Available online at www.ijournalse.org

Emerging Science Journal

(ISSN: 2610-9182)

Vol. 8, No. 5, October, 2024

An Explainable Deep Learning Approach for Classifying Monkeypox Disease by Leveraging Skin Lesion Image Data

Andino Maseleno [1], Miftachul Huda [2], Chotirat Ann Ratanamahatana [1*]

[1] *Department of Computer Engineering, Faculty of Engineering, Chulalongkorn University, Bangkok, Thailand.*

[2] *Faculty of Human Sciences, Universiti Pendidikan Sultan Idris, Tanjung Malim, Malaysia.*

Abstract

According to the World Health Organization's (WHO) external situation report on the multi-country outbreak of Monkeypox in 2023, from 11 countries in Southeast Asia Regions, Thailand recorded the highest reported cases, totaling 461. The ongoing Monkeypox outbreak has raised significant public health concerns due to its rapid spread across several nations. Early detection and diagnosis are imperative for effectively treating and controlling Monkeypox. Given this context, this study aimed to determine the most efficient model for detecting Monkeypox by employing interpretable deep learning techniques. This study utilizes deep learning techniques to diagnose Monkeypox based on images of skin lesions. We evaluate based on four models—convolutional neural network (CNN), gated recurrent unit (GRU), long short-term memory (LSTM), and bidirectional long short term memory (BiLSTM)—using a publicly available dataset. Additionally, we incorporate Local Interpretable Model-Agnostic Explanations (LIME) and techniques for explainable AI, facilitating visual interpretation of model predictions for healthcare practitioners. The CNN model's performance and LSTM model's performance have an accuracy of 100%, while the GRU model's performance and BiLSTM model's performance have an accuracy of 99.88% and 99.45%. Our findings demonstrate the effectiveness of deep learning models, including the suggested CNN model leveraging the pre-trained MobileNetV2 and LSTM. These models can play a pivotal role in combating the Monkeypox virus.

Keywords:

Monkeypox; Skin Lesion Analysis;

Convolutional Neural Network (CNN);

Gated Recurrent Unit (GRU);

Long Short Term Memory (LSTM);

Bidirectional Long Short Term Memory (BiLSTM); Deep Learning;

Explainable Artificial Intelligence.

Article History:

Received:	03	June	2024
Revised:	16	September	2024
Accepted:	21	September	2024
Published:	01	October	2024

1- Introduction

While Covid-19 was still occurring in 2022, the world was again shocked by the emergence of the Monkeypox virus in May 2022 [1]. The reported cases of Monkeypox to the World Health Organization (WHO) now exceed the cumulative total from all previous years. On May 5, three years after the declaration of COVID-19 as a pandemic, the WHO announced the conclusion of the global Public Health Emergency (PHE) for COVID-19 [2]. In 1958, Monkeypox was initially detected in the rural rainforest regions of West Africa, which are among the world's most impoverished and underprivileged areas [3]. The initial human infection of Monkeypox in 1970 was identified in the Democratic Republic of the Congo (DRC) [4, 5]. Subsequently, the next Monkeypox outbreak was documented in 2003 in the DRC, followed by occurrences in 2005 in South Sudan and 2017 in Nigeria [6, 7]. Monkeypox outbreaks in regions beyond Africa in 2003 were documented in the Midwest states of the United States, followed by occurrences in the United Kingdom, Israel, and Singapore [6, 8]. Despite a lull in Monkeypox infections, a solitary case emerged in an individual who had traveled from Nigeria to the UK on May 7, 2022 [9, 10]. Subsequently, on July 23, 2022, the World Health Organization (WHO) announced the intensifying global Monkeypox outbreak as a Public Health Emergency of International Concern (PHEIC) due to the rapid surge in cases and the severity of the situation

[*] **CONTACT**: chotirat.r@chula.ac.th

DOI: http://dx.doi.org/10.28991/ESJ-2024-08-05-013

[1, 11]. Between January 1, 2022 to September 30, 2023, WHO recorded a total of 91,123 laboratory-confirmed cases of Monkeypox, along with 157 fatalities, across 115 countries [12, 13]. In September 2023, the most heavily impacted areas, ranked by case count, were in some regions, such as the Western Pacific, Europe, South-East Asia, America, and Africa. A single case was also documented in the Eastern Mediterranean Region [12, 13].

Thailand, the researcher's residence, which is included in the Southeast Asia Region, experiences an increase in Monkeypox every month, the most common case compared to other Southeast Asia Region countries [12, 13]. A significant increase in cases in recent months has been reported in Thailand, with 48 new cases in June, 80 in July, 145 in August, and 461 in September 2023. So far, only one Monkeypox-related death has been recorded among the 461 reported cases in an immune compromised patient (case fatality ratio 0.30%). While the outbreak was initially centered in Bangkok, it has expanded, with cases reported in 28 of the 76 national provinces. Most cases (95%) do not have a recent travel history, suggesting local virus acquisition. Most of the reported cases involve adults and young males, particularly those who are men engaging in same-sex activity [12, 13].

Monkeypox is a zoonotic illness caused by members of the Orthopoxvirus genus within the Poxviridae family [14]. This genus encompasses viruses like variola (responsible for smallpox), vaccinia (utilized in smallpox vaccines), and cowpox [15]. Monkeypox transmission can occur through contact with an infected animal, person, or contaminated objects [16]. The virus can additionally traverse the placenta from a pregnant woman to the fetus. The Monkeypox virus from animals to humans can be transmitted via bites or scratches from infected animals, during the handling or processing of game animals, or through products derived from infected animals [17]. The virus can disseminate through direct contact with an infected individual's bodily fluids or wounds or with objects contaminated by these fluids or wounds, like clothing or bedding. It can be transferred from an infected person to another healthy person through direct interaction with infected wounds, scabs, or bodily fluids [18]. Moreover, respiratory droplets can propagate the illness during prolonged proximity to an infected person [19].

Efficient and prompt prevention and diagnosis of this disease are crucial to curbing its global spread. The conventional method is employed for diagnosing this infectious ailment, wherein medical professionals detect Monkeypox disease through fluid swabs from skin rashes. Confirmation of Monkeypox infection can be attained through diagnostic testing, such as amplifying viral genetic material by polymerase chain reaction from samples of skin vesicular fluid [20]. However, manual interpretation of gene sequence data demands expertise and consumes time. This approach has several drawbacks, including the need for medical expertise, high costs, slow processing, and often unsatisfactory results [21]. Deep learning technologies could aid in preventing and detecting this infectious disease.

Modern machine learning approaches, such as deep learning, have advanced significantly in recent years. This progress was made feasible by the availability of large datasets, enhanced computational capabilities, and broader access to cutting-edge technologies. Consequently, artificial intelligence and machine learning have evolved from theoretical concepts confined to research labs to practical and widely applicable tools across various commercial sectors [22]. The healthcare sector has experienced significant expansion in utilizing machine learning methods [23]. Deep learning has garnered attention in health informatics, presenting advantages in extracting features and classifying data [24]. Deep learning models often incorporate numerous hidden neurons and layers, a departure from conventional neural network structures. This decision is influenced by the volume of raw data available while learning, allowing for using more neurons [25]. Deep learning approaches are based on representation learning, a process that develops nonlinear components layer per layer to achieve ever more profound levels of depiction. Every layer refines the depiction from one form to another, eventually leading to a broader representation, allowing for the automated development of a feature set [26, 27]. The automatic generation of feature sets without human interaction presents notable benefits in health informatics. Clinical image processing is a sector in which deep learning has been successfully used [28].

Without laboratory PCR-based diagnostics, clinical diagnoses of Monkeypox infection frequently rely on professional evaluation of distinctive skin lesions. The WHO declared the increasing worldwide Monkeypox epidemic as a public health emergency of international concern in 2022, and several researchers have investigated the utilization of deep learning techniques to streamline the automated detection and classification of Monkeypox-associated skin lesions from medical imaging data. The potential for Monkeypox to trigger the next pandemic underscores the urgent need for efficient resource allocation. Deep learning offers valuable contributions in various aspects, notably in disease diagnosis. Table 1 compares prior studies employing deep learning predictions for diagnosing the Monkeypox virus. Convolutional neural networks (CNNs) are extensively used in clinical image processing due to their proficiency in picture analysis and capacity to harness Graphics Processing Units (GPUs) [29-33]. Recently, there has been significant growth in the application of Deep Learning in therapeutic settings, demonstrating impressive advancements in performance. A range of Deep Learning and Artificial Intelligence models, including bidirectional long short-term memory (BiLSTM), SVM, K-NN, DT, Artificial Neural Network (ANN), and long short-term memory (LSTM), have been utilized for the classification of Monkeypox images [29-37].

A significant challenge in deploying deep learning solutions in medical contexts is the inherent "black-box" character of these predictions. This opacity means that medical professionals may need to fully comprehend the reasoning behind specific machine predictions [38]. The deep learning models in prior research were essentially opaque, offering no clear explanation for their predictions in a format understandable to humans [39]. As a result, clinicians needed more trust in this technology, as transparency and interpretability are crucial for its acceptance and utilization in clinical environments. For a medical diagnostic system to earn the trust of clinicians, officers, and patients, it must be transparent, understandable, and capable of explaining its decisions. Ideally, it should elucidate all stakeholders' decision-making processes [40]. The proposed approach suggests an explainable deep learning-based diagnostic system designed to efficiently and promptly detect the Monkeypox virus. To instill trust within the medical community, we propose combining understandable artificial intelligence methods, for instance, Local Interpretable Model-Agnostic Explanations (LIME) [41] and Gradient-weighted Class Activation Mapping (Grad-CAM) [42]. The remainder of this study is as follows: Section 2 delves into the available literature surrounding Monkeypox diagnosis. Section 3 describes the transfer learning mechanism used. The results derived from the deep learning classifiers are deliberated in Section 4. Section 5 summarizes the findings and discusses the following directions. This study's favorable outcomes indicate this approach's superiority over current methodologies. The dataset used in this study comes from the Kaggle web repository [43].

2- Related Works

Table 1 compares prior studies employing deep learning predictions for diagnosing the Monkeypox virus. Abdelhamid et al. [29] devised an image categorization algorithm named "AI-Biruni-Earth-Radius" by leveraging the GoogLeNet deep neural network for feature extraction; they achieved a peak accuracy of 98.8% in identifying Monkeypox within a multiclass dataset. Akin et al. [30] used Explainable Artificial Intelligence (XAI) and CNN to categorize Monkeypox skin lesion photos. Twelve deep-learning models were used to divide 572 skin lesion photos into two categories, and the MobileNetV2 model achieved the greatest accuracy, at 98.25%. Alakus & Baykara [34] employed sophisticated machine learning methods to distinguish Monkeypox from blisters by DNA sequencing. Their classification process comprised three stages, and the classifiers attained the maximum precision of 96.08%. They suggested that DNA sequences could be a diagnostic tool for distinguishing the Monkeypox virus from other comparable illnesses such as smallpox and measles. Khafaga et al. [36] utilized a deep CNN to group Monkeypox images sourced from Kaggle. The dataset comprised 293 normal, 279 Monkeypox, 107 Chickenpox, and 91 Measles. The model achieved a maximum precision of 98.83%. Eid et al. [35] introduced a method centered on an LSTM deep network, incorporating several optimization techniques. The outcomes, analyzed with various optimization methods using statistical techniques, achieved a maximum accuracy of 97%. In a separate study, Manohar & Das [37] utilized ANN along with optimization and K-Fold cross-validation strategies on a dataset of skin images for detecting Monkeypox disease, achieving an accuracy of 98%.

