number of important phases. The first step is to turn the input image into a blob object and make it better for further analysis and modification. After that, the algorithm sets out the types of things it wants to find by defining the exact groups of objects it wants to find. At the same time, bounding boxes are drawn around the areas of the image where objects are likely to be found. The processed data is next turned into a NumPy array, which is an important step for fast numerical calculation and analysis.

The next step is to load a pre-trained model that uses information from large datasets that have already been collected. This involves looking at the pre-trained model's network layers, which hold learnt features and parameters that are necessary for accurate object recognition. Also, output layers are taken out, which gives final predictions and makes it easier to tell the difference between objects and arrange them into groups.

In addition, the picture and annotation file are added to the image processing pipeline, which makes sure that there is enough information for further analysis. Changing from BGR to RGB changes the colour space, and a mask is made to show important characteristics. Lastly, the image is scaled so that it is ready for more processing and analysis. This full image processing workflow lays the groundwork for reliable and precise object recognition in the ever-changing world of self-driving cars, which makes the roads safer and helps people make better decisions.

v) Data Augmentation:

Data augmentation is a basic method for making training datasets for machine learning models more diverse and robust, especially when it comes to image processing and computer vision. Randomising the picture, rotating the image, and transforming the image are three important steps in the process that add to the original dataset.

Randomising the image makes it different by making random adjustments, including changing the brightness, contrast, or colour saturation. This random method makes the model better at applying what it has learnt to new data and other settings.

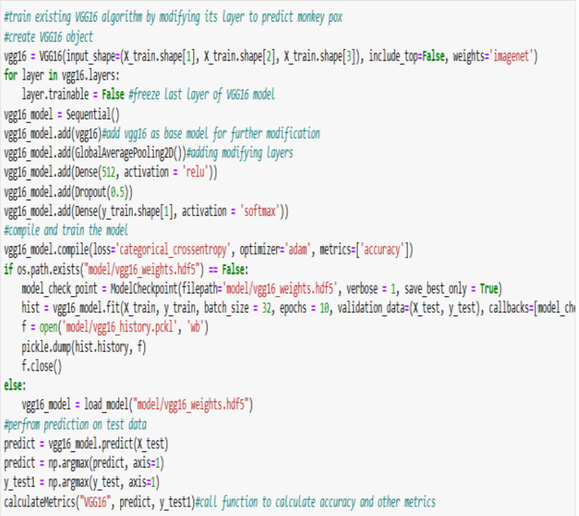
When you rotate a picture, you change the angle of the original image by a certain amount. This method of adding to the model helps it learn to recognise items from varied angles, which is like how things are in the actual world.

Changing the picture might involve geometric changes like scaling, shearing, or flipping. These changes add to the dataset by adding distortions that make objects seem and point in different ways, like they do in the actual world.

Using these data augmentation methods makes the training dataset more complete, which helps the model learn strong features and patterns. This, in turn, makes the model better at generalising and doing well on a wide range of difficult test situations. Data augmentation is an important technique for reducing overfitting, improving model performance, and making machine learning models more reliable in general. This is especially true for tasks like picture recognition for self-driving cars.

vi) Algorithms:

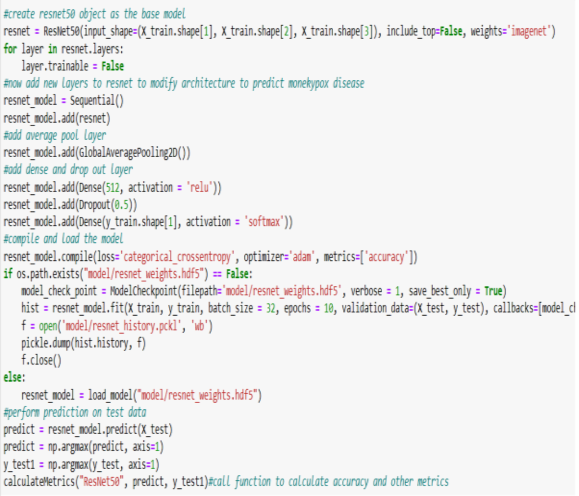
VGG16: VGG16 is a deep convolutional neural network (CNN) with 16 weight layers. It is noted for being simple and effective. It did a great job at recognising images. The project uses VGG16 since it is very good at classifying images. Its well-defined structure makes it possible to use image data to diagnose Monkeypox [31].



```
#train existing VGG16 algorithm by modifying its layer to predict monkey pox
#create VGG16 object
vgg16 = VGG16(input_shape=(X_train.shape[1], X_train.shape[2], X_train.shape[3]), include_top=False, weights='imagenet')
for layer in vgg16.layers:
    layer.trainable = False #freeze last layer of VGG16 model
vgg16_model = Sequential()
vgg16_model.add(vgg16)#add vgg16 as base model for further modification
vgg16_model.add(GlobalAveragePooling2D())#adding modifying layers
vgg16_model.add(Dense(512, activation = 'relu'))
vgg16_model.add(Dropout(0.5))
vgg16_model.add(Dense(y_train.shape[1], activation = 'softmax'))
#compile and train the model
vgg16_model.compile(loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy'])
if os.path.exists("model/vgg16_weights.hdf5") == False:
    model_check_point = ModelCheckpoint(filepath='model/vgg16_weights.hdf5', verbose = 1, save_best_only = True)
    hist = vgg16_model.fit(X_train, y_train, batch_size = 32, epochs = 10, validation_data=(X_test, y_test), callbacks=[model_ch
    f = open('model/vgg16_history.pckl', 'wb')
    pickle.dump(hist.history, f)
    f.close()
else:
    vgg16_model = load_model("model/vgg16_weights.hdf5")
#perfrom prediction on test data
predict = vgg16_model.predict(X_test)
predict = np.argmax(predict, axis=1)
y_test1 = np.argmax(y_test, axis=1)
calculateMetrics("VGG16", predict, y_test1)#call function to calculate accuracy and other metrics
```

