medRxiv preprint doi: https://doi.org/10.1101/2024.11.26.24318006; this version posted November 29, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted medRxiv a license to display the preprint in perpetuity.
It is made available under a CC-BY-ND 4.0 International license .

H-MpoxNet: A Hybrid Deep Learning Framework for Mpox Detection from Image Data

Sajal Chakroborty[a]

[a] *Worcester Polytechnic Institute, Worcester, Massachusetts, USA, email:schakroborty@wpi.edu , sajal.math@yahoo.com*

ARTICLE INFO

Keywords:
Mpox
DenseNet
MobileNet
Inception
Xception
Zoonotic Disease
Inception-Residual network
Machine Learning classifiers
Hybrid Deep Learning framework

ABSTRACT

Infectious diseases can create significant global threats to public health and economic stability by creating pandemics. SARS-CoV-2 is a recent example. Early detection of infectious diseases is crucial to prevent global outbreaks. Mpox, a contagious viral disease first detected in humans in 1970, has experienced multiple outbreaks in recent decades, which emphasizes the development of tools for its early detection. In this paper, we develop a hybrid deep learning framework for Mpox detection. This framework allows us to construct hybrid deep learning models combining deep learning architectures as a feature extraction tool with Machine Learning classifiers and perform a comprehensive analysis of Mpox detection from image data. Our best-performing model consists of MobileNetV2 with LightGBM classifier, which achieves an accuracy of 91.49%, 91.87% weighted precision, 91.49% weighted recall, 91.51% weighted F1-score and Matthews Correlation Coefficient score of 0.83.

1. Introduction

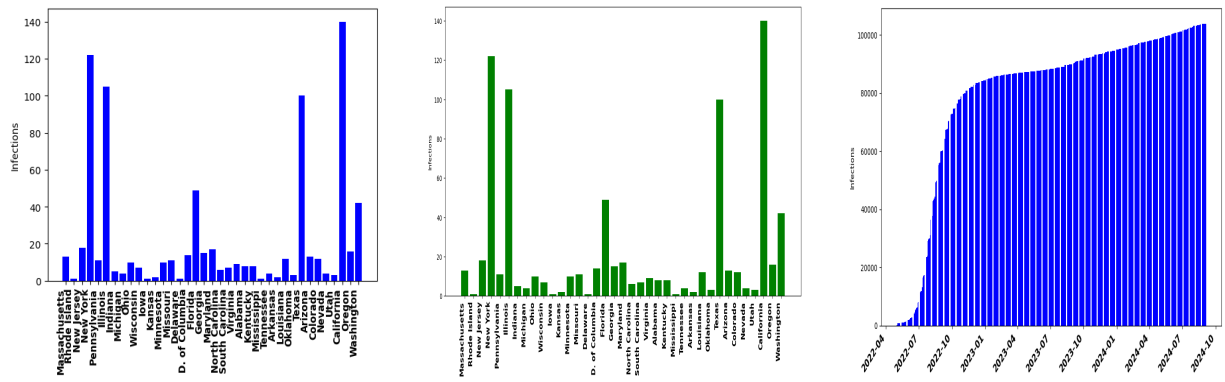
Mpox, also known as monkeypox (MPX), is caused by the Mpox virus (MPXV), is a zoonotic disease, more specifically an infectious skin disease that can spread between animals and humans. It is a 200-250 nm brick-like or ovoid shaped double helix DNA virus that belongs to the Orthopoxvirus genus, Proxviriade family, and Chordopoxvirinae subfamily (Moore et al. (2021)). Along with MPXV, variola major virus, which causes smallpox, variola minor virus (also known as variola alastrim) from the same genus, also infects the human body (Petersen et al. (2019)). MPXV can be transmitted into the human body in several ways, from animal to human and human to human (Vaughan et al. (2020)). Moreover, it can be transmitted by contact with contaminated objects, a patient's droplets, body fluids, and lesions (Vaughan et al. (2020)). After exposure to MPXV, it takes 3-17 days to develop symptoms, which can last nearly 2-5 weeks (Centers for Disease Control and Prevention (2024a)). A few symptoms of Mpox are fever, swollen lymph nodes, headache, muscle aches, and backache (Centers for Disease Control and Prevention (2024a)).

The first human case caused by MPXV was reported in 1970 (Ladnyj et al. (1972); Gessain et al. (2022)). In 2022-23, there was a global outbreak of Mpox (Ianache et al. (2024)). Before that, between 1970 and 2017, several outbreaks occurred in central and west Africa (Organization et al. (2022); Breman et al. (1980); Foster et al. (1972)). Clade I, previously known as the Congo Basin Clade or Central African Clade, and Clade II, previously known as West African Clade, were responsible for the outbreaks in Central and West Africa (Eke (1972); Brown and Leggat (2016)). The first outbreak outside Africa occurred in the USA in 2003 (McCollum and Damon (2014); Brown and Leggat (2016)). In 2023, 809 positive cases were reported in different states of the USA, and 1,945 have been reported up to September 09, 2024 (Centers for Disease Control and Prevention (2024b)). In Figure 1, we present a scenario based on data collected from Centers for Disease Control and Prevention (2024b) and Ritchie et al. (2024) respectively for different states in the USA and Worldwide. From these reported cases, it is visible that developing tools for the early detection of Mpox has evolved into a significant research problem.

Biomedical images have been a valuable tool in disease diagnosis for an extended period. In general, images have found wide-ranging applications in various fields, including health care, for example, brain tumor detection from MRI images (Bhattacharyya and Kim (2011); Amin et al. (2020); Anantharajan et al. (2024); Arumugam et al. (2024); Khan et al. (2024)), skin cancer detection from histopathological images (Lu and Mandal (2015); Esteva et al. (2017); Kimeswenger et al. (2021); Akilandasowmya et al. (2024); Ali et al. (2021); Strzelecki et al. (2024)); in agriculture, such as plant disease detection, for example, wheat disease detection (Franke and Menz (2007); Lu et al. (2017); Goyal et al. (2021); Verma et al. (2024)); in disaster management such as flood detection by satellite images (Kussul et al. (2008); Vanama et al. (2021); Composto et al. (2024)); in retail and E-commerce (Li and Li (2019); Chen et al. (2021);

NOTE: This preprint reports new research that has not been certified by peer review and should not be used to guide clinical practice.

medRxiv preprint doi: https://doi.org/10.1101/2024.11.26.24318006; this version posted November 29, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted medRxiv a license to display the preprint in perpetuity.
It is made available under a CC-BY-ND 4.0 International license .



(a) (b) (c)

Figure 1: Mpox reported cases in the USA and Worldwide. 1a and 1b illustrate the number of infected cases in 2023 and 2024, respectively, in different states of the USA, and 1c shows cumulative infected cases reported globally up to September 2024.

Liang et al. (2016)). Mathematically, one can model a grayscale image as a function $f : \mathbb{R}^2 \to [0, 255]$. Similarly, a color image can be modeled as $f : \mathbb{R}^2 \to [0, 255]^3$. Note that $[0, 255]$ denotes the intensity values of the image.

Recently, biomedical images have become very useful to detect infectious diseases such as COVID-19 using chest X-ray images (Sailunaz et al. (2023); Bhattacharya et al. (2021); Jiang et al. (2020); Ebenezer et al. (2022); Ismael and Şengür (2021)). It is important to note that a commonly used diagnostic tool for infectious diseases is Polymerase Chain Reaction (PCR) test (Fredricks and Relman (1999)). It provides precise genetic analysis. However, PCR has risks of contamination, and limitations in detecting large or unknown sequences (Yang and Rothman (2004)). Moreover, it requires specialized equipment and expertise, making it less accessible in some resource-limited settings (Yang and Rothman (2004)).

Deep Learning (DL) techniques have become an alternative to detect infectious diseases. For a detailed review, one can see Sharma and Guleria (2024), Ibrahim et al. (2024), Bhatele et al. (2024), Chen et al. (2024), Foltz et al. (2024). While PCR is more likely inclined toward the lab resources and has a risk of contamination, DL techniques are mainly focused on biomedical images. DL techniques are data driven and a subclass of artificial intelligence (Huang et al. (2019)). It can predict patterns from large and complex datasets (Huang et al. (2019)). It involves training neural networks that can mimic the neural orientation of the human brain (Huang et al. (2019)). Most importantly, once the model is trained, it can be deployed on a simpler system to predict, classify, and generate output (Huang et al. (2019)).

In addition to DL, there are Machine Learning (ML) tools for classification. For example, Support vector machine (SVM) (Suthaharan and Suthaharan (2016b)), Logistic regression (Musa (2013)), Random Forest (RF) (Breiman (2001); Biau and Scornet (2016); Rashidi et al. (2019)), Naïve bayes (NB) (Webb et al. (2010)). Due to computational efficiency these classifiers have found wide ranging applications in various fields. For detailed review see Pereira et al. (2009), Narudin et al. (2016), and Abro et al. (2021).

Furthermore, One can study the dynamics of any infectious disease by developing models using a system of ordinary or partial differential equations. A simple framework for these models is the SIR (Susceptible, Infectious, Recovered) model, given by (Hattaf and Dutta (2020)):

$$\begin{aligned} \frac{dS}{dt} &= \Lambda - a\frac{IS}{N} \\ \frac{dI}{dt} &= a\frac{IS}{N} - bI \\ \frac{dR}{dt} &= bI, \end{aligned} \tag{1}$$

with initial conditions $S(0) = S_0, I(0) = I_0, R(0) = R_0$. Here S, I, and R are the number of individuals in the Susceptible, Infectious, and Recovered compartments, Λ represents the recruitment number in the Susceptible compartment, a is the infection or disease transmission rate from susceptible to infection, and b denotes the recovery rate. Note that $t > 0$ and the total population is given by:

$$N(t) = S(t) + I(t) + R(t). \tag{2}$$

medRxiv preprint doi: https://doi.org/10.1101/2024.11.26.24318006; this version posted November 29, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted medRxiv a license to display the preprint in perpetuity.
It is made available under a CC-BY-ND 4.0 International license .

Using this model framework, one can compute the basic reproduction number $\mathcal{R}_0$, which is a threshold that determines whether the disease will spread or die out.

Due to the mathematical elegance, this model is widely popular and found wide ranging applications in the published literature. For example, consider the SEIR model, an extended version of (1) proposed by Li (2011) to study Malaria transmission. This model incorporates the life stages of mosquitoes that reflect their biological life cycles and malaria transmission dynamics between humans and mosquitoes. Etbaigha et al. (2018) developed a compartmental model to study influenza A virus (IAV) transmission within a swine farm. The model investigated different stages of infection and immunity within the swine population. Ahmed et al. (2021) proposed a SEIAPH (Susceptible, Exposed, Infectious, Asymptotic, Pre-symptomatic, Symptomatic hospitalized, and Recovered) model that reflects a critical aspect of SARS-CoV-2 virus transmission concerning the role of asymptomatic carriers. Mahata et al. (2022) proposed an advanced model incorporating fractional order derivatives and vaccination strategies in the classical SEIR model. Moreover, they introduced optimal control strategies and provided stability analysis of the model. Furthermore, they provided a comparison between the classical and fractional-order SEIR models.