Additionally, the findings were compared with previously utilized LSTM and Gated Recurrent Unit (GRU) models. Sahin et al. [31] devised a mobile app to identify Monkeypox using video footage of blisters collected and posted to Android cellphones. They built the program in Java, with a handheld gadget capturing photos and transmitting them to a CNN model. The CNN-based model, which was trained and evaluated using Matlab software linked with TensorFlow and TensorFlow Lite, obtained a maximum reliability of 91.11%. Sitaula & Shahi [32] applied deep learning techniques for Monkeypox virus diagnosis. They trained and tested thirteen diverse models on the dataset, creating a CNN-based prediction for classifying skin lesions into eight disease categories. To enhance the approach, they compared their solution against pre-trained VGG-16 models and further optimized the ensemble, which their models yielded an average accuracy of 87.13%. Nayak et al. [33] utilized various deep learning models, including GoogLeNet, Places365-GoogleNet, SqueezeNet, AlexNet, and ResNet-18 for detecting the Monkeypox virus and from that, achieved the highest accuracy of 99.49% with their approach. The objectives of this investigation are defined as follows:

- Leveraging deep learning architectures like CNN, GRU, LSTM, and BiLSTM to achieve precise detection of the Monkeypox virus.
- Assessing the efficacy of the prediction models based on essential metrics like accuracy, precision, recall, and F1-score, and doing a comparison study with previous studies.
- Integrating LIME and Grad-CAM approaches to improve model readability allows for more in-depth knowledge of the reasons impacting their decisions. It establishes trust in the medical community about their usage.
- Further exploration of the practical applications of these models for real-time and accurate diagnosis.

Table 1. Comparison of prior studies employing deep learning techniques for diagnosing the Monkeypox virus

Author	Dataset	Models Used	Best Accuracy
Abdelhamid et al. (2022) [29]	Kaggle	CNN	98.80%
Akin et al. (2022) [30]	Kaggle	CNN	98.25%
Alakus & Baykara (2022) [34]	DNA seqences of Monkeypox and human papilloma virus	BiLSTM	96.08%
Khafaga et al. (2022) [36]	Kaggle	CNN, SVM, K-NN, DT	98%
Eid et al. (2022) [35]	Kaggle	LSTM, BiLSTM	97%
Manohar & Das (2022) [37]	Monkeypox skin lesion images from UCI	ANN, CNN	98%
Sahin et al. (2022) [31]	Kaggle	CNN	91.11%
Sitaula & Shahi (2022) [32]	Kaggle	CNN	87.13%
Nayak et al. (2023) [33]	Kaggle	CNN	99.49%

3- Research Methodology

The proposed method comprises five primary phases: data gathering, data preprocessing, model training, and model assessment. Initially, images of patients with Monkeypox are collected, with data augmentation techniques employed due to the constrained data availability aimed at generating supplementary images. During the data preprocessing phase, the gathered images are subjected to resizing, standardization, and data augmentation. This phase is crucial as it aims to enhance the model's performance. Hence, four prevalent models (CNN, GRU, LSTM, and BiLSTM) were chosen and compared to improve the model's accuracy in detecting the Monkeypox virus. During this model development step, preprocessed photos train the selected models. Throughout the training stage, the model gets photos and adjusts its parameters to improve performance. The last phase evaluates the model's performance using measures like accuracy, precision, recall, and the F1 score. The model with the highest efficiency is picked as the final model. As a result, the recommended technique analyzes patients' photos using deep learning approaches. Figure 1 shows the proposed method.

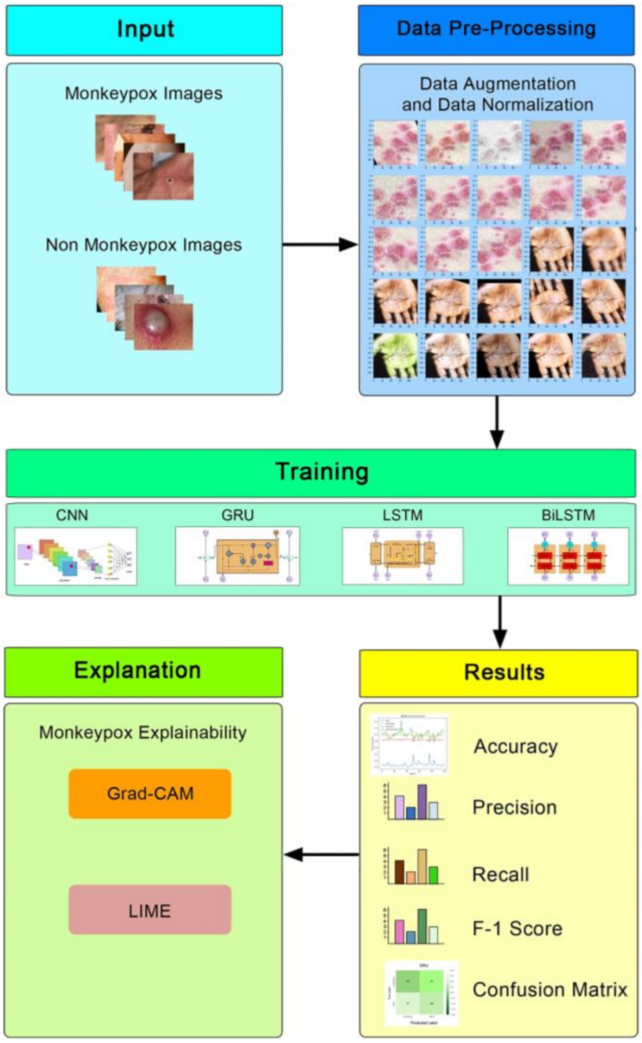


Figure 1. Proposed techniques

3-1- Data Collection

The rising global prevalence of Monkeypox infection has drawn interest, prompting the investigation of early detection strategies for this infectious ailment. A pivotal component of these initiatives involves harnessing the capabilities of machine learning methods to detect and differentiate Monkeypox from other comparable diseases precisely. We gathered data and assembled datasets as a preliminary measure to embark on this endeavor.

The dataset employed to train and validate our proposed model involves a binary classification task, distinguishing between Monkeypox and non-Monkeypox classes. This dataset was curated using images from the Internet and is publicly accessible on Kaggle, a community platform for data scientists and machine learning enthusiasts. Figure 2 and Figure 3 display samples from the dataset, comprising RGB images with dimensions of 224×224 pixels. The dataset consists of 9900 images, with 5064 classified as monkeypox and 4836 as non-monkeypox. The dataset's size is approximately 513.25 MB. Monkeypox disease classification was conducted using CNN, GRU, LSTM, and BiLSTM models, implemented in the PyTorch framework within the Python programming language, utilizing the Google Colab integrated development environment (IDE) for experimentation.

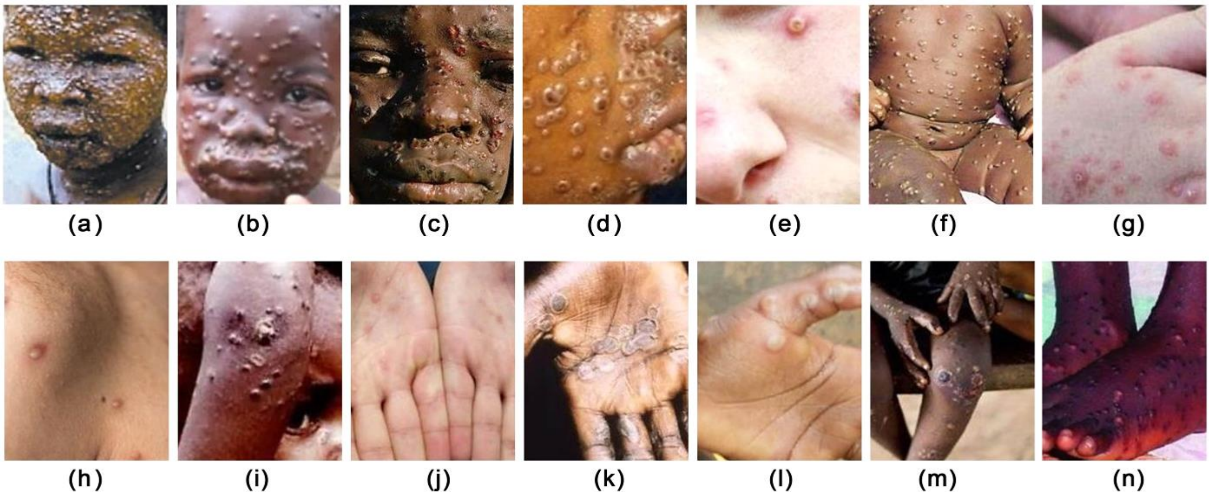


Figure 2. Images of skin lesions on monkeypox sufferers

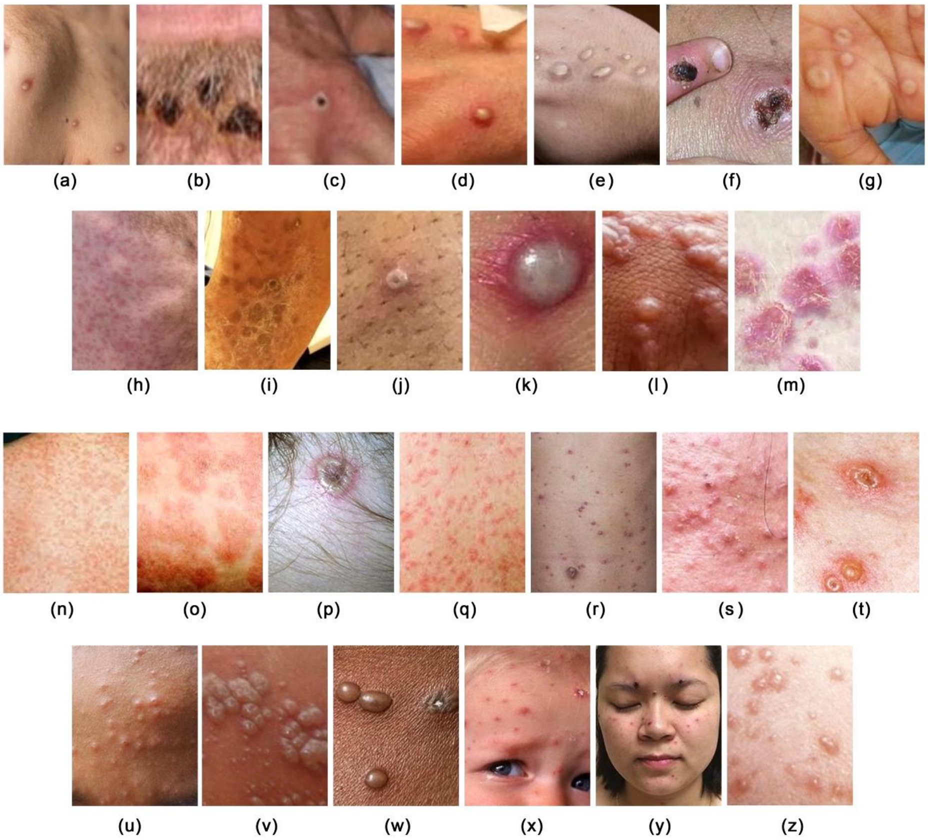


Figure 3. Images (a) – (m): class monkeypox, images (n) – (z): class non-monkeypox or others

3-2- Data Preprocessing

In this study, the preprocessing phase is critical in improving the quality and consistency of the data for image analysis. This involved resizing the images, standardizing their properties, and augmenting the dataset to ensure comprehensive coverage.

3-2-1-Image Resizing and Image Normalization

The preprocessing methodology was applied in prior experiments, such as resizing images, a technique used by scientists to transform input photos into precise dimensions suited for deep learning. While this process typically occurs before feeding images into the models, in some experiments, resizing is integrated into the network through a fully convolutional layer to mimic real-time scenarios to ensure that these models are not overfitting. In our study, all images were uniformly resized to 224×224 pixels to standardize dimensions, facilitating easier data processing by the model. In addition, normalization was used to reduce the influence of brightness and exposure fluctuations by scaling picture pixel amounts from 0 to 255 to a normalized range of zero to one.

3-2-2-Data Augmentation

The proposed augmentation procedures were utilized to diversify a data set and improve the capacity of the model to generalize [44]. The model encounters a broader spectrum of variations through these techniques, enabling it to discern the intrinsic features more effectively within the images. The data augmentation techniques employed include the following: (1) random cropping, which involves selecting a random center spot and cropping the picture to produce numerous variants; (2) rotating an image by certain angles, such as "45 degrees" done many times to produce various orientations; (3) adjusting color by adding or subtracting numbers to the red, green, and blue channels to induce color distortions (RGB); (4) To produce mirrored versions, flip the photos horizontally or vertically; (5) Changing the luminosity of pictures to make them either lighter or darker; and (6) transforming the pixels of the image by rearranging a set number of pixels to produce a variety of variants. These information augmentation strategies improved the dataset's variety and strengthened the model's generalization capacity, leading to a more precise and dependable classification of Monkeypox images. As a reference, augmented images are depicted in Figure 4.