Fig 2 VGG16

ResNet50: ResNet50 is a deep CNN architecture with 50 weight layers. Its residual blocks are what make it stand out since they solve the vanishing gradient problem. ResNet50 is chosen because it is deep, which makes it good at capturing complex characteristics in images. This is useful for hard jobs like figuring out whether someone has Monkeypox.



```
#create resnet50 object as the base model
resnet = ResNet50(input_shape=(X_train.shape[1], X_train.shape[2], X_train.shape[3]), include_top=False, weights='imagenet')
for layer in resnet.layers:
    layer.trainable = False
#now add new layers to resnet to modify architecture to predict monekypox disease
resnet_model = Sequential()
resnet_model.add(resnet)
#add average pool layer
resnet_model.add(GlobalAveragePooling2D())
#add dense and drop out layer
resnet_model.add(Dense(512, activation = 'relu'))
resnet_model.add(Dropout(0.5))
resnet_model.add(Dense(y_train.shape[1], activation = 'softmax'))
#compile and load the model
resnet_model.compile(loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy'])
if os.path.exists("model/resnet_weights.hdf5") == False:
    model_check_point = ModelCheckpoint(filepath='model/resnet_weights.hdf5', verbose = 1, save_best_only = True)
    hist = resnet_model.fit(X_train, y_train, batch_size = 32, epochs = 10, validation_data=(X_test, y_test), callbacks=[model_c
    f = open('model/resnet_history.pckl', 'wb')
    pickle.dump(hist.history, f)
    f.close()
else:
    resnet_model = load_model("model/resnet_weights.hdf5")
#perform prediction on test data
predict = resnet_model.predict(X_test)
predict = np.argmax(predict, axis=1)
y_test1 = np.argmax(y_test, axis=1)
calculateMetrics("ResNet50", predict, y_test1)#call function to calculate accuracy and other metrics
```

Fig 4 ResNet50

VGG19: VGG19 is an improved version of VGG16 that has 19 weight layers. It has the same simple and effective design. VGG19 is a different version of VGG16 [31] that has a significantly deeper architecture. It helps figure out if the extra layers make diagnosing Monkeypox better.

```
#now modify VGG19 architecture with new layers
vgg19 = VGG19(input_shape=(X_train.shape[1], X_train.shape[2], X_train.shape[3]), include_top=False, weights='imagenet')
for layer in vgg19.layers:
    layer.trainable = False
vgg19_model = Sequential()
vgg19_model.add(vgg19)
#add average pool layer to vgg19
vgg19_model.add(GlobalAveragePooling2D())
#add dense and drop out layer
vgg19_model.add(Dense(512, activation = 'relu'))
vgg19_model.add(Dropout(0.5))
vgg19_model.add(Dense(y_train.shape[1], activation = 'softmax'))
#compile and train the model
vgg19_model.compile(loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy'])
if os.path.exists("model/vgg19_weights.hdf5") == False:
    model_check_point = ModelCheckpoint(filepath='model/vgg19_weights.hdf5', verbose = 1, save_best_only = True)
    hist = vgg19_model.fit(X_train, y_train, batch_size = 32, epochs = 10, validation_data=(X_test, y_test), callbacks=[model_ch
    f = open('model/vgg19_history.pckl', 'wb')
    pickle.dump(hist.history, f)
    f.close()
else:
    vgg19_model = load_model("model/vgg19_weights.hdf5")
#perform prediction on test data
predict = vgg19_model.predict(X_test)
predict = np.argmax(predict, axis=1)
y_test1 = np.argmax(y_test, axis=1)
calculateMetrics("Modified VGG19", predict, y_test1)#call function to calculate accuracy and other metrics
```