Recently, this infectious disease model has been studied to analyze the dynamics of Mpox and predict the virus's spread. Liu et al. (2023) proposed a SEIR-type model for continuous monitoring and early intervention of the spread of Mpox virus. They proposed a robust model and performed sensitivity analyses for different parameters such as infection rate and recovery rate. Most importantly, they estimated the basic reproduction number for 2022 outbreak to investigate how rapidly the virus could spread. Batiha et al. (2023) investigated the dynamics of Mpox outbreak using fractional-order SEIR model. They analyzed the effects of vaccination in the transmission and mitigation of Mpox virus. Betti et al. (2023) developed a model focusing on pair formation that reflects how close contact with individuals contributes to the spread of Mpox. Additionally, they investigated recovery dynamics to observe how quickly infected individuals recover and how this recovery rate affects the overall spread of the infection. Using infectious disease modeling and statistical techniques, Zhang et al. (2024) forecasted the global spread and trajectory of the Mpox virus. They emphasized identifying key factors responsible for the spread of MPXV. By analyzing the current data, they predicted the future scenario of the outbreak. Kaftan et al. (2024) performed a comprehensive analysis of the 2022 Mpox outbreak in New York City. They analyzed and compared different mathematical models, including SEIR, and evaluated their effectiveness in predicting and forecasting outbreak patterns. Molla et al. (2023) has provided a comprehensive review on infectious disease modeling to investigate the Mpox transmission.

Ali et al. (2022) created an image dataset to detect and analyze Mpox with DL algorithms. The dataset consists of images with a resolution of $224 \times 224 \times 3$ and allows the implementation of DL models. Ali et al. (2022) applied three pre-trained models, VGG-16, InceptionV3, and ResNet-50, to classify Mpox images. In addition to that, they combined these DL algorithms by employing majority voting and proposed an ensemble model. Among these models, ResNet-50 achieved 82.96% accuracy, 87% precision, 83% recall, and 84% F1 score, while other models, including the ensemble, produced lower scores in every metric. Using the same dataset, Sahin et al. (2022) developed a DL framework tailored for mobile based applications to diagnose Mpox lesions. They obtained 91.11% accuracy, 90% precision, 90% recall, and 90% F1 score. Sitaula and Shahi (2022) proposed an ensemble based DL framework focused not only on Mpox detection but also on classifying other Skin diseases. Their dataset was curated with samples of different skin diseases. Initially, they implemented 13 different DL techniques, then ensembled two best-performing algorithms—Xception and DenseNet-169 and achieved 87.13% accuracy, 85.44% precision, 85.47% recall, and 85.40% F1 score. Dwivedi et al. (2022) explored ResNet-50, EfficientNetb3, and EfficientNetb7 on a dataset that contains 160 Mpox lesions, 178 chickenpox, 54 cowpox, 358 small pox, and 50 healthy skins. They obtained 87% accuracy, 92% precision, 87% recall, and 90% F1 score by EfficientNetb3.

Uysal (2023) proposed a hybrid DL framework for Mpox detection. Unlike developing an ensemble model employing majority voting, they combined the two models with the highest accuracy with the Long Short-term Memory (LSTM) model and obtained 87% accuracy, 84% precision, 87% recall, and 85% F1 score. Nayak et al. (2023) tested GoogleNet, ReseNet-18, ResNet-50, ResNet-101, and SquezeNet for multi-classification and Mpox detection and obtained 91.19% average accuracy. Explainable Artificial Intelligence (XAI) is crucial in Medical research. Nayak et al. (2023) introduced XAI to make the DL model more interpretable. Gupta et al. (2023) developed a blockchain enabled DL framework for early detection of Mpox lesions. They obtained 98% accuracy using a dataset

medRxiv preprint doi: https://doi.org/10.1101/2024.11.26.24318006; this version posted November 29, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted medRxiv a license to display the preprint in perpetuity.
It is made available under a CC-BY-ND 4.0 International license .

with 1905 images. Sorayaie Azar et al. (2023) implemented ResNetV2, InceptionV3, ResNet152V2, VGG-16, VGG-19, Xception, DenseNet-201 on a dataset that consists of 43 Mpox, 47 Chickenpox, 27 Measles, 54 normal images. They found that DenseNet-201 outperformed other models and produced 95.18% accuracy. However, they have augmented the sample size ten times before implementing transfer learning models. Akula and Pushkar (2023) investigated seven different pre-trained transfer learning models and achieved an accuracy of 89.24% with the Xception model.

Recently, Ahsan et al. (2024) proposed a modified VGG-16 and ResNet-50 model to classify Mpox. To ensure data privacy, they have used a federated learning strategy. Meena et al. (2024) used VGG19, Xception, InceptionV3 transfer learning models on augmented dataset created by Ali et al. (2021), which includes, 1428 Mpox and 1764 others Skin lesions images. They achieved 98% accuracy by using InceptionV3 model. Yolcu Oztel (2024) explored Vision Transformers (VITs) and Convolutional Neural Networks (CNNs) for Mpox lesion analysis. They proposed an ensemble learning strategy using the bagging ensemble technique and obtained 81.91% strategy.

In this paper, we predict Mpox from image dataset. Our contributions include:

- We develop a hybrid Deep learning (DL) framework combining a pre-trained DL architecture with a Machine Learning (ML) classifier. The pre-trained DL architecture will be used for feature extraction from image data. After that, the ML model will be trained for Mpox prediction.
- We develop several hybrid DL models using our DL framework and perform a comprehensive analysis of Mpox prediction. To evaluate the model, we compute accuracy, precision, weighted precision, recall, weighted recall, F1-score, weighted F1-score, and Matthews Correlation Coefficient (MCC) score.
- Finally, we compare our best-performing model with previously published results to highlight its effectiveness in Mpox prediction.

We organize this paper as follows: In section 2, we develop a hybrid DL framework. Next, we present our experimental resources along with a brief discussion about our dataset, train-test splitting procedure, and data augmentation strategy in section 3. In the same section, we also discuss the pre-trained DL architectures that we have used in this paper and different ML tools. After that, in section 4, we discuss our results and compare our best performing model with previously published results. Finally, we conclude our work.

2. Hybrid Deep Learning Framework for Mpox Detection

Consider the dataset $S = \left\{ \left(\mathcal{X}_i, \mathcal{Y}_i \right) \right\}_{i=1}^{n}$, where $\mathcal{X}_i$ denotes the ith image and $\mathcal{Y}_i$ represents its corresponding image label. We develop the framework by combining a pre-trained DL architecture and an ML classifier in the following steps:

Step 1 (Preprocessing): In this initial phase, we clean the dataset through various processes, including noise reduction, handling mislabeled images, and removing duplicate images to ensure the dataset is prepared for optimal model performance.

Step 2 (Train-test split): We partition the given dataset S into two distinct subsets, S_{Train} and S_{Test}, such that

$$S = S_{\text{Train}} \cup S_{\text{Test}}, \text{ and } S_{\text{Train}} \cap S_{\text{Test}} = \phi,$$

where S_{Train} denotes the training set containing 80% of the sample and S_{Test} denotes the test set containing 20% of the samples from the dataset S. We perform data augmentation on S_{Train}.

Step 3 (Feature extraction): In this step, using a pre-trained DL architecture $\mathcal{F}$, we extract features from each image as follows:

$$x_j^{\text{Train}} = \mathcal{F}\left(\mathcal{X}_j^{\text{Train}}\right), \; j = 1, 2, \ldots, N \tag{3}$$

$$x_k^{\text{Test}} = \mathcal{F}\left(\mathcal{X}_k^{\text{Test}}\right), \; k = 1, 2, \ldots, M, \tag{4}$$

where x_j^{Train} and x_k^{Test} represent the features extracted from $\mathcal{X}_j^{\text{Train}} \in S_{\text{Train}}$ and $\mathcal{X}_k^{\text{Test}} \in S_{\text{Test}}$ respectively.

Note that each extracted feature is a tensor of dimension $l \times p \times q$, where l and p are the height and width of the feature map and q denotes the number of feature channels. In order to train a ML classifier, we have reshaped each of these features into a d-dimensional vector where $d = l \times p \times q$. The reshaping procedure is performed as follows:

$$z_j^{\text{Train}} = \text{reshape}\left(x_j^{\text{Train}}, d\right), \tag{5}$$

$$z_k^{\text{Test}} = \text{reshape}\left(x_k^{\text{Test}}, d\right), \tag{6}$$

where $j = 1, 2, \ldots, N$, and $k = 1, 2, \ldots, M$.

Step 4 (Train classification model): Using these features z_j^{Train}, we train a classification model $\mathcal{G}$. After that, we test the model using the features z_k^{Test} and predict the labels $\hat{\mathcal{Y}}_k^{\text{Test}}$ as follows:

$$\hat{\mathcal{Y}}_k^{\text{Test}} = \mathcal{G}(z_k^{\text{Test}}), \; k = 1, 2, \ldots, M. \tag{7}$$

medRxiv preprint doi: https://doi.org/10.1101/2024.11.26.24318006; this version posted November 29, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted medRxiv a license to display the preprint in perpetuity.
It is made available under a CC-BY-ND 4.0 International license .