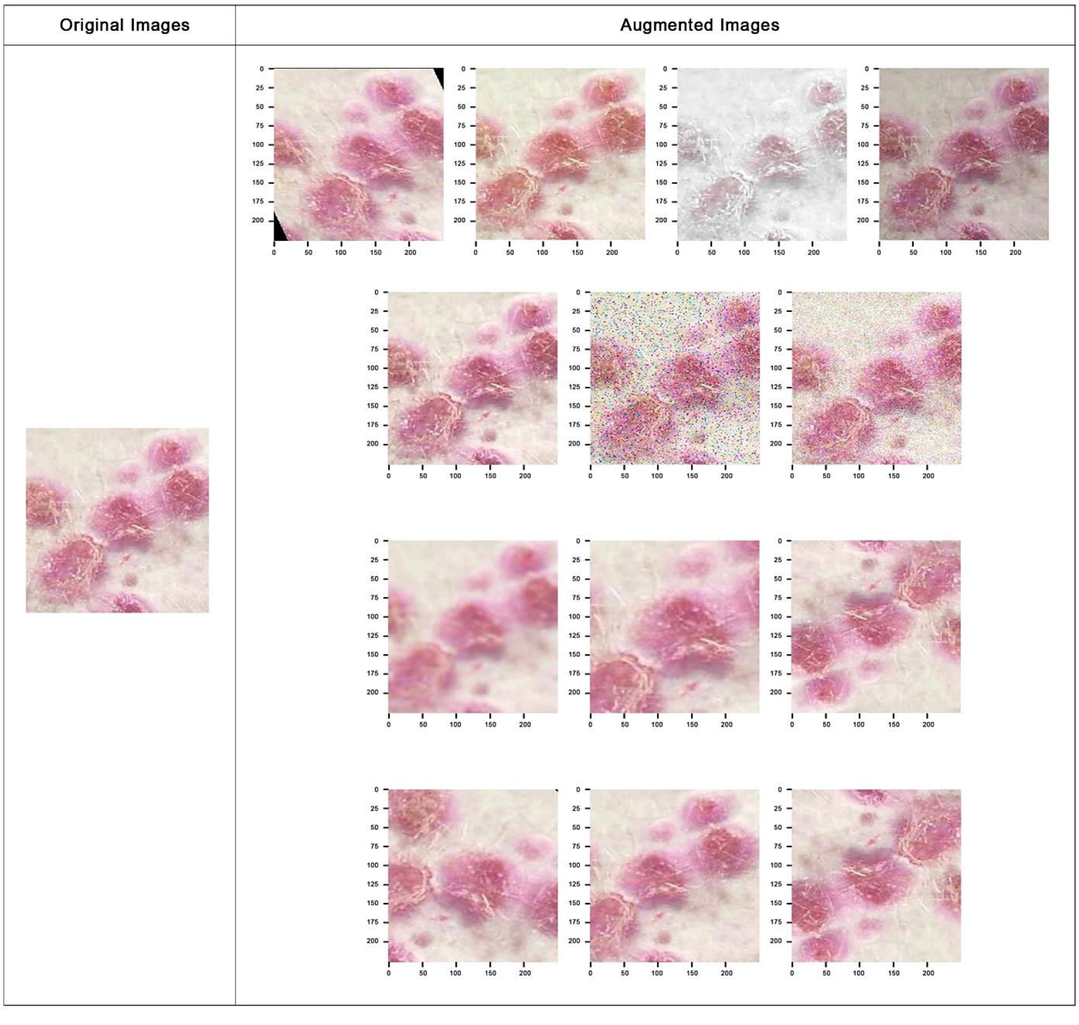


Figure 4. Augmented images

3-3- Data Preprocessing

3-3-1-Convolutional Neural Networks (CNN)

Convolutional neural networks use convolution operations instead of ordinary matrix multiplication in at least one layer. Figure 5 depicts the CNN design, which includes a completely connected layer of neurons, with every neuron in this layer coupled to each neuron in the layer directly underneath it [45]. The down-sampling technique in each pooling layer sub-region reduces individual neurons' dimensionality in the current layer by segmenting the neurons from the preceding layer into non-overlapping rectangular arrays. The two most popular pooling methods, maximum pooling, and average pooling extract the maximum or average value from each subarea. CNNs have an edge over multi-layer perceptrons (MLPs) in time series forecasting due to their ability to handle multivariate inputs and outputs.

Additionally, CNNs can learn complex functional relationships without explicitly relying on lagged observations. Thus, the CNN model can glean the most pertinent representation for the prediction problem from a diverse array of inputs [46]. CNN encodes information using convolution layers rather than tightly coupled processing units (neurons) in hidden layers, as with classic neural networks. The CNN model consists of clustering, convolution, and linked layers, serving as its primary components. Depending on the specific objective, these layers can be dynamically adjusted in terms of number or type. Through convolutional layers, the model utilizes multiple convolution kernels to grasp feature representations from inputs. Given their hierarchical structure, CNNs excel in noisy sequences, progressively filtering out noise in each subsequent layer while retaining essential patterns [47]. The choice to use the convolution layer instead of a completely connected layer is primarily driven by the latter's tendency to require a substantial number of parameters, necessitating significant computational resources. In contrast, the outputs of a convolutional layer undergo processing through a non-linear activation function before being passed to the subsequent layer. In constructing a convolutional layer h for a 1-D signal input, a series of small filters (of size L×1) indexed k = 1,..., Nk are utilized, as depicted in Equation 1.

$$h_i^k = f\left(\sum_{l=1}^{L} w_l^k \ X_{i+1} + \ b^k\right) \tag{1}$$

A frequently chosen activation function for $f(\cdot)$ is the rectified linear unit (ReLU). These layers can be integrated with other architectures to tackle more intricate tasks, such as GRU [47] or LSTM [48].

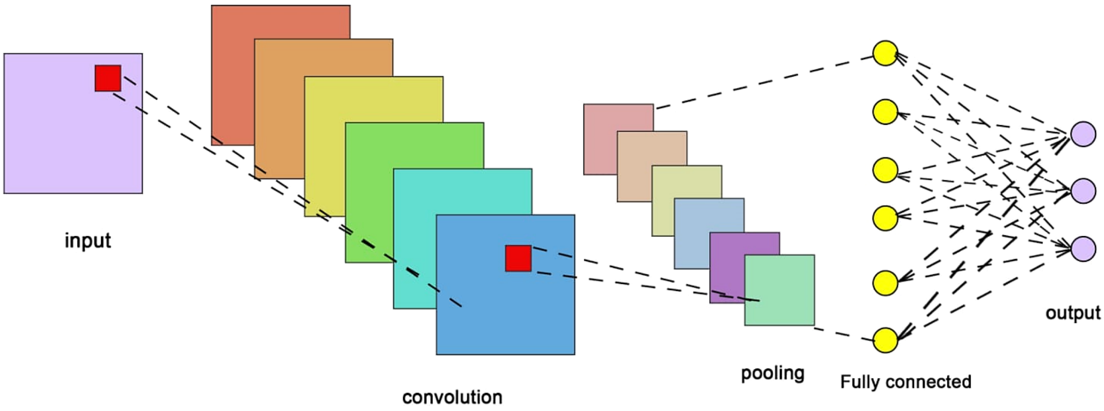


Figure 5. The basic architecture of convolutional neural networks (CNNs) [45]

The pooling layer was placed between two successive convolutional layers, reducing the picture's overall size. Pooling layers are classified into two types: max-pooling, which selects the most significant value, and average-pooling, which determines the average value from all neurons inside the preceding layer's clusters [49]. In the entirely networked layer, each neuron from a single layer is coupled to every single neuron in the following stratum. The outcome of the previous layer is used as the input for the initial ultimately linked layer. Before being fed into this layer, the final layer's output is changed from a matrix to a vector. Finally, this vector is delivered to the ultimately linked layer.

3-3-2-Gated Recurrent Unit (GRU)

The Gated Recurrent Unit (GRU) is a potent adaptation of the traditional Recurrent Neural Network (RNN) and shares similarities with LSTM, employing a sophisticated filtering system to handle short-term memory difficulties [50]. Figure 6 illustrates the fundamental structure of the GRU [51]. Within the GRU, internal gates control and modulate the information flow, aiding in determining which information is crucial for retention or deletion within the GRU cell [52]. Consequently, essential information is propagated to facilitate accurate predictions [28, 53]. Forget gate and input gates are combined as well to create an update gate z_t as shown in equation (2). The update gate manages the equilibrium between maintaining prior memory and absorbing fresh information. Here, x_t denotes the current input vector, while h_{t-1} represents the value derived from the preceding adjacent layer [54]. The parameter w_z refers to the trainable weight matrix associated with the update gate.

$$z_t = \sigma \left(w_z \, . [h_{t-1}, x_t]\right) \tag{2}$$

Additionally, in GRU, the current input is merged with the previous memory through the reset gate, denoted as r_t. This gate determines how the formula integrates the prior state with the new result, as seen in equation (3).

$$r_t = \sigma \left(w_r \, . [h_{t-1}, x_t]\right) \tag{3}$$

Tanh represents a tangential hyperbolic function with a result range of (-1,1). Additionally, h_t represents the computed value for the current cell, as depicted in equations (4) and (5). Moreover, the basic architecture of GRU can be seen in Figure 6.

$$h_t = tanh \left(r_r * \ [h_{t-1}, x_t]\right) \tag{4}$$

$$h_t = (1 - z_t) * h_{t-1} + z_t * h_t \tag{5}$$

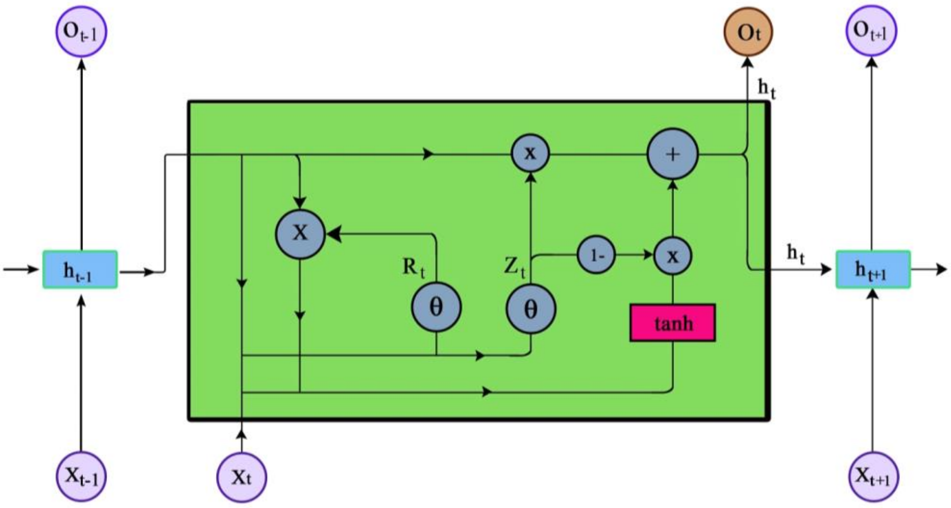


Figure 6. The basic architecture of gated recurrent unit (GRU) [51]

3-3-3- Long Short Term Memory (LSTM)

LSTM networks represent a distinct category of continuous neural networks designed to capture persistent dependencies in data [55]. Compared to simple recurrent neural networks, which may struggle to learn information from distant positions due to increasing gaps between predictions and relevant data, LSTM networks overcome this limitation, enhancing performance [56]. In contrast to the standard single-loop architecture, the LSTM network features a distinctive three-"gate" configuration comprising a forgetting gate, an input gate, and an output gate [57, 58]. LSTM networks have shown substantial success across various applications and are extensively employed in image analysis tasks [59-62]. Most existing recurrent neural networks utilize the LSTM architecture [63, 64], and the fundamental structure of the LSTM neural network is depicted in Figure 7 [65]. The number σ denotes the function sigmoid, which generates outcomes that vary between 0 to 1; tanh refers to the hyperbolic tangent function, yielding outputs between -1 and 1; h_{t-1} represents the previous cell's output, while X_t stands for the current cell's input.