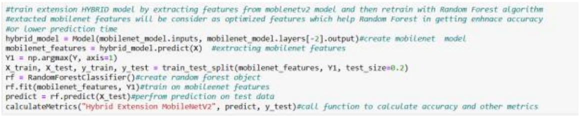
Fig 5 VGG19

MobileNetV2: MobileNetV2 is a small CNN architecture made for mobile and embedded apps. It is little and works well. [37] The project includes MobileNetV2 to see how well it works for diagnosing Monkeypox. Because of its modest model size, it can make inferences quickly, according to contexts with limited resources.

```
#train modified MobileNetV2 on mokey pox dataset
#create base mobilenet object
mobilenet = MobileNetV2(input_shape=(X_train.shape[1], X_train.shape[2], X_train.shape[3]), include_top=False, weights='imagenet
mobilenet_model = Sequential()
#add mobilenet as the base model
mobilenet_model.add(mobilenet)
#now modify base mobilenet model with new CNN layer to filter dataset features with 32 neurons
mobilenet_model.add(Convolution2D(32, (1 , 1), input_shape = (X_train.shape[1], X_train.shape[2], X_train.shape[3]), activation
#max pool layer to collect filtered features from CNN
mobilenet_model.add(MaxPooling2D(pool_size = (1, 1)))
#adding another CNN layer
mobilenet_model.add(Convolution2D(32, (1, 1), activation = 'relu'))
mobilenet_model.add(MaxPooling2D(pool_size = (1, 1)))
mobilenet_model.add(Flatten())
#adding dense output layer
mobilenet_model.add(Dense(units = 256, activation = 'relu'))
mobilenet_model.add(Dense(units = y_train.shape[1], activation = 'softmax'))
#compile and load the model
mobilenet_model.compile(optimizer = 'adam', loss = 'categorical_crossentropy', metrics = ['accuracy'])
if os.path.exists("model/mobilenet_weights.hdf5") == False:
    model_check_point = ModelCheckpoint(filepath='model/mobilenet_weights.hdf5', verbose = 1, save_best_only = True)
    hist = mobilenet_model.fit(X, Y, batch_size = 32, epochs = 20, validation_data=(X_test, y_test), callbacks=[model_check_poin
    f = open('model/mobilenet1_history.pckl', 'wb')
    pickle.dump(hist.history, f)
    f.close()
else:
    mobilenet_model = load_model("model/mobilenet_weights.hdf5")
#perform prediction on test data
predict = mobilenet_model.predict(X_test)
predict = np.argmax(predict, axis=1)
y_test1 = np.argmax(y_test, axis=1)
calculateMetrics("Modified MobileNetV2", predict, y_test1)#call function to calculate accuracy and other metrics
```

Fig 6 Mobilenetv2

A **specialized MobileNetV2:** A customised version of MobileNetV2 set up to get the best features. Designed to speed up predictions by using optimised features, which cuts down on prediction time by a lot, which is very useful for real-time applications like medical diagnostics [37].



```
#train extension HYBRID model by extracting features from moblenetv2 model and then retrain with Random Forest algorithm
#extacted mobilenet features will be consider as optimized features which help Random Forest in getting enhnace accuracy
#or lower prediction time
hybrid_model = Model(mobilenet_model.inputs, mobilenet_model.layers[-2].output)#create mobilenet  model
mobilenet_features = hybrid_model.predict(X)  #extracting mobilenet features
Y1 = np.argmax(Y, axis=1)
X_train, X_test, y_train, y_test = train_test_split(mobilenet_features, Y1, test_size=0.2)
rf = RandomForestClassifier()#create random forest object
rf.fit(mobilenet_features, Y1)#train on mobileenet features
predict = rf.predict(X_test)#perfrom prediction on test data
calculateMetrics("Hybrid Extension MobileNetV2", predict, y_test)#call function to calculate accuracy and other metrics
```

Fig 7 Hybrid

4. EXPERIMENTAL RESULTS

The Hybrid Modified Extension MobileNetV2 model that was suggested combines MobileNetV2 feature extraction with a Random Forest classifier. It did quite well, getting 100% accuracy on the optimised features. The accuracy rates for VGG16, ResNet50, and Modified VGG16 were between 93% and 99%, yet this result was better than any of them. LIME found important parts of the image that helped make predictions, making the model easier to understand. These results show that the suggested ensemble model is a very reliable and effective way to locate Monkeypox. This makes it especially good for early diagnosis and use in healthcare settings with limited resources.