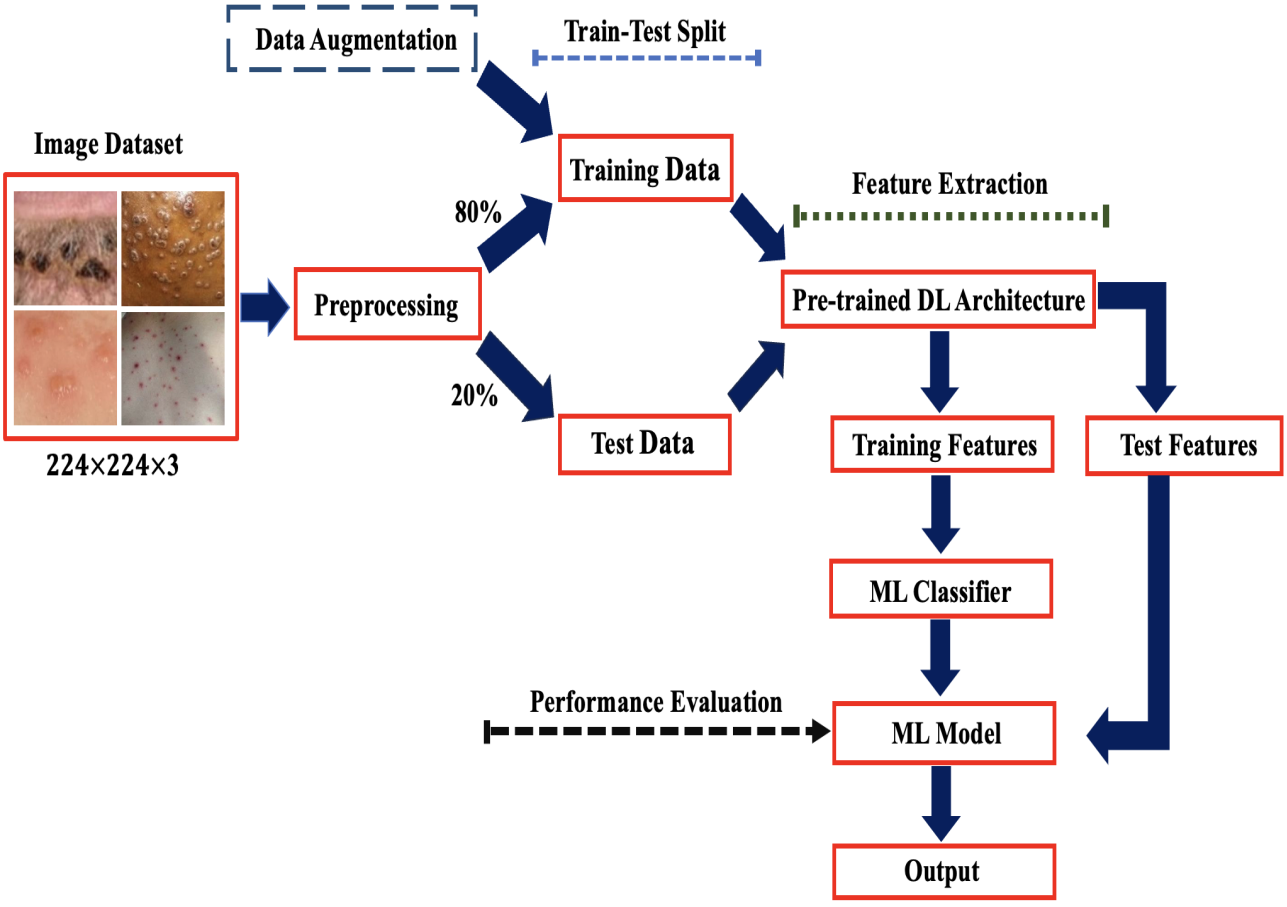


Figure 2: A hybrid Deep Learning framework for Mpox classification, combining a pre-trained Deep Learning architecture and Machine Learning classifier.

Step 5 (Model Evaluation): Finally, we evaluate the hybrid model using standard metrics such as accuracy, precision, weighted precision, recall, weighted recall, F1 score, weighted F1 score, and Matthews Correlation Coefficient (MCC) score. We discuss these metrics in subsection 3.4.

In Figure 2, we illustrate this framework in detail.

3. Resources for Constructing Hybrid Models

In this section, we discuss our dataset, data augmentation strategy, DL architectures, and ML classification models, which we have used to construct hybrid models.

3.1. Data Description

In this paper, we have used a publicly available dataset for Mpox detection created by Ali et al. (2021). The dataset contains images with a resolution of $224 \times 224 \times 3$. It is divided into two categories–Mpox and non-Mpox, where the latter includes images of other diseases–chickenpox and measles. In Table 1, we present a summary of the dataset, including the number of images in each category.

Table 1
Summary of the Dataset

Category	Number of Images
Mpox	102
Others	126
Total	**228**

3.1.1. Train-Test Split and Data Augmentation

We split the dataset into two subsets– Train set, which contains 80%, and Test set, which includes 20% of the samples from the original dataset. We present a detailed summary in Table 2.

medRxiv preprint doi: https://doi.org/10.1101/2024.11.26.24318006; this version posted November 29, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted medRxiv a license to display the preprint in perpetuity.
It is made available under a CC-BY-ND 4.0 International license .

Table 2
Summary of the Train and Test data

	Train set	Test set
Mpox	81	21
Others	100	26
Total	**181**	**47**

We have performed data augmentation in the training dataset. Since the dataset is imbalanced, one can observe it from Table 1 and Table 2; we have balanced the number of Mpox and Other images using standard augmentation strategies. In Table 3, we present these in detail.

3.2. Pre-trained DL Architectures

We have used several pre-trained DL architectures for feature extraction, including DenseNet, MobileNet, Inception, Inception-Residual Network, and Xception. We discuss them briefly in the following subsections.

3.2.1. DenseNet

All the layers of this architecture are connected in a feedforward manner, where all the features from the preceding layers are concatenated and transferred to the successive layers (Zhu and Newsam (2017); Dhillon and Verma (2020)). This process allows to reuse the features, leverage the gradient flow, and prevent the vanishing gradient problem. In addition to that, it has transition layers that consist of batch normalization, ReLu activation function, 1×1 convolution to reduce dimensions, and 2×2 average pooling that controls the growth of the number of features and consequently reduces the computational complexity (Zhu and Newsam (2017); Dhillon and Verma (2020)). In this paper, we have implemented three different versions of DenseNet. These are DenseNet-121, DenseNet-169, and DenseNet-201 (see Huang et al. (2017)). These versions, respectively, contain 121, 169, and 201 layers (Huang et al. (2017)).

3.2.2. MobileNet

It is an alternative to CNNs, computationally more efficient than CNNs and tailored for deployment on devices with limited computational capacities, such as smartphones (Howard et al. (2017)). Unlike CNNs, which use regular convolution operations where a filter processes all the input channels (e.g., red, green, blue for a color image) together, MobileNet uses a depthwise separable convolution (DPSC) operation (Nelson and Gailly (1995)). DPSC comprises into two steps—Depthwise convolution (DC) and Pointwise convolution (PC) (Nelson and Gailly (1995)). Initially DC, filters are applied to each input channel individually and reduce the number of parameters, which makes the computational process faster compared to the regular convolution operation. Then, PC combines the outputs from DC and transforms them into a single output (Nelson and Gailly (1995)). In this paper, we employed MobileNetV1 (Howard et al. (2017)) and MobileNetV2 (Sandler et al. (2018)) for feature extraction from images.

3.2.3. Inception

This DL architecture uses an inception module, which allows Inception to learn global and local patterns and leverage its ability to learn features comprehensively (Kolla et al. (2023)). More specifically, the inception module applies convolutional filters of different sizes (1×1, 3×3, and 5×5) in parallel, which allows this architecture to learn features simultaneously at multiple scales (Alzubaidi et al. (2021)). In addition, 1×1 convolution reduces the dimensionality before applying the larger filters, which leverages the architecture's computational efficiency (Alzubaidi et al. (2021)). It was first introduced in 2014 by Google (Szegedy et al. (2015)). Inception-V3 architecture was introduced in 2016. It has 159 layers, and for feature extraction, it uses a standard convolutional block (Kassani et al. (2019)).

3.2.4. Inception-Residual Network

It ensembles the architecture of Inception and Residual networks, incorporating residual connections within the inception block structure (Szegedy et al. (2017)). Residual connections resolve the vanishing gradient issue and enhance the efficiency of this architecture in classification tasks, particularly for image data (Szegedy et al. (2017)). In this paper, we have used Inception-ResNet-V2 architecture for feature extraction.

3.2.5. Xception

The architecture of Xception is an improvement over Inception (Chollet (2017)). One can consider this a bundle of depthwise separable convolution (DSC) with residual connection (RC), where DSC reduces the computational cost as well as improves memory usage by separating the learning of channel-wise and space-wise features, and RC solves the vanishing gradient issue (Alzubaidi et al. (2021)). It has 36 convolutional layers structured into 14 modules (Kassani et al. (2019)).

medRxiv preprint doi: https://doi.org/10.1101/2024.11.26.24318006; this version posted November 29, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted medRxiv a license to display the preprint in perpetuity.
It is made available under a CC-BY-ND 4.0 International license .

Table 3
Summary of the Augmentation Parameters

Augmentation Parameter	Description	Value
Rotation range	Degree range for random rotation	20 degree
Width shift range	Range of random horizontal shift	20% of the image width
Height shift range	Range of random vertical shift	20% of the image height
Shear range	Transformation that slants the image shape	0.2 radians
Zoom range	Range of random zooming	20% of the image
Horizontal flip	Randomly flip images horizontally	True
Fill mode	Fill strategy for newly created pixels	Nearest

3.3. Machine Learning Classifiers

We have used the following feature set

$$\mathcal{A} = \{z_j^{\text{Train}} : j = 1, 2, \dots, N\},\ z_j^{\text{Train}} \in \mathbb{R}^d$$

to train several ML classifiers. In the following subsections, we provide an overview of how these classifiers generally function.

3.3.1. Support Vector Machine

A crucial step in training a support vector machine (SVM) is to find a hyperplane, given by:

$$w^T z + b = 0, \tag{8}$$

so that the margin between the nearest data points of each class and the hyperplane is maximized. Here, $w \in \mathbb{R}^d$ denotes the weight, and $b \in \mathbb{R}$ denotes the bias in (8). It is important to note that to train an SVM; the labels must be mapped either -1 and 1 or 0 and 1 (Stitson et al. (1996); Suthaharan and Suthaharan (2016b)). For linearly separable data, to obtain the final classifier $x \mapsto \text{sign}\left(w^T z + b\right)$, one has to solve the following optimization problem (Kecman (2005)):

$$\min_{w,b} \frac{1}{2}\|w\|_2^2 \tag{9}$$

$$\text{subject to, } \mathcal{Y}_j \left(w^T z_j + b\right) \geq 1, \tag{10}$$

where $j \in \{1, 2, \dots, N\}$. Note that $\text{sign}\left(w^T z + b\right) = 1$ if $w^T z + b > 0$, and 0 otherwise.

In this paper, to train SVM, we have mapped the labels associated with each feature into 0 and 1.

3.3.2. Random Forest

It performs classification task by ensembling predictions made by multiple decision trees based on majority voting strategy (Breiman (2001)). Suppose there are ℓ decision trees $T_1, T_2, \dots, T_\ell$ in the forest. These trees are also known as "base learner" (Cutler et al. (2012)). Initially, it trains each decision tree on a subset drawn from the training feature space using bootstrap sampling with replacement. Each of these trees make a prediction for a given input z, then final prediction is made as (Cutler et al. (2012)):

$$\hat{\mathcal{Y}} = \underset{\mathcal{Y} \in C}{\text{argmax}} \sum_{i=1}^{\ell} \mathbb{I}\left(\mathcal{Y} = \hat{T}_\ell(z)\right), \tag{11}$$

where $\mathbb{I}$ denotes the indicator function, $\hat{T}_\ell$ denotes the ℓ-th fitted tree at z, and C is the set of all possible values of $\mathcal{Y}$.

3.3.3. Logistic Regression

For any probability of success p expressed in terms of odds as (Kleinbaum et al. (2002); Nick and Campbell (2007); LaValley (2008)):

$$\frac{p}{1-p} = \exp\left(\beta_0 + \beta^t z\right), \tag{12}$$

the Logisitic regression (LR) is defined as follows:

$$\log\left(\frac{p}{1-p}\right) = \beta_0 + \beta^t z. \tag{13}$$

Here $\beta = \{\beta_1, \beta_2, \dots, \beta_n\}$ is the coefficient vector of z and β_0 is a constant.

3.3.4. Decision Tree

In general, a Decision tree (DT) is trained by deploying a continuously growing binary decision tree in training features z_j^{Train}, $j = 1, 2, \dots, N$. It starts from a root node and ends with the decision outputs known as the leaf node (De Ville (2013); Song and Ying (2015); Suthaharan and Suthaharan (2016a)).