During the early stages of the LSTM neural net, it decides either to keep or discard data within cell state. The equation 6 shows the computing formula for the forgetting gate.

$$f_t = \sigma \left(W_f \, . [\, h_{t-1}, X_t] + b_f\right) \tag{6}$$

here, σ denotes the function of sigmoid activation, f_t represents the forgetting gate, h_{t-1} signifies the output at time $t-1$, and X_t indicates the input vector at time t. W_f and b_f are the weight and bias vectors for the forget gate, respectively. A number of f_t nearing 0 implies the prior data has been discarded, while a value near 1 doesn't necessarily denote retention of the previous data. In the second phase of the LSTM neural network, the process involves deciding which new data to retain in the cell state. This involves two stages: Initially, the sigmoid layer decides which information requires updating, while the tanh layer produces a vector, denoted as $\tilde{C}_t$, serving as an alternative candidate value for updating, which is then incorporated into the cell state. After combining the two types of data, the resulting model generates novel data for updating the cell state. The computation for calculating the input gate is delineated by the following equations (Equations 7 and 8):

$$i_t = \sigma (W_i \,.\, [h_{t-1}, X_t] + b_i) \tag{7}$$

$$\tilde{C}_t = \tanh(W_C \,.\, [h_{t-1}, X_t] + b_C) \tag{8}$$

With these equations, the term σ denotes a function of sigmoid-shaped activation, i_t signifies the input gate, W_i and b_i signify the weight and bias vectors of the input gate, while W_C and b_C represent the updated weights and biases, respectively.

During the third stage of the LSTM neural network, the prior cell state C_{t-1} undergoes an update process determined by f_t and i_t. By multiplying C_t and C_{t-1} with f_t redundant information is eliminated. The resulting new cell state C_t is derived from the updated past state C_{t-1}, as described in Equation 9:

$$C_t = f_t . C_{t-1} + i_t . \tilde{C}_t \tag{9}$$

In the final stage of the LSTM neural network, we activate a layer with sigmoid shape to determine which fraction of the cell state will be output. The cell state is then processed using the function tanh to generate a value inside the range of -1 to 1, which is followed by multiplying by the output from the sigmoid gate. This process ensures that only the designated part of the output is produced. The calculation is outlined in Equations 10 and 11:

$$O_t = \sigma (W_O . [h_{t-1}, X_t] + b_O \tag{10}$$

$$h_t = O_t \tanh(C_t) \tag{11}$$

where h_t signifies the updated output value, O_t denotes the output gate, and W_O for the weight vector and b_O for bias vector of the output gate, correspondingly.

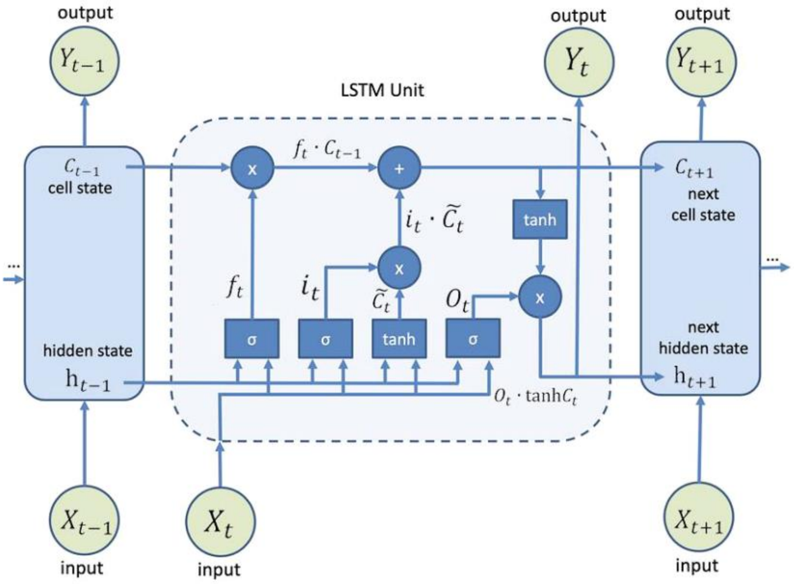


Figure 7. The basic architecture of long short term memory (LSTM) [65]

3-3-4- Bidirectional Long Short Term Memory (BiLSTM)

The BiLSTM network consists of LSTM units that handle data inputs in forward and reverse directions, enabling the model's algorithm to gather context data from both prior and future viewpoints. This enables BiLSTM to grasp long-term dependencies in sequences while avoiding redundancy in context information [66]. In BiLSTM, two LSTM layers are linked to the output layer. This configuration, treating two LSTM layers as a single layer, enhances model's ability to learn long-term dependencies, thereby improving its overall performance [67]. Previous research has demonstrated the superiority of bidirectional networks over standard ones across various domains, including estimation tasks [68-70]. The unfolded structure of a BiLSTM layer, comprising both forward and backward LSTM layers, is depicted in Figure 8 [71].

The upstream LSTM layer creates the outcome sequence $\vec{h}$ in a conventional way, whereas the reverse LSTM layer generates the outcome sequence $\overleftarrow{h}$ utilizing reversed data at times $t - 1$ to $t - n$. The σ function is used to aggregate the output sequences, resulting in the vector y_t [72]. A BiLSTM layer's final output is a vector, $Y_t = [y_{t-n,\ldots,}y_{t-1}]$ where the final component, y_{t-1}, is the estimate for the next repetition, similar to the LSTM layer.

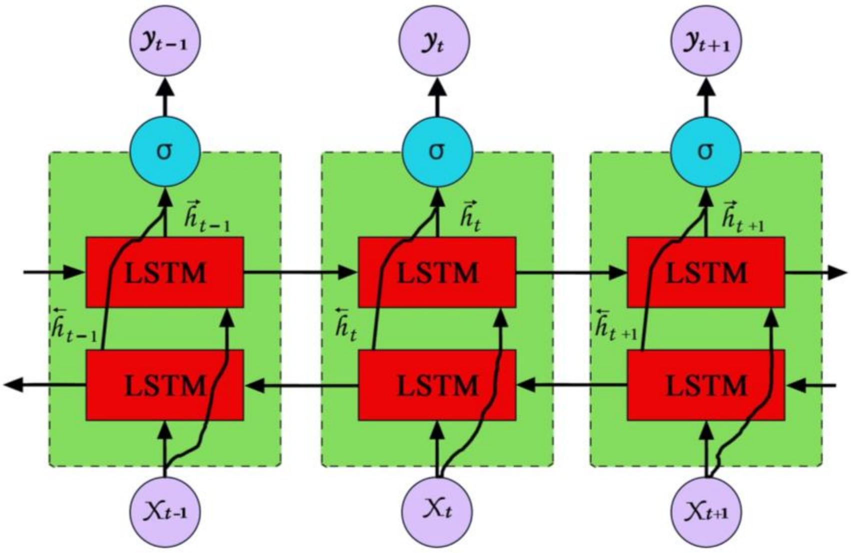


Figure 8. The basic architecture of bidirectional long short term memory (BiLSTM) [71]

3-4- Evaluation Metrics

In binary classification, evaluation metrics are represented by a 2×2 matrix, which includes numbers for true positives, true negatives, false positives, and false negatives [73]. True positive cases refer to instances where the model correctly identifies samples belonging to the Monkeypox class. True negative cases indicate accurate identification of samples that do not belong to the Monkeypox class. False positive and false negative outcomes represent incorrect predictions. False positive findings arise once non-Monkeypox samples are mistakenly recognized, while false negative results occur when Monkeypox cases are wrongly forecasted. Designs demonstrate well in terms of reducing false positives and false negatives.

To thoroughly assess the efficacy of our suggested approach for Monkeypox identification and classification, we used a range of conventional assessment indicators, including accuracy, precision, recall, F1-score, and Receiver Operating Characteristics (ROC) curve. Additionally, we computed extended evaluation metrics such as True Positive Rate, True Negative Rate, False Positive Rate, and False Negative Rate to provide a comprehensive evaluation.

1) Accuracy

Accuracy is the fraction of accurately predicted samples among the entire data. It represents several correctly predicted samples, including Monkeypox and non-Monkeypox (or other) instances, about the overall amount of data. This metric is determined using the Equation 12.

$$Accuracy\ (h) = \frac{1}{|X|} \sum_{x \in X}[h(x) = y] \tag{12}$$

2) Precision

Precision measures a ratio of correctly classified harmful programs relative to the number of detected harmful applications. This metric focuses on both true positive and false positive outcomes. Precision is highest when the number of false positives is low. The calculation is based on Equation 13.

$$Precision\ (h) = \frac{\sum_{j=1}^{l} t_{Pj}}{\sum_{j=1}^{l}(t_{Pj} + f_{Pj})} \tag{13}$$

where, t_p corresponds to the count of true-positive determinations, and f_p represents the count of false-positive identifications.

3) Recall

Recall, also known as True Positive Rate (TPR), represents the ratio of precisely predicted values to the total number of records for each class. It assesses how well the model identifies positive instances from the entire dataset and the Equation 14 is utilized to compute recall.

$$Recall\ (h) = \frac{\sum_{j=1}^{l} t_{Pj}}{\sum_{j=1}^{l}(t_{Pj} + f_{nj})} \tag{14}$$

where, t_p signifies the count of true-positive determinations made by the algorithm model, while f_n indicates the count of false-negative identifications by the model.

4) F1-score

The F1 score indicates the harmonic mean of recall and precision. Its calculation is represented by Equation 15:

$$F1 - score = \frac{2 \times True\ Positives}{2 \times True\ Positives + False\ Positives + False\ Negatives} \tag{15}$$

5) Confusion Matrix

The matrix of confusion is a comprehensive tool for assessing the efficacy of a classification model, whether in binary or multi-class contexts. It gives useful information on measures like as accuracy, precision, recall, and the AUC-ROC curve.

4- Results and Discussion

Figure 9 displays an extensive confusion matrix illustrating the multi-classification task performed by our proposed models, encompassing CNN, GRU, LSTM, and BiLSTM. The classifier's evaluation primarily relies on this confusion matrix. In a pair of categories situation, the confusion matrix is a 2×2 square matrix where the columns denote the classifier's predictions, and the rows indicate the actual class labels. This matrix enables a comparison between predicted and true labels, facilitating the computation of measures such as accuracy, precision, recall, and F1-score. The confusion matrix comprises four components: true positives (TPs), true negatives (TNs), false positives (FPs), and false negatives (FNs). TPs correspond to accurately predicted positive cases, with both actual and predicted classes being Monkeypox. TNs signify accurately predicted negative instances, where both actual and predicted classes are not Monkeypox or others. FPs indicate cases where the true class is not Monkeypox or others, but the predicted class is Monkeypox. FNs denote cases where the true class is Monkeypox, but the predicted class is not Monkeypox or others. The final model for detecting and classifying Monkeypox was chosen based on its superior performance. The accuracy, precision, recall, and F1-score measures were used to assess the performance of the four models.

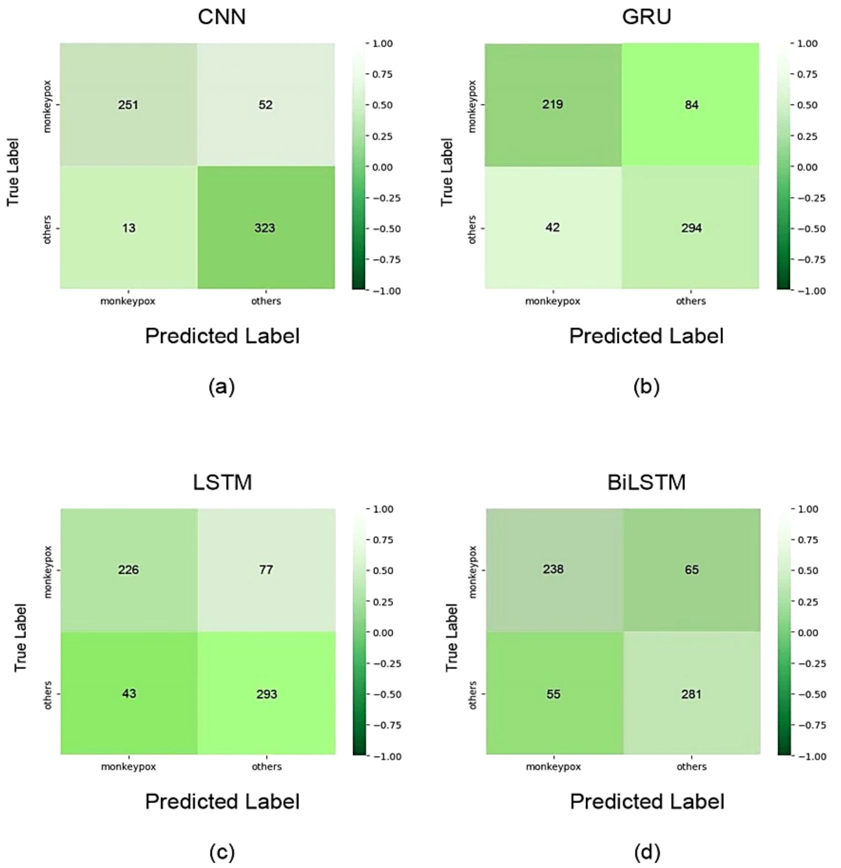


Figure 9. Confusion matrix results of monkeypox classification: (a) CNN Model, (b) GRU Model, (c) LSTM Model, (d) BiLSTM Model

The precision, recall, F1-score, and accuracy of our model were extracted from the confusion matrix to assess the robustness of our pre-trained model in binary categorization. CNN achieved the highest precision of 100%, followed by GRU, LSTM, and BiLSTM with accuracies of 99% each. This demonstrates the effectiveness of the methods employed in the classification process. The uniformity in performance evaluation metrics results across models suggests that they have reached a plateau in classification performance. The precision values for CNN, GRU, LSTM, and BiLSTM stand at 90%, 81%, 81%, and 81%, respectively. In terms of recall, CNN achieved 90%, GRU and LSTM both attained 80%, and BiLSTM reached 81%. Similarly, the f1-scores for CNN, GRU, LSTM, and BiLSTM are 90%, 80%, 81%, and 81%, respectively.