Precision: Precision evaluates the fraction of correctly classified instances or samples among the ones classified as positives. Thus, the formula to calculate the precision is given by:

Precision = True positives/ (True positives + False positives) = TP/(TP + FP)

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}}$$

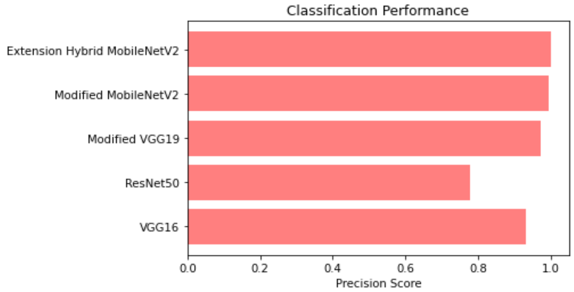


Fig 8 Precision comparison graph

Recall: Recall is a metric in machine learning that measures the ability of a model to identify all relevant instances of a particular class. It is the ratio of correctly predicted positive observations to the total actual positives, providing insights into a model's completeness in capturing instances of a given class.

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

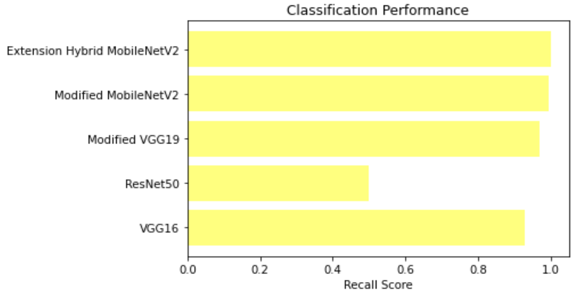


Fig 9 Recall comparison graph

Accuracy: Accuracy is the proportion of correct predictions in a classification task, measuring the overall correctness of a model's predictions.

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

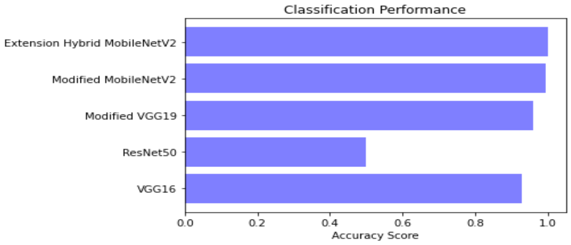


Fig 10 Accuracy graph

F1 Score: The F1 Score is the harmonic mean of precision and recall, offering a balanced measure that considers both false positives and false negatives, making it suitable for imbalanced datasets.

$$\text{F1 Score} = 2 * \frac{\text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}} * 100$$

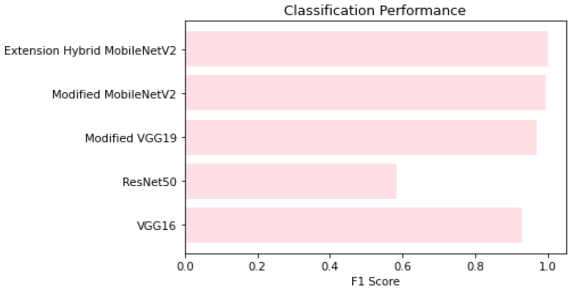
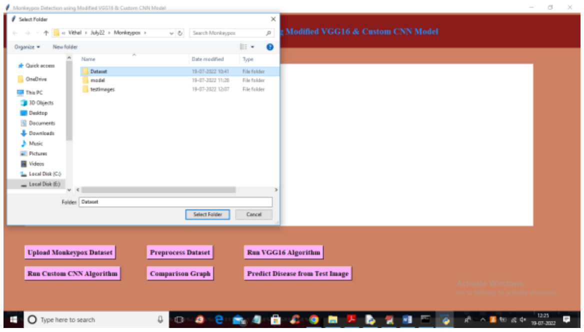


Fig 11 F1Score



Name	Date modified	Type
Dataset	19-07-2022 10:41	File folder
model	19-07-2022 11:28	File folder
testImages	19-07-2022 12:07	File folder