3.3.5. K-Nearest Neighborhood

The K-Nearest Neighbor (K-NN) classifier predicts the label of a target point z^* based on the patterns that are nearest to z^*. To identify these patterns, the classifier computes the similarity between the target point z^* and the input features using distance metrics

medRxiv preprint doi: https://doi.org/10.1101/2024.11.26.24318006; this version posted November 29, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted medRxiv a license to display the preprint in perpetuity.
It is made available under a CC-BY-ND 4.0 International license .

such as the Euclidean distance or the Minkowski distance (Peterson (2009); Kramer (2013)). For binary classification, K-NN is defined as (Kramer (2013)):

$$\hat{\mathcal{Y}} = \begin{cases} 1, & \text{if } \sum_{i \in N_K(z^*)} \mathcal{Y}_i \geq 0, \\ -1, & \text{if } \sum_{i \in N_K(z^*)} \mathcal{Y}_i < 0, \end{cases} \tag{14}$$

where K denotes the neighborhood size, and $N_K(z^*)$ denotes the set of indices of the K nearest neighbor. In this paper, we have considered $K = 5$.

3.3.6. Naïve Bayes Classifier

It assumes that the features are conditionally independent given any target label (Szostak (2012)). In other words, it considers the contribution of each feature independently. For binary classification, it predicts a label as follows (Szostak (2012); Permission (2005)):

$$\hat{\mathcal{Y}} = \underset{k \in \{0,1\}}{\arg\max}\, \mathbb{P}(\mathcal{Y} = k) \prod_{j=1}^{d} \mathbb{P}\left(z_j | \mathcal{Y} = k\right). \tag{15}$$

In this paper, for each z_j, we have considered that the likelihood function $\mathbb{P}\left(z_j | \mathcal{Y} = k\right)$ is normally distributed with mean μ_{jk} and variance σ_i.

3.3.7. Adaptive Boosting

Adaptive boosting (AdaBoost) combines weak classifiers $T_1, T_2, \ldots, T_m$ and creates a strong classifier as follows:

$$\mathcal{Y} = \sum_{i=1}^{m} w_i T_i(z), \tag{16}$$

where m denotes the total number of classifiers, and w_i is the weight assigned to the weak classifier T_i (see Freund and Schapire (1997); Singh et al. (2023)).

3.3.8. Extreme Gradient Boosting (XGBoost)

It is an improvement over Gradient boosting (GB), which forms a model for prediction by ensembling multiple decision trees where each successive tree depends upon the previous tree and tries to fix errors made by the preceding ones (Chen (2015); Chen and Guestrin (2016)). One of the important features of XGBoost is it prevents overfitting by regularizing the loss function as follows:

$$\sum_{i=1}^{n} \mathcal{L}\left(\mathcal{Y}_i, \hat{\mathcal{Y}}_i\right) + \mathcal{R}, \tag{17}$$

where $\hat{\mathcal{Y}}_i$ denotes the predicted observations, $\mathcal{L}$ denotes the loss function, and $\mathcal{R}$ is the regularization term, given by:

$$\mathcal{R} = \gamma t + \frac{1}{2}\lambda \sum_{j=1}^{t} w_j^2. \tag{18}$$

Here λ denotes the regularized parameter to scale the penalty, γ the minimum loss needed to further partition the leaf node, w denotes the weight of the leafs, t is the number of leaves in the tree.

3.3.9. Light Gradient Boosting (LightGBM)

It was built at the top of the Gradient boosting framework for classification and other machine-learning tasks. It was developed by Microsoft (Ke et al. (2017)). Its architecture includes histogram based learning, which reduces memory usage and accelerates computational speed, leaf-wise tree growth, Gradient based one-sided sampling (GOSS), exclusive feature bundling, which groups categorical features with common values, parallel and GPU learning, and regularization to prevent overfitting such as L_1 or L_2 regularization (Ke et al. (2017)).

In this paper, we utilized packages and modules from the Scikit-learn library in Python to implement the classifiers (Pedregosa et al. (2011)).

3.4. Evaluation Metrics

In this section, we present evaluation metrics to evaluate hybrid DL models. Our goal is to classify Mpox and other conditions from image data. To formalize the evaluation process, we construct the following hypotheses:

H_0 : The observation belongs to the "Others" class

H_A : The observation belongs to the "Mpox" class

where H_0 denotes the null hypothesis and H_A is the alternative hypothesis. Based on the predictions made by a hybrid DL model, the possible outcomes of this hypothesis test are:

i) True positive (TP): observation is correctly predicted as Mpox. In other words, H_0 is rejected.

ii) True negative (TN): H_0 is accepted, that is observation is classified as others.

iii) False positive (FP): observation is incorrectly classified as Mpox when it is actually others, which means that the H_0 is rejected when it is true.

iv) False negative (FN): observation is incorrectly classified as others when it is actually Mpox. In other words, H_0 is accepted when it is false.

medRxiv preprint doi: https://doi.org/10.1101/2024.11.26.24318006; this version posted November 29, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted medRxiv a license to display the preprint in perpetuity.
It is made available under a CC-BY-ND 4.0 International license .

These outcomes can be represented in a matrix form, commonly referred to as the confusion matrix (CM), as follows:

$$CM = \begin{bmatrix} \text{TP} & \text{FN} \\ \text{FP} & \text{TN} \end{bmatrix}$$

We have computed accuracy, precision, recall, F1 score, weighted precision, weighted recall, weighted F1 score, and Matthews Correlation Coefficient (MCC) using these outcomes. In Table 4, we present the evaluation metrics in detail.

Table 4
Evaluation metrics

Metric	Formula
Accuracy	$\frac{TP+TN}{TP+TN+FP+FN}$
Precision	$\frac{TP}{TP+FP}$
Recall (Sensitivity)	$\frac{TP}{TP+FN}$
F1 Score	$2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision}+\text{Recall}}$
Weighted Precision	$\frac{\sum_{i=1}^{k} w_i \times \text{Precision}_i}{\sum_{i=1}^{k} w_i}$
Weighted Recall	$\frac{\sum_{i=1}^{k} w_i \times \text{Recall}_i}{\sum_{i=1}^{k} w_i}$
Weighted F1 Score	$\frac{\sum_{i=1}^{k} w_i \times \text{F1}_i}{\sum_{i=1}^{k} w_i}$
MCC score	$\frac{(TP \times TN)-(FP \times FN)}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}}$

In addition to these evaluation metrics, we have also computed the area under the receiver operating characteristic curve (ROC). ROC plots the sensitivity (true positive rate) against the false positive rate (FPR). FPR is given by:

$$\text{FPR} = \frac{\text{FP}}{\text{FP} + \text{TN}} \tag{19}$$

For n points on the ROC curve, area under the curve (AUC) can be computed using Trapezoidal rule approximation as follows:

$$\text{AUC} \approx \sum_{i=1}^{n-1} \left(\text{FPR}_{i+1} - \text{FPR}_i\right) \frac{\text{TPR}_{i+1} + \text{TPR}_i}{2}. \tag{20}$$

4. Results and Discussion

The objective of this research is to develop a hybrid DL framework that integrates pre-trained DL architectures with ML classifiers to predict Mpox from image data. Lack of image data and computational complexity are two stumbling blocks for Mpox prediction. This framework enables the construction of hybrid DL models capable of identifying Mpox using a small, publicly available dataset while optimizing memory usage and computational time. We have comprehensively analyzed this framework using several hybrid DL models (see Table 5, Table 6, and Table 7).

The hybrid models listed in Table 5 combine different DenseNet architectures (DenseNet-201, DenseNet-169, and DensNet-121) with ML classifiers. Notably, the hybrid model consists of DenseNet-201, and LR (D201LR) obtained an accuracy of 85.11%, outperforming other models that incorporated DenseNet-201 as a feature extraction tool. This model identified 18 true positive Mpox images out of 21 Mpox cases reported in the test dataset (see Figure 3a). On the other hand, DenseNet-201 combined with K-NN (D201KNN) achieved the highest precision (90%) among all DenseNet-201 incorporated hybrid models, highlighting its ability to minimize false positive cases.

The highest overall accuracy, 87.23%, was achieved by the model consisting of DenseNet-169 with LR (D169LR). While it outperforms other hybrid models, its precision is significantly small (82.61%) compared to D201KNN. It shows that D169LR had a slightly higher rate of predicting false positive cases. However, D169LR identified 19 true positive Mpox images (Figure 3b), which is slightly higher than D201KNN.

DenseNet-169 with Adaboost (D169Adaboost) is another model which attained the same accuracy as D169LR (87.23%). Other evaluation metrics of this model, such as weighted precision (87.39%), weighted recall (87.23%), F1-score (85%), and weighted F1-score (87.15%) are comparable to D169LR, although its precision is higher than D169LR, which indicates that its strength while minimizing false positive is essential.

DenseNet-169 with SVM (D169SVM) is another noteworthy hybrid model for its highest recall rate, 90.48%, the same as D169LR. Hybird Models, DenseNet-121 with LR (D121LR) and DenseNet-121 with LightGBM (D121LightGBM) both performed equally in terms of every metric for Mpox prediction (see Table 5).

However, D169LR outperformed other models in MCC score (Table 5). Note that, MCC score provides valuable information about the model performance, most importantly when the dataset is imbalanced. D169LR provided a score of 0.75, which highlights its strength in Mpox prediction. While D169Adaboost archived a very close sore (0.74) to D169LR, performance of other competing models are significantly low.

From the hybrid models enlisted in Table 5, we have selected three best performing models based on their MCC scores.

medRxiv preprint doi: https://doi.org/10.1101/2024.11.26.24318006; this version posted November 29, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted medRxiv a license to display the preprint in perpetuity.
It is made available under a CC-BY-ND 4.0 International license .

Table 5
Hybrid DL models combining DenseNet architectures and ML Classifiers

	Classifiers	Accuracy	Precision	Weighted Precision	Recall (Sensitivity)	Weighted Recall	F1-score	Weighted F1 score	MCC score
	SVM	82.98	78.26	83.37	85.71	82.98	81.82	83.03	0.66
	RF	80.85	83.33	81.11	71.43	80.85	76.92	80.64	0.61
	LR	**85.11**	**81.82**	**85.24**	**85.71**	**85.11**	**83.72**	**85.13**	**0.70**
	DT	51.06	46.15	52.23	57.14	51.06	51.06	51.06	0.03
DenseNet-201	K-NN	72.34	90.00	77.59	42.86	72.34	58.06	69.85	0.47
	NB	72.34	66.67	73.08	76.19	72.34	71.11	72.42	0.45
	XGBoost	74.47	76.47	74.74	61.90	74.47	68.42	74.04	0.48
	AdaBoost	78.72	73.91	79.12	80.95	78.72	77.27	78.78	0.58
	LightGBM	76.60	77.78	76.72	66.67	76.60	71.79	76.33	0.52
	SVM	85.11	79.17	85.88	90.48	85.11	84.44	85.15	0.71
	RF	76.60	75.00	76.54	71.43	76.60	73.17	76.53	0.52
	LR	**87.23**	**82.61**	**87.62**	**90.48**	**87.23**	**86.36**	**87.27**	**0.75**
	DT	53.19	48.00	54.14	57.14	53.19	52.17	53.28	0.07
DenseNet-169	K-NN	65.96	85.71	72.87	28.57	65.96	42.86	61.06	0.35
	NB	68.09	63.64	68.26	66.67	68.09	65.12	68.14	0.36
	XGBoost	76.60	72.73	76.75	76.19	76.60	74.42	76.64	0.53
	AdaBoost	87.23	89.47	87.39	80.95	87.23	85.00	87.15	0.74
	LightGBM	78.72	78.95	78.74	71.43	78.72	75.00	78.59	0.57
	SVM	80.85	77.27	80.99	80.95	80.85	79.07	80.89	0.61
	RF	78.72	76.19	78.72	76.19	78.72	76.19	78.72	0.57
	LR	**82.98**	**80.95**	**82.98**	**80.95**	**82.98**	**80.95**	**82.98**	**0.66**
	DT	63.83	58.33	64.55	66.67	63.83	62.22	63.93	0.28
DenseNet-121	K-NN	63.83	83.33	70.97	23.81	63.83	37.04	57.83	0.30
	NB	74.47	69.57	74.88	76.19	74.47	72.73	74.54	0.49
	XGBoost	74.47	68.00	75.64	80.95	74.47	73.91	74.51	0.50
	AdaBoost	74.47	69.57	74.88	76.19	74.47	72.73	74.54	0.49
	LightGBM	**82.98**	**80.95**	**82.98**	**80.95**	**82.98**	**80.95**	**82.98**	**0.66**

medRxiv preprint doi: https://doi.org/10.1101/2024.11.26.24318006; this version posted November 29, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted medRxiv a license to display the preprint in perpetuity.
It is made available under a CC-BY-ND 4.0 International license .