Our model effectively identifies both Monkeypox and non-Monkeypox cases, showcasing high precision. It has the capability to act as a significant tool for the rapid and accurate detection of Monkeypox in clinical applications. A crucial aspect of the proposed model is its capacity to achieve a remarkable accuracy rate of 100% in detecting Monkeypox cases. Utilizing the MobileNetV2 architecture, the CNN model ensures rapid and efficient processing for image classification duties.

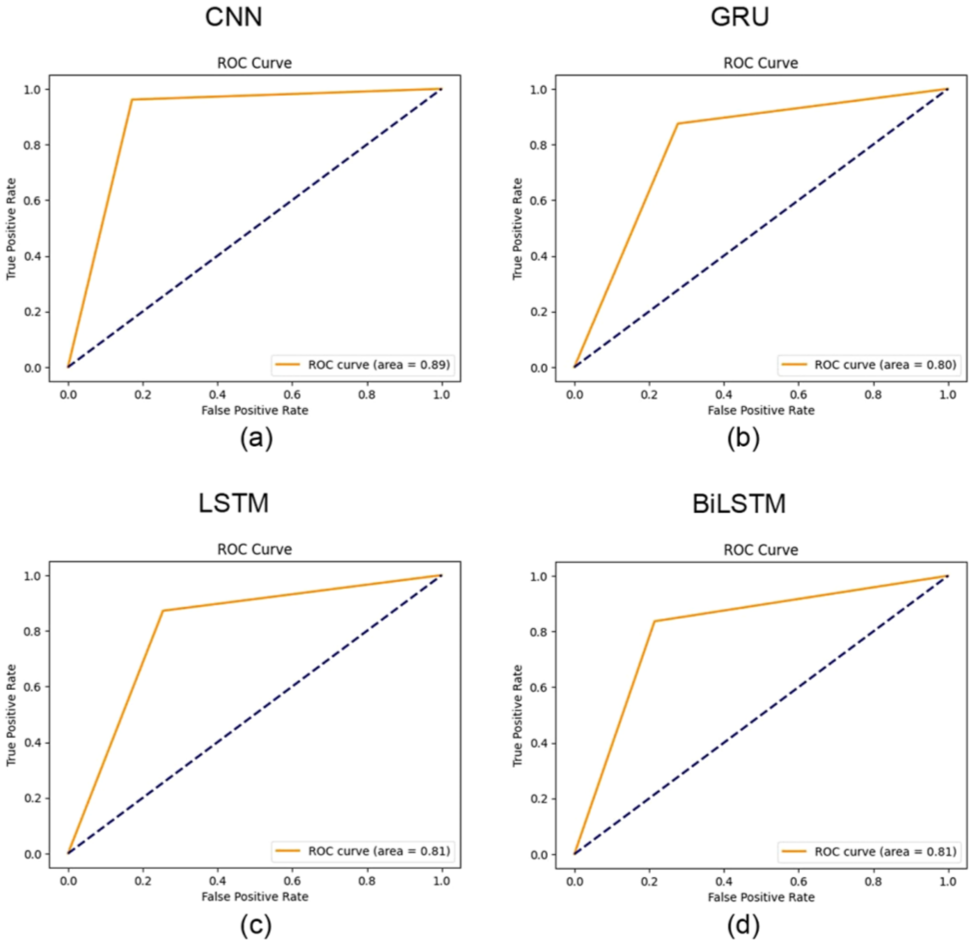


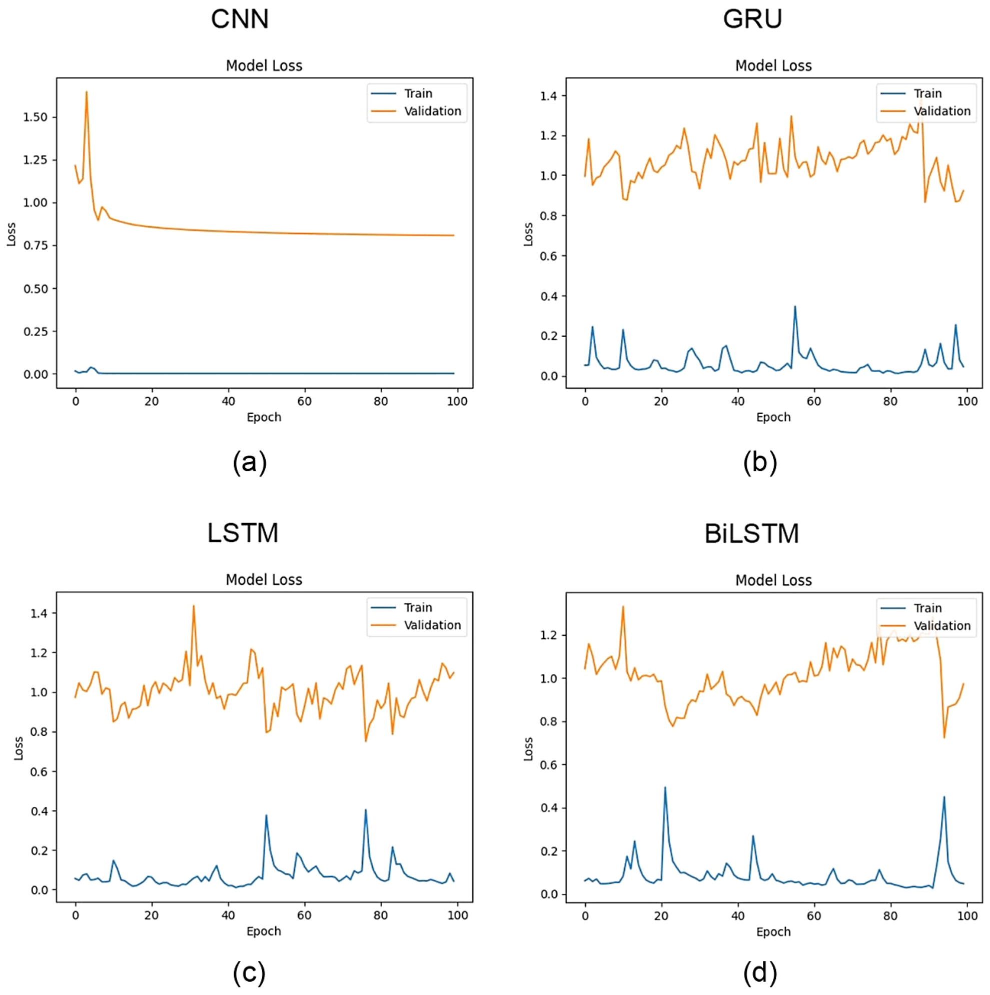
Figure 10. Deep learning model receiver operating characteristic (ROC) curves value: (a) CNN model, (b) GRU model, (c) LSTM model, (d) BiLSTM model

Figure 10 depicts the ROC curves generated by our models, encompassing CNN, GRU, LSTM, and BiLSTM. An ROC curve depicts a categorizing model's efficacy at different categorization criteria. Shown above, the ROC curves of our model demonstrate its performance across different classes. This graph visualizes two essential variables: True Positive Rate and False Positive Rate, serving as crucial evaluation metrics for assessing the success of any classification algorithm. Visually, the ROC is typically depicted as a curve plotting the true positive rate against the false positive rate for a given dataset. More precisely, The ROC approach involves charting recall values at different threshold levels and linking them to construct a curve. A curve sloping toward the top left corner reflects a classifier's greater ability to discriminate between positive and negative classifications.

The area under the ROC curve (AUC) is a simple statistic that summarizes a classifier's performance, reducing it to an individual measure. In contrast to the issue of comparing ROC curves, especially when they cross, the AUC allows models to be ranked based on their overall performance. Thus, the AUC is highly valued in the evaluation of models

[74, 75]. The estimation of the AUC employs several techniques, with the trapezoidal method being the most common. This method involves geometric calculations based on linear interpolation between points on the ROC curve. Alternatively, some researchers suggest approximating the AUC, particularly in binary learning scenarios, using Balanced Accuracy for simplicity [76, 77]. The AUC possesses a significant statistical characteristic: it reflects the likelihood that a classifier would score a selected by random positive case more than a picked at random negative case [78, 79]. The AUC serves as a complete assessment of efficacy throughout every feasible categorization criteria. It may be regarded as the chance that the model favors a random positive case over a random negative one. Specifically, the AUC values for CNN, GRU, LSTM, and BiLSTM are 89%, 80%, 81%, and 81%, respectively.

Figure 11(a) – 11(d) model loss curve gives us insights into how the model's efficacy improves over time by assessing the error or dissimilarity between its predicted output and the true output. The loss is the difference between the model's predicted and actual values. The model aims to minimize this loss, aiming to ensure its predictions closely align with the true values. So, the loss curve shows us how the model's error decreases as it learns, which indicates an improvement in its performance. The narrow blue line (‘*Train’)* represents the learning curve derived from the training dataset, providing insight into the model's learning progress, while the orange line (‘*Validation’)* represents the learning curve derived from a separate validation dataset, offering insight into the model's generalization capability.



(a) (b) (c) (d)

Figure 11. Model loss: (a) CNN model, (b) GRU model, (c) LSTM model, (d) BiLSTM model

Figure 12 illustrates the performance of the proposed models in classifying Monkeypox disease and other diseases based on accuracy and loss metrics. These models, including CNN, GRU, LSTM, and BiLSTM, are evaluated regarding their capacity to effectively classify instances and the extent of deviation between their predictions and the actual output, quantified by the loss metric. Loss serves as an indicator of the model's accuracy, with lower loss values indicating higher correctness. It is computed separately for validation and training datasets, reflecting the general efficacy of the model on these sets by aggregating the errors made across individual examples. Given the definitions of Accuracy and Loss, there should be an inverse relationship between the two: when Accuracy is high, Loss tends to be low, and vice versa.

Additionally, given that the weights and biases are chosen randomly, the precision pattern should start low (with large loss values, suggesting that the network is generating inaccurate estimations). However, as the network learns over numerous "epochs" (i.e., both forward and reverse runs of every training sample), the accuracy should gradually increase in subsequent iterations (resulting in lower loss values). The epoch is a selected variable, typically selected such that a loss reaches at least a minimum and does not worsen in the subsequent epochs. As a result, the accuracy value achieved is maximized. If it does not improve in the following epochs, it indicates that the network has attained stability and that more epochs will not boost performance. We set the number of epochs to a hundred, as both models achieve stability within or before this number of epochs. Specifically, the performance plots feature the following lines: The thin blue line ('loss') represents a training loss; the orange line ('accuracy') identifies a training precision; the green line ('val_loss') represents the result of validation loss; and the thin red line ('val_accuracy') shows the reliability of the validation.