Folder: Dataset

Fig 12 Input image folder

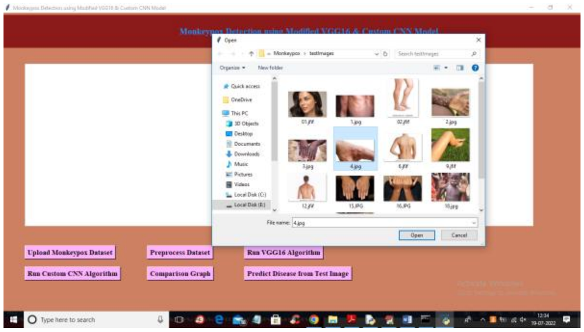
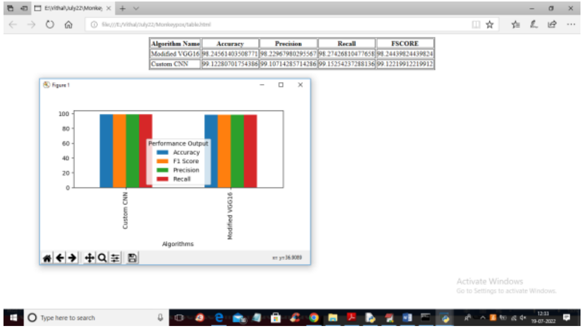


Fig 13 Upload input image

Fig 14 Predict result for given input\



Algorithm Name	Accuracy	Precision	Recall	FSCORE
Modified VGG16	98.24561403508771	98.22967980295567	98.27426810477658	98.24398244398824
Custom CNN	99.12280701754386	99.10714285714286	99.15254237288136	99.12199122199122

Fig:15 accuracy graph

5. CONCLUSION

The study shows that DL models, especially Modified MobileNetV2 and VGG19, are quite good at finding Monkeypox. The advent of Hybrid Modified Extension MobileNetV2, which combines Random Forest with feature extraction, makes predictions even better, reaching 100% accuracy. The model is more transparent thanks to Local Interpretable Model-Agnostic Explanations (LIME), which makes it simpler to find the most important aspects that help with illness identification. The results show that deep learning-based diagnostic tools might be useful in places with few resources since they are efficient, cheap, and easy to scale up compared to traditional lab-based procedures. This method can help a lot with early detection and containment efforts, which lowers the danger of large epidemics.

6. FUTURE SCOPE

The work gives us useful information on how we might use transfer learning models and explainable AI to develop safe and reliable Monkeypox illness diagnosis algorithms. The study shows that transfer learning techniques may be used in illness diagnosis and are still useful, which opens up new areas for research and improvement. The results of this study can help future research and real-world efforts to stop and control diseases, especially in locations that are hard to reach and don't have many resources. It will be interesting to see how the proposed model works on a big, multiclass dataset in the near future. This might help us learn more about Monkeypox and make diagnoses more accurate. Transfer learning and explainable AI can help make illness diagnosis models better and more reliable in healthcare. The results and in-depth analysis of this study might help future researchers and professionals look at how transfer learning models and explainable AI could be able to help in diagnosing Monkeypox sickness.

REFERENCES

[1] L. V. Patrono, K. Pléh, L. Samuni, M. Ulrich, C. Röthemeier, A. Sachse, S. Muschter, A. Nitsche, E.

Couacy-Hymann, C. Boesch, R. M. Wittig, S. Calvignac-Spencer, and F. H. Leendertz, ‘‘Monkeypox virus emergence in wild chimpanzees reveals distinct clinical outcomes and viral diversity,’’ Nature Microbiol., vol. 5, no. 7, pp. 955–965, Apr. 2020.

[2] S. Parker, A. Nuara, R. M. L. Buller, and D. A. Schultz, ‘‘Human monkeypox: An emerging zoonotic disease,’’ Future Microbiol., vol. 2, no. 1, pp. 17–34, Feb. 2007.

[3] S. Wong, S. Lau, P. Woo, and K.-Y. Yuen, ‘‘Bats as a continuing source of emerging infections in humans,’’ Rev. Med. Virol., vol. 17, no. 2, pp. 67–91, Mar. 2007.

[4] Centres for Disease Control and Prevention. (2022). Monkeypox Signs and Symptoms. [Online]. Available: https://www.cdc.gov/poxvirus/mpox/symptoms/index.html

[5] M. Reynolds, A. McCollum, B. Nguete, R. S. Lushima, and B. Petersen, ‘‘Improving the care and treatment of monkeypox patients in low-resource settings: Applying evidence from contemporary biomedical and smallpox biodefense research,’’ Viruses, vol. 9, no. 12, p. 380, Dec. 2017.

[6] A. S. Fauci, ‘‘Emerging and reemerging infectious diseases: The perpetual challenge,’’ Academic Med., vol. 80, no. 12, pp. 1079–1085, Dec. 2005.