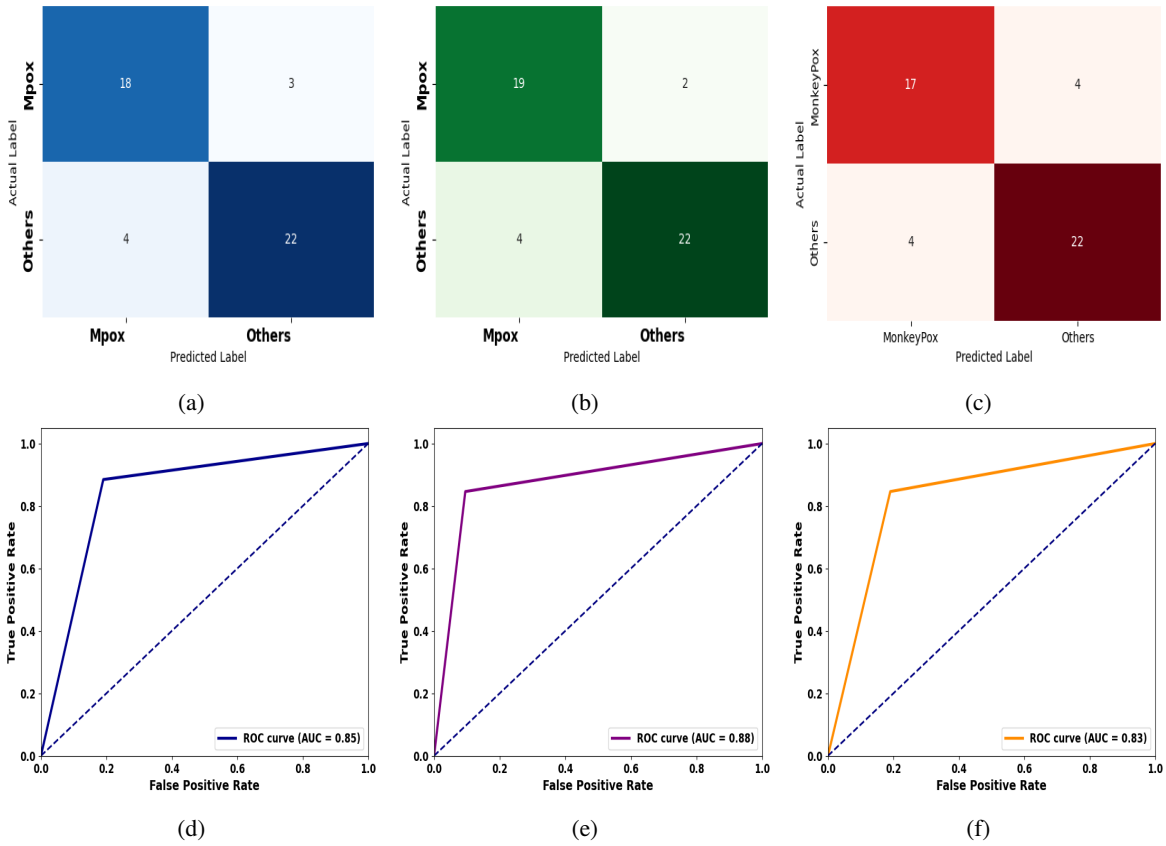


Figure 3: Confusion matrices and ROC curves for the selected hybrid models, incorporating DenseNet-201, DenseNet-169, and DenseNet-121 deep learning architectures as feature extraction tools from the image data. These models are selected based on their MCC score from each category listed in Table 5. Figure 3a, 3b and 3c are the confusion matrices for D201LR, D169LR, and D121LightGBM respectively, and 3d, 3e, and 3f illustrate their corresponding ROC curves.

In Figure 3, we incorporated Confusion matrices and Receiver Operating Characteristic (ROC) curves of these models. Since D121SVM and D121LightGBM performed equally in each evaluation metric, we only considered one of these two models in Figure 3. Confusion matrices allowed us to distinguish between positive and negative cases with accuracy. Along with these matrices, we investigated ROC curves for these models to assess model performance. Area Under Curve (AUC) of D161LR reflects its efficiency in Mpox identification.

In Table 6, we have constructed a few more hybrid models by incorporating InceptionV3, InceptionResNetV2, and Xception architecture as a feature extraction tool. Out of all the models developed based on InceptionV3, InceptionV3 with AdaBoost (IV3AdaBoost) obtained the highest accuracy (87.23%). Although IV3XGBoost (InceptionV3 with XGBoost) outperformed IV3AdaBoost in sensitivity (95.24%), its MCC score (0.72) and other metric scores are significantly lower than IV3AdaBoost. IV3AdaBoost and D169LR performed equally to classify Mpox and other images since both models scored equal values in every evaluation metric.

InceptionResNetV2 with RF (IRV2LR) performed better in every metric compared to other models, which are based on InceptionResNetV2 architecture. Even though IRV2LR and D201LR have equal accuracy (85.11%) and MCC score (0.70), their precision, weighted precision, recall, weighted recall, F1 score, and weighted F1 score are significantly different. However, the AUC of D201LR is slightly higher than IRV2LR.

Among hybrid models integrating Xception as a feature extraction tool, XcepSVM (Xception with SVM) and XecpLR (Xeception with LR) performed better than others. However, its accuracy (78.72%) is significantly low compared to D169LR and IRV2LR.

medRxiv preprint doi: https://doi.org/10.1101/2024.11.26.24318006; this version posted November 29, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted medRxiv a license to display the preprint in perpetuity.
It is made available under a CC-BY-ND 4.0 International license .

Table 6
Hybrid DL models combining InceptionV3, InceptionResNetV2, and Xception architectures with ML classifiers

	Classifiers	Accuracy	Precision	Weighted Precision	Recall (Sensitivity)	Weighted Recall	F1-score	Weighted F1 score	MCC score
	SVM	80.85	77.27	80.99	80.95	80.85	79.07	80.89	0.61
	RF	82.98	78.26	83.37	85.71	82.98	81.82	83.03	0.66
	LR	82.98	80.95	82.98	80.95	82.98	80.95	82.98	0.66
	DT	78.72	73.91	79.12	80.95	78.72	77.27	78.78	0.58
InceptionV3	K-NN	63.83	70.00	65.66	33.33	63.83	45.16	60.57	0.26
	NB	74.47	71.43	74.47	71.43	74.47	71.43	74.47	0.48
	XGBoost	85.11	76.92	87.05	95.24	85.11	85.11	85.11	0.72
	AdaBoost	**87.23**	**82.61**	**87.62**	**90.48**	**87.23**	**86.36**	**87.27**	**0.75**
	LightGBM	82.98	80.95	82.98	80.95	82.98	80.95	82.98	0.66
	SVM	78.72	82.35	79.21	66.67	78.72	73.68	78.36	0.57
	RF	**85.11**	**88.89**	**85.50**	**76.19**	**85.11**	**82.05**	**84.94**	**0.70**
	LR	78.72	82.35	79.21	66.67	78.72	73.68	78.36	0.57
	DT	68.08	66.67	67.94	57.14	68.09	61.54	67.73	0.35
InceptionResNetV2	K-NN	55.32	50.00	53.37	14.29	55.32	22.22	47.91	0.04
	NB	68.09	66.67	67.94	57.14	68.09	61.54	67.73	0.35
	XGBoost	78.72	73.91	79.12	80.95	78.72	77.27	78.78	0.58
	AdaBoost	76.60	77.78	76.72	66.67	76.60	71.79	76.33	0.52
	LightGBM	76.60	77.78	76.72	66.67	76.60	71.79	76.33	0.52
	SVM	**78.72**	**86.67**	**80.21**	**61.90**	**78.72**	**72.22**	**78.05**	**0.58**
	RF	70.21	73.33	70.80	52.38	70.21	61.11	69.27	0.39
	LR	**78.72**	**86.67**	**80.21**	**61.90**	**78.72**	**72.22**	**78.05**	**0.58**
	DT	65.96	63.16	65.76	57.14	65.96	60.00	65.74	0.31
Xception	K-NN	59.57	60.00	59.70	28.57	59.57	38.71	55.93	0.16
	NB	65.96	57.14	76.24	95.24	65.96	71.43	63.94	0.43
	XGBoost	70.21	73.33	70.80	52.38	70.21	61.11	69.27	0.39
	AdaBoost	76.60	81.25	77.35	61.90	76.60	70.27	76.04	0.53
	LightGBM	76.60	81.25	77.35	61.90	76.60	70.27	76.04	0.53

medRxiv preprint doi: https://doi.org/10.1101/2024.11.26.24318006; this version posted November 29, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted medRxiv a license to display the preprint in perpetuity.
It is made available under a CC-BY-ND 4.0 International license .