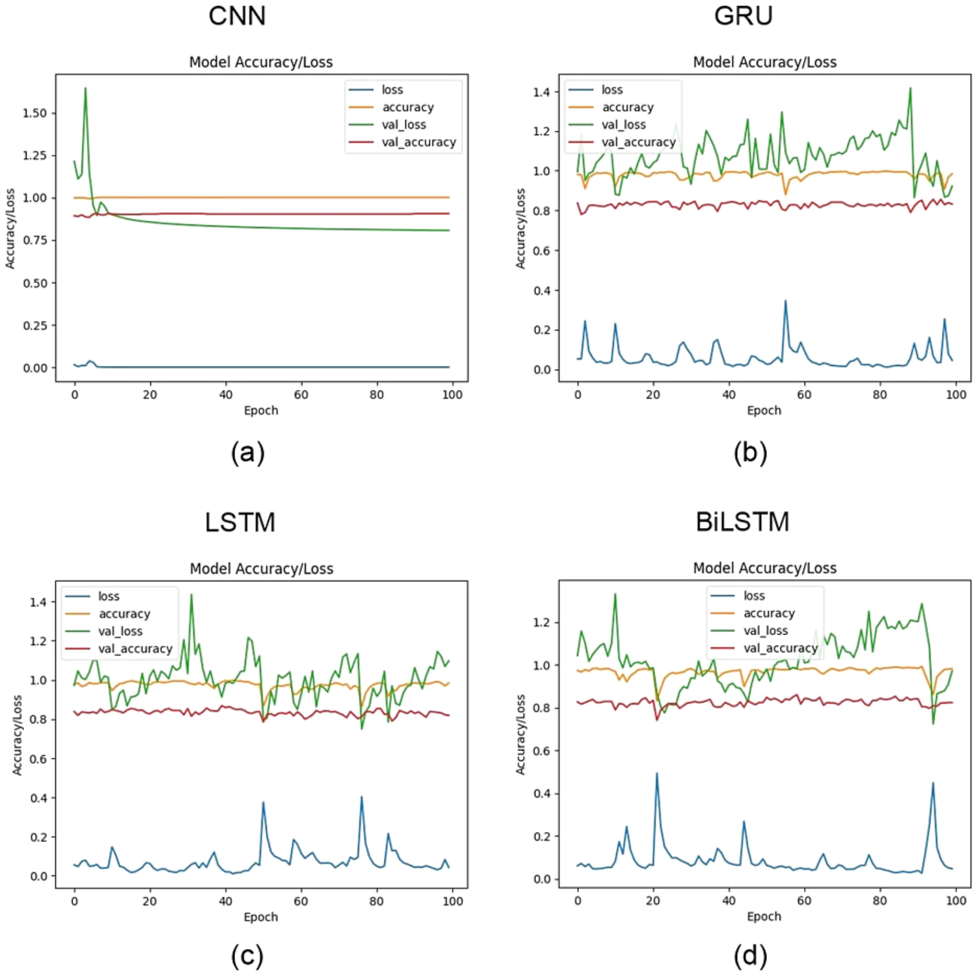


Figure 12. Model accuracy/loss: (a) CNN model, (b) GRU model, (c) LSTM model, (d) BiLSTM model

Figure 13, the accuracy curve, also known as the accuracy of the training curve, demonstrates the model's ability to make reliable forecasts on the training information as it is trained. Accuracy, expressed as a percentage, indicates the proportion of instances correctly classified by the model out of the total cases. Therefore, the accuracy curve shows how effectively its model matches the training set and improves its ability to make correct predictions. The thin blue line ('*Accuracy')* represents the training accuracy, while the orange line ('*Validation')* depicts the accuracy of the validation dataset.

Figures 11(a), 12(a), and 13(a) depict the performance plots of Model Loss, Model Accuracy/Loss, and Model Accuracy and Validation for the 100 epochs using the CNN model. As illustrated in Figure 12(a), the accuracy of the CNN model is 1. Figures 11(b), 12(b), and 13(b) exhibit the performance plots of Model Loss, Model Accuracy/Loss, and Model Accuracy and Validation for the 100 epochs using the GRU model. Similarly, Figures 11(c), 12(c), and 13(c) present the performance plots of Model Loss, Model Accuracy/Loss, and Model Accuracy and Validation on 100 epochs with the LSTM model. As indicated by Figure 12(c), the accuracy of the LSTM model is 1. Finally, Figures 11(d), 12(d), and 13(d) demonstrate the performance plots of Model Loss, Model Accuracy/Loss, and Model Accuracy and Validation for the 100 epochs using the BiLSTM model.

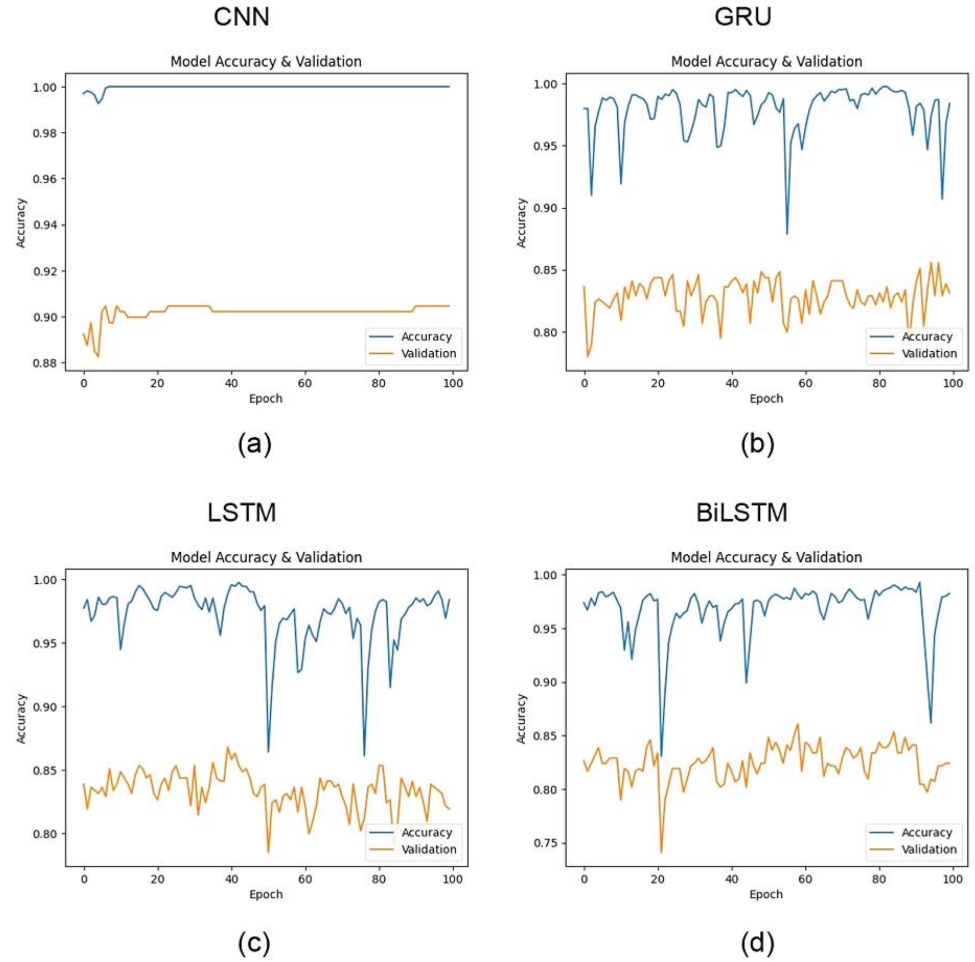


Figure 13. **Model accuracy and validation: (a) CNN model, (b) GRU model, (c) LSTM model, (d) BiLSTM model**

The overall performance of the proposed four models, which include CNN, GRU, LSTM, and BiLSTM, is shown in Figure 14. The study results indicate that the performance of the CNN and LSTM models is higher than that of the GRU and BiLSTM models. The proportion of true positives to true negatives among all classes measures a classifier's accuracy. The CNN model's performance and LSTM model's performance have an accuracy of 100%, while the GRU model's performance and BiLSTM model's performance have an accuracy of 99.88% and 99.45%. The precision, recall, F1 score, and AUC score achieved by the CNN model notably surpass those of the GRU, LSTM, and BiLSTM models.

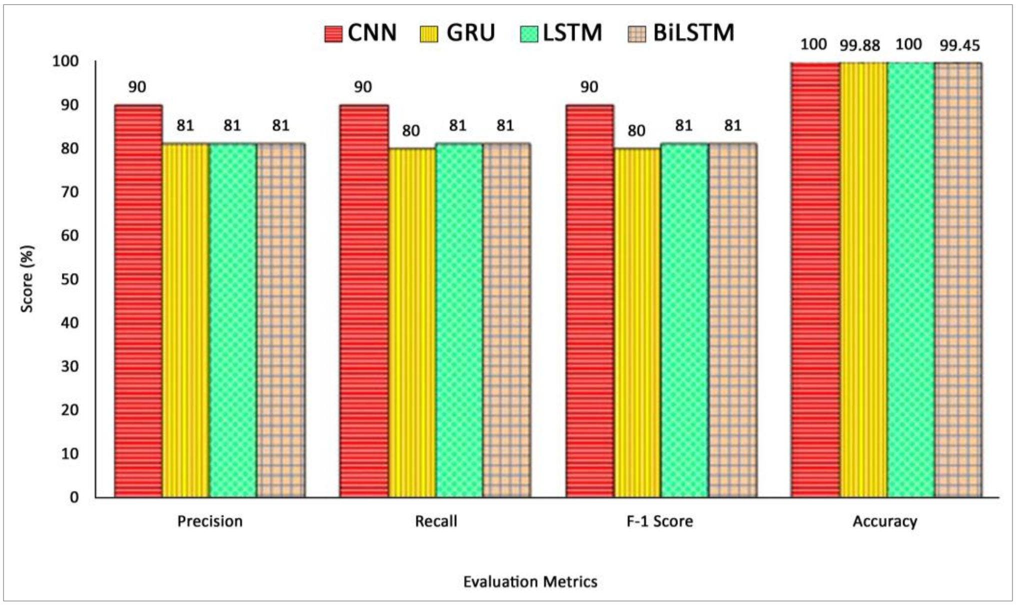


Figure 14. **Evaluation metrics**

In summary, the results suggest that CNN and LSTM models offer a promising avenue for enhancing the effectiveness of models in classifying Monkeypox. Precision is a crucial measure in model classification, indicating the proportion of true positives compared to all positives recognized by the classifier for a particular class. This metric inversely correlates with the count of false positives attributed to the class. The precision obtained by CNN, GRU, LSTM and BiLSTM is 90%, 81%, 81%, and 81%, respectively. CNN achieved the highest precision of 90% in classifying Monkeypox. Recall, formerly known as sensitivity, is a measure of a classifier's capability to accurately identify positive occurrences compared to the total of true positives and false negatives for the given class. This measure is oppositely proportional to a few false negative occurrences connected with the class. Recall plays a significant role in machine learning evaluations, particularly in highlighting false negative occurrences. The recall obtained by CNN, GRU, LSTM, and BiLSTM is 90%, 80%, 81%, and 81%, respectively. CNN obtained the highest precision of 90% for Monkeypox classification. The F1-Score given a particular class considers both recall and accuracy to assess the classifier's efficacy better. The F1-score is a statistic used to determine the effectiveness of a predictive model across a dataset, especially in binary categorization scenarios where Monkeypox is classified as 'positive' or 'negative'. This score is determined by taking a harmonic average of the model's accuracy and recall. CNN, GRU, LSTM, and BiLSTM attained F1-scores of 90%, 80%, 81%, and 81%, respectively. CNN notably achieved the highest F1-score of 90% among all models for Monkeypox classification.

A detailed summary of the performance measures is included in Table 2 for CNN, GRU, LSTM, and BiLSTM models. CNN demonstrated a precision, recall, and F1-score of 90%, along with the highest accuracy of 100%. GRU achieved a precision of 81%, recall of 80%, F1-score of 80%, and a peak accuracy of 99.88%. LSTM exhibited a precision, recall, and F1-score of 81%, along with a top accuracy of 100%. BiLSTM attained a precision, recall, and F1-score of 81%, with the best accuracy of 99.45%.