[7] Michaeleen Doucleff. (2022). Scientists Warned us About Monkeypox in 1988. Here’s Why they were Right. [Online]. Available: https://t.ly/QbTJ

[8] M. Dwivedi, R. G. Tiwari, and N. Ujjwal, ‘‘Deep learning methods for early detection of monkeypox skin lesion,’’ in Proc. 8th Int. Conf. Signal Process. Commun. (ICSC), Dec. 2022, pp. 343–348.

[9] Centres for Disease Control and Prevention. (2022). 2022 Monkeypox and Orthopoxvirus Outbreak Global Map. [Online]. Available: https://www.who.int/emergencies/situations/monkeypox-oubreak-2022

[10] M. M. Ahsan, M. R. Uddin, M. Farjana, A. N. Sakib, K. Al Momin, and S. A. Luna, ‘‘Image data collection and implementation of deep learningbased model in detecting monkeypox disease using modified VGG16,’’ 2022, arXiv:2206.01862.

[11] M. M. Ahsan, M. R. Uddin, and S. A. Luna, ‘‘Monkeypox image data collection,’’ 2022, arXiv:2206.01774.

[12] Centres for Disease Control and Prevention. (2022). Monkeypox and Smallpox Vaccine. [Online]. Available: https://t.ly/e3b5

[13] F. Yasmin, M. M. Hassan, S. Zaman, S. T. Aung, A. Karim, and S. Azam, ‘‘A forecasting prognosis of the monkeypox outbreak based on a comprehensive statistical and regression analysis,’’ Computation, vol. 10, no. 10, p. 177, Oct. 2022.

[14] D. L. Heymann, M. Szczeniowski, and K. Esteves, ‘‘Re-emergence of monkeypox in africa: A review of the past six years,’’ Brit. Med. Bull., vol. 54, no. 3, pp. 693–702, Jan. 1998.

[15] T. Wawina-Bokalanga, N. Sklenovska, B. Vanmechelen, M. Bloemen, V. Vergote, L. Laenen, E. Andre, M. Van Ranst, J.-J. T. Muyembe, and P. Maes, ‘‘An accurate and rapid real-time PCR approach for human monkeypox virus diagnosis,’’ medRxiv, 2022.

[16] Centres for Disease Control and Prevention. (2022). Diagnostic Tests. [Online]. Available: https://www.cdc.gov/dpdx/diagnosticprocedures/index.html

[17] M. M. Ahsan, M. T. Ahad, F. A. Soma, S. Paul, A. Chowdhury, S. A. Luna, M. M. S. Yazdan, A. Rahman, Z. Siddique, and P. Huebner, ''Detecting SARS-CoV-2 from chest X-ray using artificial intelligence,'' IEEE Access, vol. 9, pp. 35501–35513, 2021.

[18] S. Upadhayay, R. Arthur, D. Soni, P. Yadav, U. Navik, R. Singh, T. G. Singh, and P. Kumar, ''Monkeypox infection: The past, present, and future,'' Int. Immunopharmacology, vol. 113, Dec. 2022, Art. no. 109382.

[19] M. M. Ahsan, T. E. Alam, T. Trafalis, and P. Huebner, ''Deep MLPCNN model using mixed-data to distinguish between COVID-19 and non-COVID-19 patients,'' Symmetry, vol. 12, no. 9, p. 1526, Sep. 2020.

[20] M. M. Ahsan and Z. Siddique, ''Machine learning-based heart disease diagnosis: A systematic literature review,'' Artif. Intell. Med., vol. 128, Jun. 2022, Art. no. 102289.

[21] M. M. Ahsan, S. A. Luna, and Z. Siddique, ''Machine-learning-based disease diagnosis: A comprehensive review,'' Healthcare, vol. 10, no. 3, p. 541, Mar. 2022.

[22] S. Chae, S. Kwon, and D. Lee, ''Predicting infectious disease using deep learning and big data,'' Int. J. Environ. Res. Public Health, vol. 15, no. 8, p. 1596, Jul. 2018.

[23] R. Arias and J. Mejía, ''Varicella zoster early detection with deep learning,'' in Proc. IEEE Eng. Int. Res. Conf. (EIRCON), Oct. 2020, pp. 1–4.

[24] S. Bhadula, S. Sharma, P. Juyal, and C. Kulshrestha, ''Machine learning algorithms based skin disease detection,'' Int. J. Innov. Technol. Exploring Eng., vol. 9, no. 2, pp. 4044–4049, 2019.