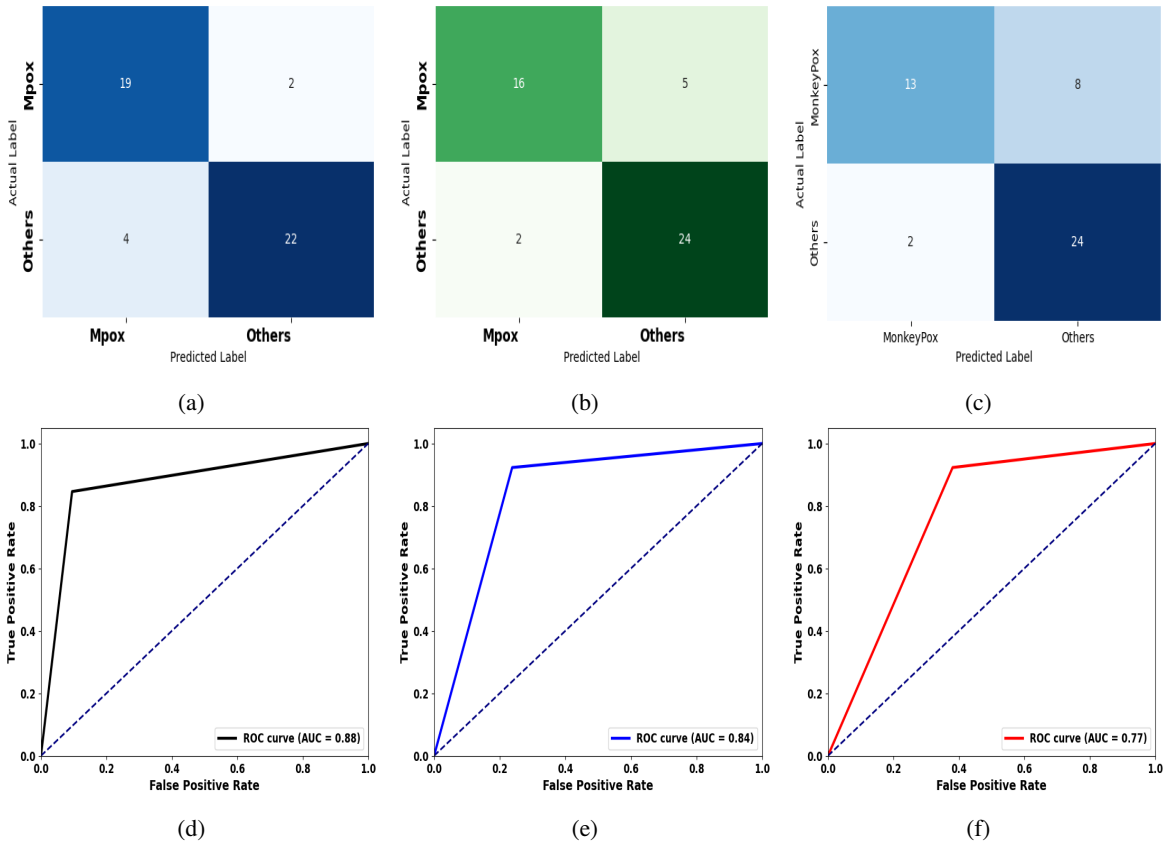


Figure 4: Performance evaluation of the hybrid models IV3AdaBoost (4a, 4d), IRV2LR (4b, 4e), and XcepSVM (4c, 4f) based on confusion matrices and ROC curves. These models are selected based on their MCC score from the list of models generated incorporating InceptionV3, InceptionResNetV2, and Xception as feature extraction tools (see Table 6).

XcepSVM and XecpLR identified 13 images as Mpox (see Figure 4c). Since XcepSVM and XecpLR performed equally, we incorporated a confusion matrix and ROC curve only for XcepSVM (Figure 4c).

In Table 7, we have more hybrid models that were developed by combining MobileNet architecture and ML classifiers. MobileNetV1 and MobileNetV2 were used to create these models. MV1LR (MobileNetV1 with LR) obtained 80.85% accuracy, which indicates its efficiency in Mpox prediction. In addition to that, its other evaluation metrics, such as precision (83.33%), are also significantly higher than other models that were developed based on MobileNetV1. Although its sensitivity or recall (71.43%) is lower than the model MV1LightGBM (MobileNetV1 with LightGBM), which is 76.19%, its MCC score of 0.61 outperformed other models. It identified 15 Mpox images and 23 other images (see Figure 5 (a)).

MV2LightGBM obtained accuracy 91.49%, which is highest among the models developed based on the MobileNetV2 architecture. Although its precision score (86.96%) is slightly lower than the precision score of MV2AdaBosst (MobileNetV2 with AdaBoost) (88.89%), it has outperformed other models in MCC score. Moreover, MV2LightGBM obtained the highest accuracy over all the hybrid models developed listed in Table-5, Table-6 and Table-7.

86.96% Precision score of MV2LightGBM indicates its ability to identify true positive cases of Mpox, which is higher than MV1LR. Furthermore, its weighted precision of 91.87% shows an overall scenario of how MV2LightGBM is suitable for binary classification, especially for an imbalanced dataset. In addition to that, its 91.87% recall and 95.24% weighted recall highlighted the robustness of this model in detecting TP positive cases out of all TP positive cases. More specifically, one can see that this model identified 20 Mpox images out of 21 (Figure 5b), which is the highest number of predictions over all other models.

medRxiv preprint doi: https://doi.org/10.1101/2024.11.26.24318006; this version posted November 29, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted medRxiv a license to display the preprint in perpetuity.
It is made available under a CC-BY-ND 4.0 International license .

Table 7
Hybrid DL models combining MobileNetV1 and MobileNetV2 architectures with ML classifiers

	Classifiers	Accuracy	Precision	Weighted Precision	Recall (Sensitivity)	Weighted Recall	F1-score	Weighted F1 score	MCC score
MobileNetV1	SVM	78.72	78.95	78.74	71.43	78.72	75.00	78.59	0.57
	RF	78.72	73.91	79.12	80.95	78.72	77.27	78.78	0.58
	LR	**80.85**	**83.33**	**81.11**	**71.43**	**80.85**	**76.92**	**80.64**	**0.61**
	DT	63.83	59.09	64.02	61.90	63.83	60.47	63.90	0.27
	K-NN	57.45	100	75.95	4.76	57.45	9.09	44.01	0.16
	NB	51.06	47.73	76.64	100	51.06	64.62	40.32	0.23
	XGBoost	78.72	76.19	78.72	76.19	78.72	76.19	78.72	0.57
	AdaBoost	78.72	78.95	78.74	71.43	78.72	75.00	78.59	0.57
	LightGBM	78.72	76.19	78.72	76.19	78.72	76.19	78.72	0.57
MobileNetV2	SVM	82.98	84.21	83.07	76.19	82.98	80.00	82.87	0.65
	RF	80.85	80.00	80.82	76.19	80.85	78.05	80.80	0.61
	LR	82.98	88.24	83.68	71.43	82.98	78.95	82.69	0.66
	DT	74.47	68.00	75.64	80.95	74.47	73.91	74.51	0.50
	K-NN	59.57	75.00	65.67	14.29	59.57	24.00	50.81	0.19
	NB	53.19	48.78	67.89	95.24	53.19	64.52	46.11	0.22
	XGBoost	85.11	85.00	85.10	80.95	85.11	82.93	85.07	0.70
	AdaBoost	85.11	88.89	85.50	76.19	85.11	82.05	84.94	0.70
	LightGBM	**91.49**	**86.96**	**91.87**	**95.24**	**91.49**	**90.91**	**91.51**	**0.83**

medRxiv preprint doi: https://doi.org/10.1101/2024.11.26.24318006; this version posted November 29, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted medRxiv a license to display the preprint in perpetuity.
It is made available under a CC-BY-ND 4.0 International license .

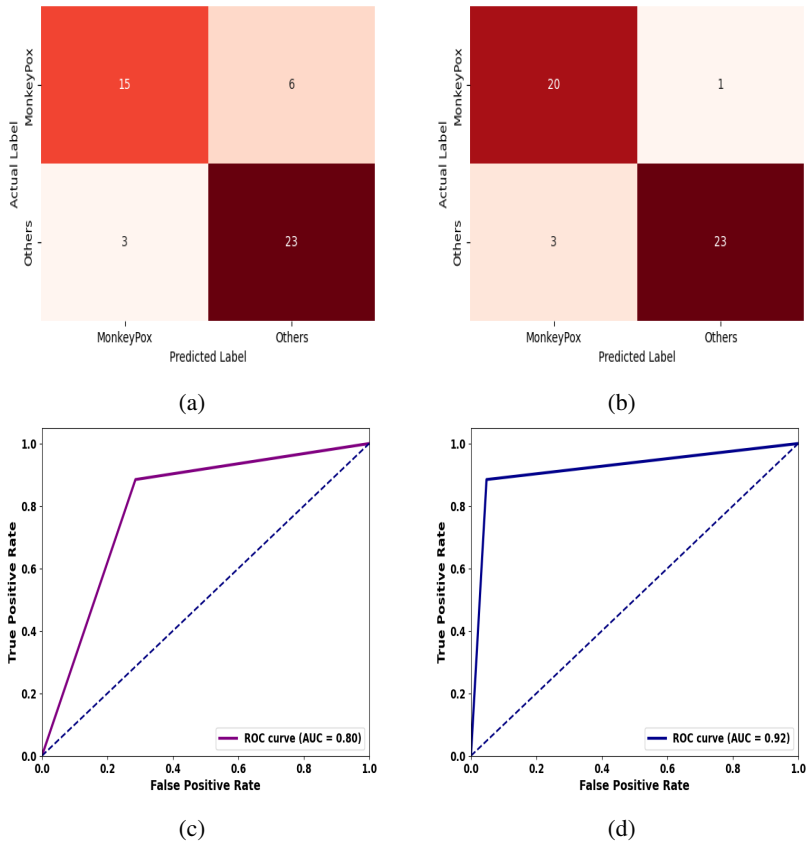


Figure 5: Performance evaluation of hybrid models constructed using MobileNetV1 and MobileNetV2 architectures. Figures 5a and 5c are the confusion matrix and ROC curve for the MV1LR model, respectively, while Figures 5b and 5d show the confusion matrix and ROC curve for the MV2LightGBM model.

Next, the 90.91% F1 score and the 91.51% weighted F1 score of MV2LightGBM highlighted the strength of this model in minimizing false positive and false negative cases. In other words, it can correctly identify 90.91% of the true positive instances while keeping the false positive and false negative cases low. Finally, the MCC score of this model (0.83) established it as an efficient model for the prediction of Mpox.

4.1. Comparison with Existing Methods

Through our analysis, MV2LightGBM emerged as the best-performing hybrid model, achieving the highest accuracy compared to all other evaluated hybrid models. This section compares this model with the different models previously published in the literature for Mpox detection. In this paper, we have used the dataset created by Ali et al. (2022). However, MV2LightGBM, which obtained 91.49% accuracy, outperformed the model proposed by Ali et al. (2022). Moreover, it has outperformed the model proposed by Sahin et al. (2022) by a slight margin. Note that Sahin et al. (2022) also used the same dataset created by Ali et al. (2022). Furthermore, MV2LightGBM outperformed Sahin et al. (2022) and Dwivedi et al. (2022), not only in terms of accuracy but also across other evaluation metrics (see Table 8).

5. Conclusion

Mpox is a zoonotic infectious skin disease that has raised concern due to its global outbreaks. The construction of an early detection tool for Mpox has emerged as a significant problem in preventing a possible future outbreak. In this paper, we developed a framework for the early detection of Mpox from biomedical image data. Our framework is constructed with a DL architecture and an ML classifier. In our framework, the DL architecture is accountable for feature extraction, and ML classifiers are used for classification. We built several hybrid models to perform a comprehensive analysis and evaluated these models

medRxiv preprint doi: https://doi.org/10.1101/2024.11.26.24318006; this version posted November 29, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted medRxiv a license to display the preprint in perpetuity.
It is made available under a CC-BY-ND 4.0 International license .

Table 8
Comparison of Our Best-Performing Model with Previously Published Results

	Accuracy	Precision	Recall	F1 score
MV2LightGBM	91.49	86.96	95.24	90.91
Ali et al. (2022)	82.96	87	83	84
Sahin et al. (2022)	91.11	90	90	90
Sitaula and Shahi (2022)	87.13	85.44	85.47	85.40
Dwivedi et al. (2022)	87	92	87	90

with standard evaluation metrics. Finally, we compared the results of our best-performing model with those published in the literature. Our results showed significant improvements over the results reported in the literature. In future work, we aim to investigate the efficiency of this framework for other infectious diseases.