Table 2. Efficacy of classifier models

	Precision	Recall	F-1 Score	AUC Score	Accuracy
CNN	90%	90%	90%	89%	100%
GRU	81%	80%	80%	80%	99.88%
LSTM	81%	81%	81%	81%	100%
BiLSTM	81%	81%	81%	81%	99.45%

4-1- Explainable Deep Learning

To address the inherent opacity of deep learning classifiers, it is critical to employ visual approaches that reveal the learned features by CNNs during training. Such insights facilitate subsequent endeavors like fine-tuning and exploring alternative models to address erroneous feature acquisition and overfitting. Our study utilized a method for generating 'visual explanations' for decisions made by various CNN-based models, enhancing their interpretability. This technique, Grad-CAM, leverages the gradients of Monkeypox images streaming through the ultimate convolutional neural layer to produce an approximate localization map, highlighting essential parts in the visualization for notion prediction. Grad-CAM employs gradient data from CNN's final convolutional layer to assign significance levels to individual neurons during specific decision-making steps. Figure 15 illustrates Grad-CAM explanations, shedding light on the models' interpretability.

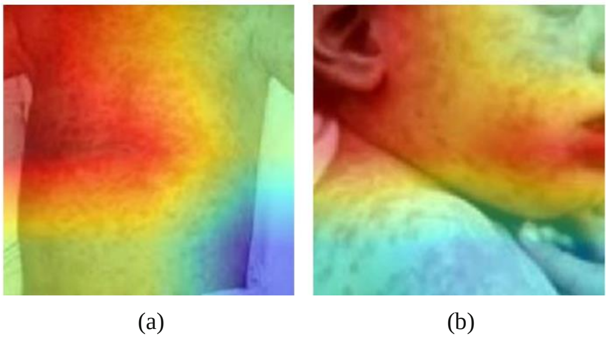

(a) (b)

Figure 15. Grad-CAM explanations. Image (a): monkeypox, image (b): non monkeypox

The visualization method employed, LIME, is widely utilized to illuminate the predictions generated by convolutional neural networks. LIME operates by collecting input picture data and applying random perturbations to figure out methods underlying the forecasts. The XAI visual representation focuses on MobileNetV2, the top-performing deep learning model. Figures 16 and 17 provide insights into LIME explanations. In the machine learning academia, interpreting a deep learning model typically involves discerning the significance of features. This entails determining which aspects of the input features of a given data point contribute to the prediction outcome at the output layer and/or the heightened activation of an internal layer or node. Machine learning and artificial intelligence have devised novel methodologies to address this challenge. Techniques such as perturbation experiments [80] and saliency map-based methods [41] have demonstrated their efficacy in elucidating which sections of the input image exert the most influence on a model's ultimate prediction. Input perturbation assesses the degree to which regions of the input image identified as significant by XAI tools fulfill that role. The underlying concept is as follows: With a trained model in place, a test image is employed as input to generate its heatmap using an XAI technique tailored for that deep network.

Additionally, the most pertinent regions, clusters of pixels, undergo alterations in their values within the original (input) image, using uniformly and randomly generated values. Subsequently, the adjusted image is reintroduced to the network for classification purposes. This process is iterated several times to significantly increase the number of altered patches. The assumption is that the model's performance will deteriorate as the count of modified patches rises. A proficient XAI technique is anticipated to be more impacted by these modifications than a less effective one.

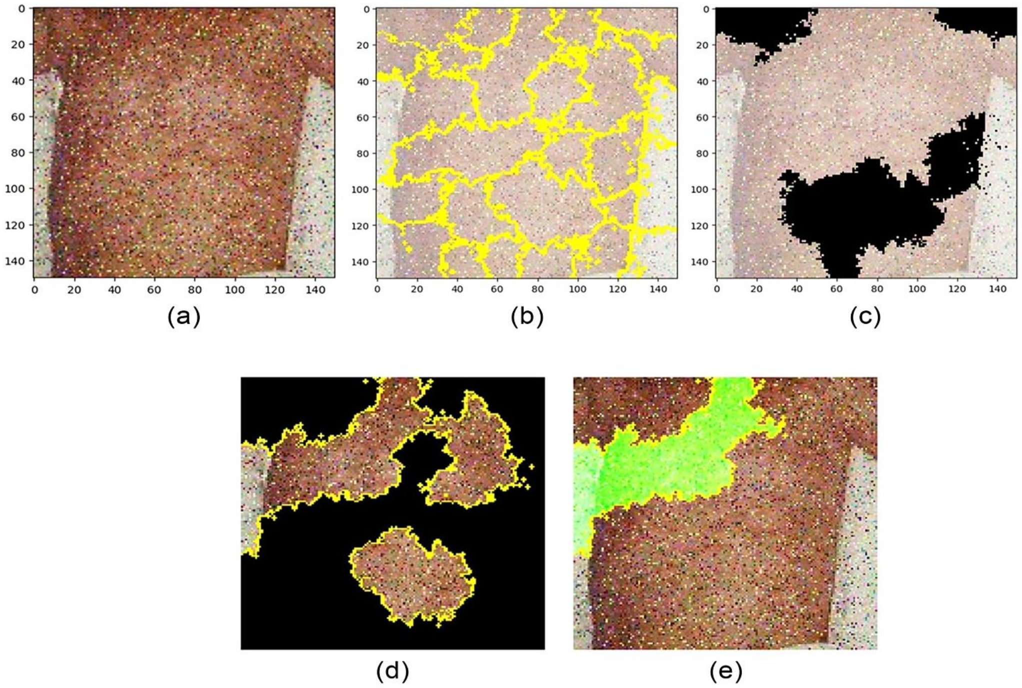


Figure 16. **Training explanations for top-performing using LIME for monkeypox images**

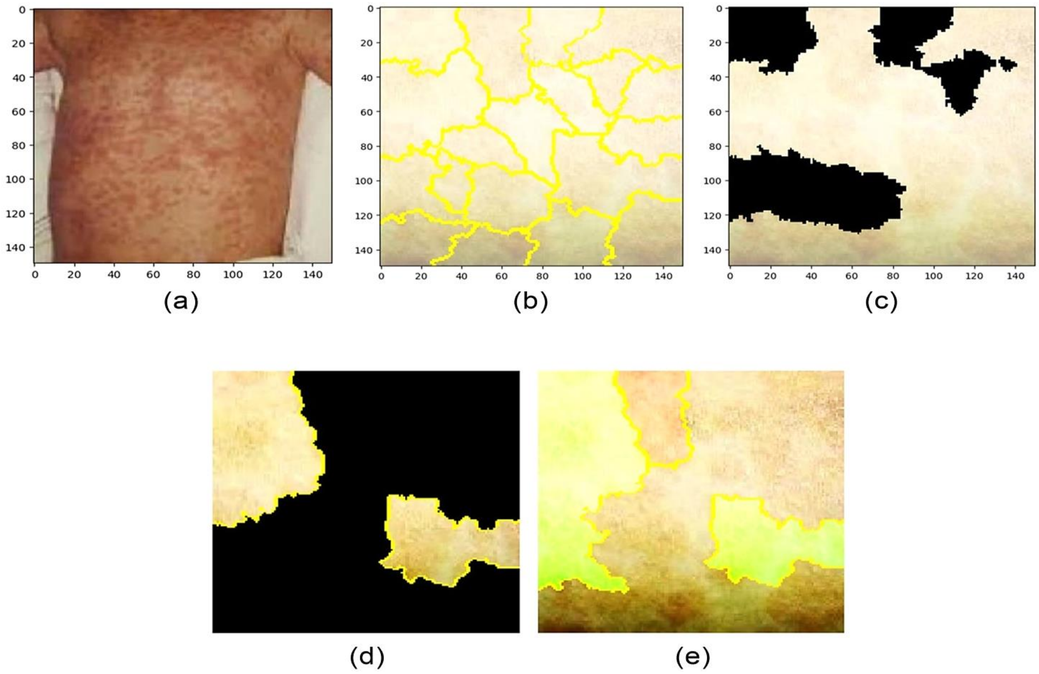


Figure 17. **Training explanations for top-performing using LIME for non monkeypox images**

A novel approach known as LIME [81] has emerged, effectively constructing a locally linear model from a complex one. This enables the interpretation of the weights in the linear combination as indicators of feature importance. Moreover, a classical method known as influence functions has efficiently pinpointed the training data instances that impact specific prediction outputs most. Integrating these advanced computational techniques with interactive visualizations holds considerable promise for enhancing the interpretability of deep learning, although it presents a significant challenge in practical applications [82].

5- Conclusion

This work presents an explainable deep learning method for identifying and categorizing Monkeypox ailment through skin lesion image data. Employing an openly accessible dataset, we assessed four models—CNN, GRU, LSTM, and BiLSTM—on the dataset. Incorporating interpretative artificial intelligence methods like LIME and Grad-CAM allows for visual comprehension of the model's predictions, aiding healthcare professionals in utilizing the model. The CNN model's performance and LSTM model's performance have an accuracy of 100%, while the GRU model's performance and BiLSTM model's performance have an accuracy of 99.88% and 99.45%. The results confirm the applicability of deep learning models, including the suggested CNN model utilizing the pre-trained MobileNetV2 and LSTM, in combating the Monkeypox virus, highlighting their potential significance in this endeavor. To bolster trust and transparency, we integrated LIME and Grad-CAM techniques, providing perspectives on the choice-making process of deep learning model algorithms. Future research endeavors will explore alternative deep learning or machine learning algorithms, building upon the findings of this study. This contribution enriches the existing literature in the field, paving the way for further investigations.

6- Declarations

6-1- Author Contributions

Conceptualization, A.M. and C.A.R.; methodology, A.M.; software, A.M.; validation, A.M., M.H., and C.A.R.; formal analysis, A.M. and M.H.; investigation, A.M. and C.A.R.; resources, A.M.; data curation, A.M.; writing—original draft preparation, A.M.; writing—review and editing, A.M., M.H., and C.A.R.; visualization, A.M.; supervision, M.H. and C.A.R.; project administration, C.A.R.; funding acquisition, C.A.R. All authors have read and agreed to the published version of the manuscript.

6-2- Data Availability Statement

The data presented in this study are available on request from the corresponding author.

6-3- Funding and Acknowledgments

This research project is supported by Second Century Fund (C2F), Chulalongkorn University, Thailand. We gratefully appreciate this support.

6-4- Institutional Review Board Statement

Not Applicable.

6-5- Informed Consent Statement

Not Applicable.

6-6- Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this manuscript. In addition, the ethical issues, including plagiarism, informed consent, misconduct, data fabrication and/or falsification, double publication and/or submission, and redundancies have been completely observed by the authors.

7- References

[1] WHO. (2022). WHO Director-General declares the ongoing monkeypox outbreak a Public Health Emergency of International Concern. World Health Organisation, Geneva, Switzerland. Available online: https://www.who.int/europe/news/item/23-07-2022-who-director-general-declares-the-ongoing-monkeypox-outbreak-a-public-health-event-of-international-concern (accessed on May 2024).

[2] WHO. (2023). Statement on the fifteenth meeting of the IHR (2005) Emergency Committee on the COVID-19 pandemic. WHO Director General’s Speeches, World Health Organisation, Geneva, Switzerland.

[3] Sadeuh-Mba, S. A., Yonga, M. G., Els, M., Batejat, C., Eyangoh, S., Caro, V., Etoundi, A., Carniel, E., & Njouom, R. (2019). Monkeypox virus phylogenetic similarities between a human case detected in Cameroon in 2018 and the 2017-2018 outbreak in Nigeria. Infection, Genetics and Evolution, 69, 8–11. doi:10.1016/j.meegid.2019.01.006.

[4] Adler, H., Gould, S., Hine, P., Snell, L. B., Wong, W., Houlihan, C. F., Osborne, J. C., Rampling, T., Beadsworth, M. B., Duncan, C. J., Dunning, J., Fletcher, T. E., Hunter, E. R., Jacobs, M., Khoo, S. H., Newsholme, W., Porter, D., Porter, R. J., Ratcliffe, L., … Hruby, D. E. (2022). Clinical features and management of human monkeypox: a retrospective observational study in the UK. The Lancet Infectious Diseases, 22(8), 1153–1162. doi:10.1016/s1473-3099(22)00228-6.