[25] K. Sriwong, S. Bunrit, K. Kerdprasop, and N. Kerdprasop, ''Dermatological classification using deep learning of skin image and patient background knowledge,'' Int. J. Mach. Learn. Comput., vol. 9, no. 6, pp. 862–867, Dec. 2019.

[26] C. Sitaula and T. B. Shahi, ''Monkeypox virus detection using pre-trained deep learning-based approaches,'' J. Med. Syst., vol. 46, no. 11, pp. 1–9, Oct. 2022.

[27] V. H. Sahin, I. Oztel, and G. Y. Oztel, ''Human monkeypox classification from skin lesion images with deep pre-trained network using mobile application,'' J. Med. Syst., vol. 46, no. 11, pp. 1–10, Oct. 2022.

[28] K. D. Akin, C. Gurkan, A. Budak, and H. Karataş, ''Classification of monkeypox skin lesion using the explainable artificial intelligence assisted convolutional neural networks,'' Avrupa Bilim ve Teknoloji Dergisi, vol. 40, pp. 106–110, Sep. 2022.

[29] F. Yasmin, M. M. Hassan, M. Hasan, S. Zaman, C. Kaushal, W. El-Shafai, and N. F. Soliman, ''PoxNet22: A fine-tuned model for the classification of monkeypox disease using transfer learning,'' IEEE Access, vol. 11, pp. 24053–24076, 2023.

[30] A. Shah, ''Monkeypox skin lesion classification using transfer learning approach,'' in Proc. IEEE

Bombay Sect. Signature Conf. (IBSSC), Dec. 2022, pp. 1–5.

[31] K. Simonyan and A. Zisserman, ''Very deep convolutional networks for large-scale image recognition,'' 2014, arXiv:1409.1556.

[32] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, ''Inception-v4, inception-ResNet and the impact of residual connections on learning,'' in Proc. 31st AAAI Conf. Artif. Intell., 2017, pp. 4278–4284.

[33] T. Akiba, S. Suzuki, and K. Fukuda, ''Extremely large minibatch SGD: Training ResNet-50 on ImageNet in 15 minutes,'' 2017, arXiv:1711.04325.

[34] W. Islam, M. Jones, R. Faiz, N. Sadeghipour, Y. Qiu, and B. Zheng, ''Improving performance of breast lesion classification using a ResNet50 model optimized with a novel attention mechanism,'' Tomography, vol. 8, no. 5, pp. 2411–2425, Sep. 2022.

[35] W. Islam, G. Danala, H. Pham, and B. Zheng, ''Improving the performance of computer-aided classification of breast lesions using a new feature fusion method,'' Proc. SPIE, vol. 12033, pp. 84–91, Apr. 2022.

[36] K. He, X. Zhang, S. Ren, and J. Sun, ''Deep residual learning for image recognition,'' in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2016, pp. 770–778.

[37] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, ''MobileNetV2: Inverted residuals and linear bottlenecks,'' in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2018, pp. 4510–4520.

[38] M. K. Islam, M. S. Ali, M. S. Miah, M. M. Rahman, M. S. Alam, and M. A. Hossain, ''Brain tumor detection in MR image using superpixels, principal component analysis and template based K-means clustering algorithm,'' Mach. Learn. With Appl., vol. 5, Sep. 2021, Art. no. 100044.

[39] M. S. Ali, M. K. Islam, J. Haque, A. A. Das, D. S. Duranta, and M. A. Islam, ''Alzheimer's disease detection using m-Random forest algorithm with optimum features extraction,'' in Proc. 1st Int. Conf. Artif. Intell. Data Anal. (CAIDA), Apr. 2021, pp. 1–6.

[40] S. K. M S. Islam, M. A. Al Nasim, I. Hossain, D. M. A. Ullah, D. K. D. Gupta, and M. M. H. Bhuiyan, ''Introduction of medical imaging modalities,'' 2023, arXiv:2306.01022.

[41] M. S. Ali, M. S. Miah, J. Haque, M. M. Rahman, and M. K. Islam, ''An enhanced technique of skin cancer classification using deep convolutional neural network with transfer learning models,'' Mach. Learn. with Appl., vol. 5, Sep. 2021, Art. no. 100036.

[42] M. Hassan, S. Zaman, S. Mollick, M. M. Hassan, M. Raihan, C. Kaushal, and R. Bhardwaj, ''An efficient Apriori algorithm for frequent pattern in human intoxication data,'' Innov. Syst. Softw. Eng., vol. 19, pp. 1–9, Oct. 2023.