Funding

The authors did not receive any funding for this research.

Competing Interests

The authors declare no competing interests.

References

Abro, A.A., Khan, A.A., Talpur, M.S.H., Kayijuka, I., Yaşar, E., 2021. Machine learning classifiers: a brief primer. University of Sindh Journal of Information and Communication Technology 5, 63–68.

Ahmed, I., Modu, G.U., Yusuf, A., Kumam, P., Yusuf, I., 2021. A mathematical model of coronavirus disease (covid-19) containing asymptomatic and symptomatic classes. Results in physics 21, 103776.

Ahsan, M.M., Alam, T.E., Haque, M.A., Ali, M.S., Rifat, R.H., Nafi, A.A.N., Hossain, M.M., Islam, M.K., 2024. Enhancing monkeypox diagnosis and explanation through modified transfer learning, vision transformers, and federated learning. Informatics in Medicine Unlocked 45, 101449.

Akilandasowmya, G., Nirmaladevi, G., Suganthi, S., Aishwariya, A., 2024. Skin cancer diagnosis: Leveraging deep hidden features and ensemble classifiers for early detection and classification. Biomedical Signal Processing and Control 88, 105306.

Akula, A., Pushkar, S., 2023. Mpox skin lension detection using deep learning approaches. Authorea Preprints .

Ali, M.S., Miah, M.S., Haque, J., Rahman, M.M., Islam, M.K., 2021. An enhanced technique of skin cancer classification using deep convolutional neural network with transfer learning models. Machine Learning with Applications 5, 100036.

Ali, S.N., Ahmed, M.T., Paul, J., Jahan, T., Sani, S., Noor, N., Hasan, T., 2022. Monkeypox skin lesion detection using deep learning models: A feasibility study. arXiv preprint arXiv:2207.03342 .

Alzubaidi, L., Zhang, J., Humaidi, A.J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., Santamaría, J., Fadhel, M.A., Al-Amidie, M., Farhan, L., 2021. Review of deep learning: concepts, cnn architectures, challenges, applications, future directions. Journal of big Data 8, 1–74.

Amin, J., Sharif, M., Yasmin, M., Fernandes, S.L., 2020. A distinctive approach in brain tumor detection and classification using mri. Pattern Recognition Letters 139, 118–127.

Anantharajan, S., Gunasekaran, S., Subramanian, T., Venkatesh, R., 2024. Mri brain tumor detection using deep learning and machine learning approaches. Measurement: Sensors 31, 101026.

Arumugam, M., Thiyagarajan, A., Adhi, L., Alagar, S., 2024. Crossover smell agent optimized multilayer perceptron for precise brain tumor classification on mri images. Expert Systems with Applications 238, 121453.

Batiha, I.M., Abubaker, A.A., Jebril, I.H., Al-Shaikh, S.B., Matarneh, K., Almuzini, M., 2023. A mathematical study on a fractional-order seir mpox model: analysis and vaccination influence. Algorithms 16, 418.

Betti, M.I., Farrell, L., Heffernan, J., 2023. A pair formation model with recovery: Application to mpox. Epidemics 44, 100693.

Bhatele, K.R., Jha, A., Tiwari, D., Bhatele, M., Sharma, S., Mithora, M.R., Singhal, S., 2024. Covid-19 detection: A systematic review of machine and deep learning-based approaches utilizing chest x-rays and ct scans. Cognitive Computation 16, 1889–1926.

Bhattacharya, S., Maddikunta, P.K.R., Pham, Q.V., Gadekallu, T.R., Chowdhary, C.L., Alazab, M., Piran, M.J., et al., 2021. Deep learning and medical image processing for coronavirus (covid-19) pandemic: A survey. Sustainable cities and society 65, 102589.

Bhattacharyya, D., Kim, T.h., 2011. Brain tumor detection using mri image analysis, in: Ubiquitous Computing and Multimedia Applications: Second International Conference, UCMA 2011, Daejeon, Korea, April 13-15, 2011. Proceedings, Part II 2, Springer. pp. 307–314.

Biau, G., Scornet, E., 2016. A random forest guided tour. Test 25, 197–227.

Breiman, L., 2001. Random forests. Machine learning 45, 5–32.

Breman, J.G., Steniowski, M., Zanotto, E., Gromyko, A., Arita, I., et al., 1980. Human monkeypox, 1970-79. Bulletin of the World Health Organization 58, 165.

Brown, K., Leggat, P.A., 2016. Human monkeypox: current state of knowledge and implications for the future. Tropical medicine and infectious disease 1, 8.

Centers for Disease Control and Prevention, 2024a. About mpox (monkeypox). URL: https://www.cdc.gov/poxvirus/mpox/about/index.html. accessed: 2024-09-14.

Centers for Disease Control and Prevention, 2024b. Nndss - table 968: Weekly cases of notifiable diseases - 2024. https://wonder.cdc.gov/nndss/static/2024/36/2024-36-table968.html. Accessed: 2024-09-04.

Chen, J., Li, Y., Guo, L., Zhou, X., Zhu, Y., He, Q., Han, H., Feng, Q., 2024. Machine learning techniques for ct imaging diagnosis of novel coronavirus pneumonia: a review. Neural Computing and

medRxiv preprint doi: https://doi.org/10.1101/2024.11.26.24318006; this version posted November 29, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted medRxiv a license to display the preprint in perpetuity.
It is made available under a CC-BY-ND 4.0 International license .

Applications 36, 181–199.

Chen, T., 2015. Xgboost: extreme gradient boosting. R package version 0.4-2 1.

Chen, T., Guestrin, C., 2016. Xgboost: A scalable tree boosting system, in: Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining, pp. 785–794.

Chen, X., Lei, C., Liu, D., Wang, G., Tang, H., Zha, Z.J., Li, H., 2021. E-commerce storytelling recommendation using attentional domain-transfer network and adversarial pre-training. IEEE Transactions on Multimedia 24, 506–518.

Chollet, F., 2017. Xception: Deep learning with depthwise separable convolutions, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1251–1258.

Composto, R.W., Tulbure, M.G., Tiwari, V., Gaines, M.D., Caineta, J., 2024. Quantifying urban flood extent using satellite imagery and machine learning. Natural Hazards , 1–25.

Cutler, A., Cutler, D.R., Stevens, J.R., 2012. Random forests. Ensemble machine learning: Methods and applications , 157–175.

De Ville, B., 2013. Decision trees. Wiley Interdisciplinary Reviews: Computational Statistics 5, 448–455.

Dhillon, A., Verma, G.K., 2020. Convolutional neural network: a review of models, methodologies and applications to object detection. Progress in Artificial Intelligence 9, 85–112.

Dwivedi, M., Tiwari, R.G., Ujjwal, N., 2022. Deep learning methods for early detection of monkeypox skin lesion, in: 2022 8th International Conference on Signal Processing and Communication (ICSC), IEEE. pp. 343–348.

Ebenezer, A.S., Kanmani, S.D., Sivakumar, M., Priya, S.J., 2022. Effect of image transformation on efficientnet model for covid-19 ct image classification. Materials Today: Proceedings 51, 2512–2519.

Eke, R., 1972. Monkey-pox in a four-year old girl: case report. .

Esteva, A., Kuprel, B., Novoa, R.A., Ko, J., Swetter, S.M., Blau, H.M., Thrun, S., 2017. Dermatologist-level classification of skin cancer with deep neural networks. nature 542, 115–118.

Etbaigha, F., R. Willms, A., Poljak, Z., 2018. An seir model of influenza a virus infection and reinfection within a farrow-to-finish swine farm. PLOS one 13, e0202493.

Foltz, E.A., Witkowski, A., Becker, A.L., Latour, E., Lim, J.Y., Hamilton, A., Ludzik, J., 2024. Artificial intelligence applied to non-invasive imaging modalities in identification of non-melanoma skin cancer: A systematic review. Cancers 16, 629.

Foster, S.O., Brink, E.W., Hutchins, D.L., Pifer, J.M., Lourie, B., Moser, C.R., Cummings, E.C., Kuteyi, O., Eke, R.E., Titus, J., et al., 1972. Human monkeypox. Bulletin of the World Health Organization 46, 569.

Franke, J., Menz, G., 2007. Multi-temporal wheat disease detection by multi-spectral remote sensing. Precision Agriculture 8, 161–172.

Fredricks, D.N., Relman, D.A., 1999. Application of polymerase chain reaction to the diagnosis of infectious diseases. Clinical infectious diseases , 475–486.

Freund, Y., Schapire, R.E., 1997. A decision-theoretic generalization of on-line learning and an application to boosting. Journal of computer and system sciences 55, 119–139.

Gessain, A., Nakoune, E., Yazdanpanah, Y., 2022. Monkeypox. New England Journal of Medicine 387, 1783–1793.

Goyal, L., Sharma, C.M., Singh, A., Singh, P.K., 2021. Leaf and spike wheat disease detection & classification using an improved deep convolutional architecture. Informatics in Medicine Unlocked 25, 100642.

Gupta, A., Bhagat, M., Jain, V., 2023. Blockchain-enabled healthcare monitoring system for early monkeypox detection. The Journal of Supercomputing 79, 15675–15699.

Hattaf, K., Dutta, H., 2020. Mathematical modelling and analysis of infectious diseases. volume 302. Springer.

Howard, A.G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., Adam, H., 2017. Mobilenets: Efficient convolutional neural networks for mobile vision applications. CoRR abs/1704.04861. URL: http://arxiv.org/abs/1704.04861, arXiv:1704.04861.

Huang, G., Liu, Z., Van Der Maaten, L., Weinberger, K.Q., 2017. Densely connected convolutional networks, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 4700–4708.

Huang, K., Hussain, A., Wang, Q.F., Zhang, R., 2019. Deep learning: fundamentals, theory and applications. volume 2. Springer.

Ianache, I., Skrzat-Klapaczynska, A., Jilich, D., Fleischhans, L., Gmizic, I., Ranin, J., Papadopoulos, A., Protopapas, K., Mulabdic, V., Lakatos, B., et al., 2024. Mpox across countries from central and eastern europe-2022 outbreak. Travel Medicine and Infectious Disease 59, 102719.

Ibrahim, A.U., Ozsoz, M., Serte, S., Al-Turjman, F., Yakoi, P.S., 2024. Pneumonia classification using deep learning from chest x-ray images during covid-19. Cognitive computation 16, 1589–1601.

Ismael, A.M., Şengür, A., 2021. Deep learning approaches for covid-19 detection based on chest x-ray images. Expert Systems with Applications 164, 114054.

Jiang, Y., Chen, H., Loew, M., Ko, H., 2020. Covid-19 ct image synthesis with a conditional generative adversarial network. IEEE Journal of Biomedical and Health Informatics 25, 441–452.

Kaftan, D., Kim, H.Y., Ko, C., Howard, J.S., Dalal, P., Yamamoto, N., Braithwaite, R.S., Bershteyn, A., 2024. Performance analysis of mathematical methods used to forecast the 2022 new york city mpox outbreak. Journal of Medical Virology 96, e29791.

Kassani, S.H., Kassani, P.H., Khazaeinezhad, R., Wesolowski, M.J., Schneider, K.A., Deters, R., 2019. Diabetic retinopathy classification using a modified xception architecture, in: 2019 IEEE international symposium on signal processing and information technology (ISSPIT), IEEE. pp. 1–6.

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., Liu, T.Y., 2017. Lightgbm: A highly efficient gradient boosting decision tree. Advances in neural information processing systems 30.

Kecman, V., 2005. Support vector machines–an introduction, in: Support vector machines: theory and applications. Springer, pp. 1–47.

Khan, S.U.R., Zhao, M., Asif, S., Chen, X., 2024. Hybrid-net: A fusion of densenet169 and advanced machine learning classifiers for enhanced brain tumor diagnosis. International Journal of Imaging Systems and Technology 34, e22975.

Kimeswenger, S., Tschandl, P., Noack, P., Hofmarcher, M., Rumetshofer, E., Kindermann, H., Silye, R., Hochreiter, S., Kaltenbrunner, M., Guenova, E., et al., 2021. Artificial neural networks and pathologists recognize basal cell carcinomas based on different histological patterns. Modern Pathology 34, 895–903.

Kleinbaum, D.G., Dietz, K., Gail, M., Klein, M., Klein, M., 2002. Logistic regression. Springer.

Kolla, B.S., Reddy, B.R., Sahithi, S.V., Madala, L.P., 2023. Comparative analysis of vgg19, resnet50, and googlenet inception models for bci .

Kramer, O., 2013. K-nearest neighbors. Dimensionality reduction with unsupervised nearest neighbors , 13–23.

Kussul, N., Shelestov, A., Skakun, S., 2008. Grid system for flood extent extraction from satellite images. Earth Science Informatics 1, 105–117.

medRxiv preprint doi: https://doi.org/10.1101/2024.11.26.24318006; this version posted November 29, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted medRxiv a license to display the preprint in perpetuity.
It is made available under a CC-BY-ND 4.0 International license .

Ladnyj, I., Ziegler, P., Kima, E., 1972. A human infection caused by monkeypox virus in basankusu territory, democratic republic of the congo. Bulletin of the World Health Organization 46, 593.

LaValley, M.P., 2008. Logistic regression. Circulation 117, 2395–2399.

Li, G., Li, N., 2019. Customs classification for cross-border e-commerce based on text-image adaptive convolutional neural network. Electronic Commerce Research 19, 779–800.

Li, J., 2011. Malaria model with stage-structured mosquitoes. Mathematical Biosciences & Engineering 8, 753–768.

Liang, X., Lin, L., Yang, W., Luo, P., Huang, J., Yan, S., 2016. Clothes co-parsing via joint image segmentation and labeling with application to clothing retrieval. IEEE Transactions on Multimedia 18, 1175–1186.

Liu, T., Yang, S., Luo, B., Fan, X., Zhuang, Y., Gao, G.F., Bi, Y., Teng, Y., 2023. Anticipating the transmissibility of the 2022 mpox outbreak. Journal of Medical Virology 95, e28683.

Lu, C., Mandal, M., 2015. Automated analysis and diagnosis of skin melanoma on whole slide histopathological images. Pattern Recognition 48, 2738–2750.

Lu, J., Hu, J., Zhao, G., Mei, F., Zhang, C., 2017. An in-field automatic wheat disease diagnosis system. Computers and electronics in agriculture 142, 369–379.

Mahata, A., Paul, S., Mukherjee, S., Das, M., Roy, B., 2022. Dynamics of caputo fractional order SEIRV epidemic model with optimal control and stability analysis. International Journal of Applied and Computational Mathematics 8, 28.

McCollum, A.M., Damon, I.K., 2014. Human monkeypox. Clinical infectious diseases 58, 260–267.

Meena, G., Mohbey, K.K., Kumar, S., 2024. Monkeypox recognition and prediction from visuals using deep transfer learning-based neural networks. Multimedia Tools and Applications , 1–25.

Molla, J., Sekkak, I., Ortiz, A.M., Moyles, I., Nasri, B., 2023. Mathematical modeling of mpox: A scoping review. One Health 16, 100540.

Moore, M.J., Rathish, B., Zahra, F., 2021. Mpox (monkeypox) .

Musa, A.B., 2013. Comparative study on classification performance between support vector machine and logistic regression. International Journal of Machine Learning and Cybernetics 4, 13–24.

Narudin, F.A., Feizollah, A., Anuar, N.B., Gani, A., 2016. Evaluation of machine learning classifiers for mobile malware detection. Soft Computing 20, 343–357.

Nayak, T., Chadaga, K., Sampathila, N., Mayrose, H., Muralidhar Bairy, G., Prabhu, S., Katta, S.S., Umakanth, S., 2023. Detection of monkeypox from skin lesion images using deep learning networks and explainable artificial intelligence. Applied Mathematics in Science and Engineering 31, 2225698.

Nelson, M., Gailly, J.L., 1995. The data compression book 2nd edition. M & T Books, New York, NY .

Nick, T.G., Campbell, K.M., 2007. Logistic regression. Topics in biostatistics , 273–301.

Organization, W.H., et al., 2022. Monkeypox https://www. who. int/news-room/fact-sheets/detail/monkeypox. Accessed 10th Jun .

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E., 2011. Scikit-learn: Machine learning in Python. Journal of Machine Learning Research 12, 2825–2830.

Pereira, F., Mitchell, T., Botvinick, M., 2009. Machine learning classifiers and fmri: a tutorial overview. Neuroimage 45, S199–S209.

Permission, S., 2005. Generative and discriminative classifiers: naive bayes and logistic regression.

Petersen, E., Kantele, A., Koopmans, M., Asogun, D., Yinka-Ogunleye, A., Ihekweazu, C., Zumla, A., 2019. Human monkeypox: epidemiologic and clinical characteristics, diagnosis, and prevention. Infectious Disease Clinics 33, 1027–1043.

Peterson, L.E., 2009. K-nearest neighbor. Scholarpedia 4, 1883.

Rashidi, H.H., Tran, N.K., Betts, E.V., Howell, L.P., Green, R., 2019. Artificial intelligence and machine learning in pathology: the present landscape of supervised methods. Academic pathology 6, 2374289519873088.

Ritchie, H., Roser, M., in Data, O.W., 2024. Monkeypox explorer - our world in data. https://ourworldindata.org/explorers/monkeypox?country=%7EOWID_WRL&Metric=Confirmed+cases&Frequency=Cumulative&Relative+to+population=false. Accessed: 2024-09-04.

Sahin, V.H., Oztel, I., Yolcu Oztel, G., 2022. Human monkeypox classification from skin lesion images with deep pre-trained network using mobile application. Journal of Medical Systems 46, 79.

Sailunaz, K., Özyer, T., Rokne, J., Alhajj, R., 2023. A survey of machine learning-based methods for covid-19 medical image analysis. Medical & Biological Engineering & Computing 61, 1257–1297.

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L.C., 2018. Mobilenetv2: Inverted residuals and linear bottlenecks, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 4510–4520.

Sharma, S., Guleria, K., 2024. A systematic literature review on deep learning approaches for pneumonia detection using chest X-ray images. Multimedia Tools and Applications 83, 24101–24151.

Singh, D., Khandelwal, A., Bhandari, P., Barve, S., Chikmurge, D., 2023. Predicting lung cancer using xgboost and other ensemble learning models, in: 2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT), IEEE. pp. 1–6.

Sitaula, C., Shahi, T.B., 2022. Monkeypox virus detection using pre-trained deep learning-based approaches. Journal of Medical Systems 46, 78.

Song, Y.Y., Ying, L., 2015. Decision tree methods: applications for classification and prediction. Shanghai archives of psychiatry 27, 130.

Sorayaie Azar, A., Naemi, A., Babaei Rikan, S., Bagherzadeh Mohasefi, J., Pirnejad, H., Wiil, U.K., 2023. Monkeypox detection using deep neural networks. BMC Infectious Diseases 23, 438.

Stitson, M., Weston, J., Gammerman, A., Vovk, V., Vapnik, V., 1996. Theory of support vector machines. University of London 117, 188–191.

Strzelecki, M., Kociołek, M., Strąkowska, M., Kozłowski, M., Grzybowski, A., Szczypiński, P.M., 2024. Artificial intelligence in the detection of skin cancer: state of the art. Clinics in Dermatology .

Suthaharan, S., Suthaharan, S., 2016a. Decision tree learning. Machine Learning Models and Algorithms for Big Data Classification: Thinking with Examples for Effective Learning , 237–269.

Suthaharan, S., Suthaharan, S., 2016b. Support vector machine. Machine learning models and algorithms for big data classification: thinking with examples for effective learning , 207–235.

Szegedy, C., Ioffe, S., Vanhoucke, V., Alemi, A., 2017. Inception-v4, inception-resnet and the impact of residual connections on learning, in: Proceedings of the AAAI conference on artificial intelligence.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A., 2015. Going deeper with convolutions, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1–9.

medRxiv preprint doi: https://doi.org/10.1101/2024.11.26.24318006; this version posted November 29, 2024. The copyright holder for this preprint **(which was not certified by peer review)** is the author/funder, who has granted medRxiv a license to display the preprint in perpetuity.
It is made available under a CC-BY-ND 4.0 International license .

Szostak, R., 2012. The basic concepts classification, in: Categories, Contexts and Relations in Knowledge Organization, Ergon-Verlag. pp. 24–30.

Uysal, F., 2023. Detection of monkeypox disease from human skin images with a hybrid deep learning model. Diagnostics 13, 1772.

Vanama, V., Rao, Y., Bhatt, C., 2021. Change detection based flood mapping using multi-temporal earth observation satellite images: 2018 flood event of kerala, india. European Journal of Remote Sensing 54, 42–58.

Vaughan, A., Aarons, E., Astbury, J., Brooks, T., Chand, M., Flegg, P., Hardman, A., Harper, N., Jarvis, R., Mawdsley, S., et al., 2020. Human-to-human transmission of monkeypox virus, united kingdom, october 2018. Emerging infectious diseases 26, 782.

Verma, S., Kumar, P., Singh, J.P., 2024. A unified lightweight cnn-based model for disease detection and identification in corn, rice, and wheat. IETE Journal of Research 70, 2481–2492.

Webb, G.I., Keogh, E., Miikkulainen, R., 2010. Naïve bayes. Encyclopedia of machine learning 15, 713–714.

Yang, S., Rothman, R.E., 2004. Pcr-based diagnostics for infectious diseases: uses, limitations, and future applications in acute-care settings. The Lancet infectious diseases 4, 337–348.

Yolcu Oztel, G., 2024. Vision transformer and cnn-based skin lesion analysis: classification of monkeypox. Multimedia Tools and Applications 83, 71909–71923.

Zhang, L., Huang, J., Yan, W., Zhao, Y., Wang, D., Chen, B., 2024. Global prediction for mpox epidemic. Environmental Research 243, 117748.

Zhu, Y., Newsam, S., 2017. Densenet for dense flow, in: 2017 IEEE international conference on image processing (ICIP), IEEE. pp. 790–794.