[43] A. Kwasigroch, A. Mikolajczyk, and M. Grochowski, ''Deep neural networks approach to skin lesions classification—A comparative analysis,'' in Proc. 22nd Int. Conf. Methods Models Autom. Robot. (MMAR), Aug. 2017, pp. 1069–1074.

[44] M. I. Hasan, M. S. Ali, M. H. Rahman, and M. K. Islam, ''Automated detection and characterization

of colon cancer with deep convolutional neural networks,'' J. Healthcare Eng., vol. 2022, pp. 1–12, Aug. 2022.

[45] F. Chollet, ''Building powerful image classification models using very little data,'' Keras Blog, vol. 5, pp. 90–95, Jun. 2016.

[46] L. Perez and J. Wang, ''The effectiveness of data augmentation in image classification using deep learning,'' 2017, arXiv:1712.04621.

[47] M. M. Ahsan, M. R. Uddin, M. S. Ali, M. K. Islam, M. Farjana, A. N. Sakib, K. A. Momin, and S. A. Luna, ''Deep transfer learning approaches for monkeypox disease diagnosis,'' Expert Syst. Appl., vol. 216, Apr. 2023, Art. no. 119483.

[48] M. K. Islam, M. S. Ali, M. M. Ali, M. F. Haque, A. A. Das, M. M. Hossain, D. S. Duranta, and M. A. Rahman, ''Melanoma skin lesions classification using deep convolutional neural network with transfer learning,'' in Proc. 1st Int. Conf. Artif. Intell. Data Anal. (CAIDA), Apr. 2021, pp. 48–53.

[49] M. M. Ahsan, Y. Li, J. Zhang, M. T. Ahad, and M. M. S. Yazdan, ''Face recognition in an unconstrained and real-time environment using novel BMC-LBPH methods incorporates with DJI vision sensor,'' J. Sensor Actuator Netw., vol. 9, no. 4, p. 54, Nov. 2020.

[50] M. M. Ahsan, Y. Li, J. Zhang, M. T. Ahad, and K. D. Gupta, ''Evaluating the performance of eigenface, fisherface, and local binary pattern histogram-based facial recognition methods under various weather conditions,'' Technologies, vol. 9, no. 2, p. 31, Apr. 2021.

[51] N. A. Baghdadi, A. Malki, S. F. Abdelaliem, H. M. Balaha, M. Badawy, and M. Elhosseini, ''An automated diagnosis and classification of COVID-19 from chest CT images using a transfer learning-based convolutional neural network,'' Comput. Biol. Med., vol. 144, May 2022, Art. no. 105383.

[52] M. M. Najafabadi, F. Villanustre, T. M. Khoshgoftaar, N. Seliya, R. Wald, and E. Muharemagic, ''Deep learning applications and challenges in big data analytics,'' J. Big Data, vol. 2, pp. 1–21, 2015.

[53] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, ''An image is worth 16×16 words: Transformers for image recognition at scale,'' 2020, arXiv:2010.11929.

[54] S. Khan, M. Naseer, M. Hayat, S. W. Zamir, F. S. Khan, and M. Shah, ''Transformers in vision: A survey,'' ACM Comput. Surv., vol. 54, no. 10s, pp. 1–41, Jan. 2022.

[55] A. Krizhevsky, I. Sutskever, and G. E. Hinton, ''ImageNet classification with deep convolutional neural networks,'' Commun. ACM, vol. 60, no. 6, pp. 84–90, May 2017.

[56] A. Narin, C. Kaya, and Z. Pamuk, ''Automatic detection of coronavirus disease (COVID-19) using X-ray images and deep convolutional neural networks,'' Pattern Anal. Appl., vol. 24, no. 3, pp. 1207–1220, Aug. 2021.

[57] K. Jankowsky and U. Schroeders, ''Validation and generalizability of machine learning prediction models on attrition in longitudinal studies,'' Int. J.

Behav. Develop., vol. 46, no. 2, pp. 169–176, Mar. 2022.

[58] S. Khedkar et al., ‘‘Explainable AI in healthcare,’’ in Proc. 2nd Int. Conf. Adv. Sci. Technol. (ICAST), Apr. 2019.

[59] M. T. Ribeiro, S. Singh, and C. Guestrin, ‘‘‘Why should I trust you?’ Explaining the predictions of any classifier,’’ in Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, 2016, pp. 1135–1144.

[60] C. Kaushal, M. K. Islam, S. A. Althubiti, F. Alenezi, and R. F. Mansour, ‘‘A framework for interactive medical image segmentation using optimized swarm intelligence with convolutional neural networks,’’ Comput. Intell. Neurosci., vol. 2022, pp. 1–21, Aug. 2022.