[5] Girometti, N., Byrne, R., Bracchi, M., Heskin, J., McOwan, A., Tittle, V., Gedela, K., Scott, C., Patel, S., Gohil, J., Nugent, D., Suchak, T., Dickinson, M., Feeney, M., Mora-Peris, B., Stegmann, K., Plaha, K., Davies, G., Moore, L. S. P., … Whitlock, G. (2022). Demographic and clinical characteristics of confirmed human monkeypox virus cases in individuals attending a sexual health centre in London, UK: an observational analysis. The Lancet Infectious Diseases, 22(9), 1321–1328. doi:10.1016/S1473-3099(22)00411-X.

[6] Bryer, J., Freeman, E. E., & Rosenbach, M. (2022). Monkeypox emerges on a global scale: A historical review and dermatologic primer. Journal of the American Academy of Dermatology, 87(5), 1069–1074. doi:10.1016/j.jaad.2022.07.007.

[7] Lai, C. C., Hsu, C. K., Yen, M. Y., Lee, P. I., Ko, W. C., & Hsueh, P. R. (2022). Monkeypox: An emerging global threat during the COVID-19 pandemic. Journal of Microbiology, Immunology and Infection, 55(5), 787–794. doi:10.1016/j.jmii.2022.07.004.

[8] Reynolds, M. G., Yorita, K. L., Kuehnert, M. J., Davidson, W. B., Huhn, G. D., Holman, R. C., & Damon, I. K. (2006). Clinical manifestations of human monkeypox influenced by route of infection. Journal of Infectious Diseases, 194(6), 773–780. doi:10.1086/505880.

[9] Hemati, S., Farhadkhani, M., Sanami, S., & Mohammadi-Moghadam, F. (2022). A review on insights and lessons from COVID-19 to the prevent of monkeypox pandemic. Travel Medicine and Infectious Disease, 50, 102441. doi:10.1016/j.tmaid.2022.102441.

[10] CDC. (2023). 2022-2023 Mpox Outbreak Global Map. Centers for Disease Control and Prevention, Georgia, United States. Available online: https://archive.cdc.gov/#/details?url=https://www.cdc.gov/poxvirus/mpox/response/2022/world-map.html. (accessed on May 2024).

[11] Kasarla, R. R. (2022). Human Monkeypox: An Emerging and Neglected Viral Zoonosis of Public Health Concern. Journal of Universal College of Medical Sciences, 10(01), 1–3. doi:10.3126/jucms.v10i01.47113.

[12] WHO. (2023). Multi-country outbreak of mpox, External situation report #29-20 October 2023. World Health Organisation, Geneva, Switzerland. Available online: https://www.who.int/publications/m/item/multi-country-outbreak-of-mpox--external-situation-report-29---20-october-2023 (accessed on May 2024).

[13] WHO. (2023). Multi-country outbreak of mpox, External situation report #28-19 September 2023. World Health Organisation, Geneva, Switzerland. Available online: https://www.who.int/publications/m/item/multicountry-outbreak-of-mpox--external-situation-report-28---19-september-2023 (accessed on May 2024).

[14] Gessain, A., Nakoune, E., & Yazdanpanah, Y. (2022). Monkeypox. New England Journal of Medicine, 387(19), 1783-1793. doi:10.1056/NEJMra2208860.

[15] Nakhaie, M., Arefinia, N., Charostad, J., Bashash, D., Haji Abdolvahab, M., & Zarei, M. (2023). Monkeypox virus diagnosis and laboratory testing. Reviews in medical virology, 33(1), e2404. doi:10.1002/rmv.2404.

[16] Upadhayay, S., Arthur, R., Soni, D., Yadav, P., Navik, U., Singh, R., ... & Kumar, P. (2022). Monkeypox infection: The past, present, and future. International immunopharmacology, 113, 109382. doi:10.1016/j.intimp.2022.109382.

[17] Aljabali, A. A., Obeid, M. A., Nusair, M. B., Hmedat, A., & Tambuwala, M. M. (2022). Monkeypox virus: An emerging epidemic. Microbial Pathogenesis, 173, 105794. doi:10.1016/j.micpath.2022.105794.

[18] Kmiec, D., & Kirchhoff, F. (2022). Monkeypox: a new threat? International Journal of Molecular Sciences, 23(14), 7866. doi:10.3390/ijms23147866.

[19] Hernaez, B., Muñoz-Gómez, A., Sanchiz, A., Orviz, E., Valls-Carbo, A., Sagastagoitia, I., Ayerdi, O., Martín, R., Puerta, T., Vera, M., Cabello, N., Vergas, J., Prieto, C., Pardo-Figuerez, M., Negredo, A., Lagarón, J. M., del Romero, J., Estrada, V., & Alcamí, A. (2023). Monitoring monkeypox virus in saliva and air samples in Spain: a cross-sectional study. The Lancet Microbe, 4(1), e21–e28. doi:10.1016/S2666-5247(22)00291-9.

[20] 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 of North America, 33(4), 1027–1043. doi:10.1016/j.idc.2019.03.001.

[21] Altindis, M., Puca, E., & Shapo, L. (2022). Diagnosis of monkeypox virus–An overview. Travel medicine and infectious disease, 50, 102459. doi:10.1016/j.tmaid.2022.102459.

[22] Qayyum, A., Qadir, J., Bilal, M., & Al-Fuqaha, A. (2021). Secure and Robust Machine Learning for Healthcare: A Survey. IEEE Reviews in Biomedical Engineering, 14, 156–180. doi:10.1109/RBME.2020.3013489.

[23] Ozaydin, B., Berner, E. S., & Cimino, J. J. (2021). Appropriate use of machine learning in healthcare. Intelligence-Based Medicine, 5, 100041. doi:10.1016/j.ibmed.2021.100041.

[24] Naemi, A., Schmidt, T., Mansourvar, M., Naghavi-Behzad, M., Ebrahimi, A., & Wiil, U. K. (2021). Machine learning techniques for mortality prediction in emergency departments: A systematic review. BMJ Open, 11(11), 52663. doi:10.1136/bmjopen-2021-052663.

[25] 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(1), 438. doi:10.1186/s12879-023-08408-4.

[26] Panayides, A. S., Amini, A., Filipovic, N. D., Sharma, A., Tsaftaris, S. A., Young, A., Foran, D., Do, N., Golemati, S., Kurc, T., Huang, K., Nikita, K. S., Veasey, B. P., Zervakis, M., Saltz, J. H., & Pattichis, C. S. (2020). AI in Medical Imaging Informatics: Current Challenges and Future Directions. IEEE Journal of Biomedical and Health Informatics, 24(7), 1837–1857. doi:10.1109/JBHI.2020.2991043.

[27] Dash, S., Acharya, B. R., Mittal, M., Abraham, A., & Kelemen, A. (2020). Deep learning techniques for biomedical and health informatics. Springer International Publishing, Cham, Switzerland. doi:10.1007/978-3-030-33966-1.

[28] Wang, R., Li, C., Fu, W., & Tang, G. (2020). Deep Learning Method Based on Gated Recurrent Unit and Variational Mode Decomposition for Short-Term Wind Power Interval Prediction. IEEE Transactions on Neural Networks and Learning Systems, 31(10), 3814–3827. doi:10.1109/TNNLS.2019.2946414.

[29] Abdelhamid, A. A., El-Kenawy, E. S. M., Khodadadi, N., Mirjalili, S., Khafaga, D. S., Alharbi, A. H., Ibrahim, A., Eid, M. M., & Saber, M. (2022). Classification of Monkeypox Images Based on Transfer Learning and the Al-Biruni Earth Radius Optimization Algorithm. Mathematics, 10(19), 3614. doi:10.3390/math10193614.

[30] Akin, K.D., Gurkan, C., Budak, A., & Karatas¸ H. (2022). Classification of monkeypox skin lesion using the explainable artificial intelligence assisted convolutional neural networks. European Journal of Science and Technology 2022;(40):106–10. https://doi.org/10.31590/ejosat.1171816.

[31] 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(11). doi:10.1007/s10916-022-01863-7.

[32] Sitaula, C., & Shahi, T. B. (2022). Monkeypox Virus Detection Using Pre-trained Deep Learning-based Approaches. Journal of Medical Systems, 46(11). doi:10.1007/s10916-022-01868-2.

[33] Nayak, T., Chadaga, K., Sampathila, N., Mayrose, H., Gokulkrishnan, N., Bairy G, M., Prabhu, S., S, S. K., & Umakanth, S. (2023). Deep learning based detection of monkeypox virus using skin lesion images. Medicine in Novel Technology and Devices, 18. doi:10.1016/j.medntd.2023.100243.

[34] Alakus, T. B., & Baykara, M. (2022). Comparison of Monkeypox and Wart DNA Sequences with Deep Learning Model. Applied Sciences (Switzerland), 12(20), 10216. doi:10.3390/app122010216.

[35] Eid, M. M., El-Kenawy, E. S. M., Khodadadi, N., Mirjalili, S., Khodadadi, E., Abotaleb, M., Alharbi, A. H., Abdelhamid, A. A., Ibrahim, A., Amer, G. M., Kadi, A., & Khafaga, D. S. (2022). Meta-Heuristic Optimization of LSTM-Based Deep Network for Boosting the Prediction of Monkeypox Cases. Mathematics, 10(20), 3845. doi:10.3390/math10203845.

[36] Khafaga, D. S., Ibrahim, A., El-Kenawy, E. S. M., Abdelhamid, A. A., Karim, F. K., Mirjalili, S., Khodadadi, N., Lim, W. H., Eid, M. M., & Ghoneim, M. E. (2022). An Al-Biruni Earth Radius Optimization-Based Deep Convolutional Neural Network for Classifying Monkeypox Disease. Diagnostics, 12(11), 2892. doi:10.3390/diagnostics12112892.

[37] Manohar, B., & Das, R. (2022). Artificial Neural Networks for the Prediction of Monkeypox Outbreak. Tropical Medicine and Infectious Disease, 7(12), 424. doi:10.3390/tropicalmed7120424.

[38] Singh, R. K., Pandey, R., & Babu, R. N. (2021). COVIDScreen: explainable deep learning framework for differential diagnosis of COVID-19 using chest X-rays. Neural Computing and Applications, 33(14), 8871–8892. doi:10.1007/s00521-020-05636-6.

[39] Khafaga, D. S., Ibrahim, A., El-Kenawy, E. S. M., Abdelhamid, A. A., Karim, F. K., Mirjalili, S., ... & Ghoneim, M. E. (2022). An Al-Biruni earth radius optimization-based deep convolutional neural network for classifying monkeypox disease. Diagnostics, 12(11), 2892. doi:10.3390/diagnostics12112892.

[40] van der Velden, B. H. M., Kuijf, H. J., Gilhuijs, K. G. A., & Viergever, M. A. (2022). Explainable artificial intelligence (XAI) in deep learning-based medical image analysis. Medical Image Analysis, 79, 102470. doi:10.1016/j.media.2022.102470.

[41] Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). “Why Should I Trust You?” Explaining the Predictions of Any Classifier. NAACL-HLT 2016 - 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Proceedings of the Demonstrations Session, 97–101. doi:10.18653/v1/n16-3020.

[42] Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., & Batra, D. (2020). Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. International Journal of Computer Vision, 128(2), 336–359. doi:10.1007/s11263-019-01228-7.

[43] Kaggle (20224). Data_Monkeypox. Available online: https://www.kaggle.com/datasets/piyush19/monkeypox-dataset (accessed on May 2024).

[44] Lewy, D., & Mańdziuk, J. (2023). An overview of mixing augmentation methods and augmentation strategies. Artificial Intelligence Review, 56(3), 2111–2169. doi:10.1007/s10462-022-10227-z.

[45] Maseleno, A., Kavitha, D., Ashok, K., Al Ansari, M. S., Satheesh, N., & Reddy, R. V. K. (2023). An Ensemble Learning Approach for Multi-Modal Medical Image Fusion using Deep Convolutional Neural Networks. International Journal of Advanced Computer Science and Applications, 14(8), 758–769. doi:10.14569/IJACSA.2023.0140884.

[46] Ismail Fawaz, H., Forestier, G., Weber, J., Idoumghar, L., & Muller, P. A. (2019). Deep learning for time series classification: a review. Data Mining and Knowledge Discovery, 33(4), 917–963. doi:10.1007/s10618-019-00619-1.

[47] Miau, S., & Hung, W. H. (2020). River flooding forecasting and anomaly detection based on deep learning. IEEE Access, 8, 198384–198402. doi:10.1109/ACCESS.2020.3034875.

[48] Ghimire, S., Yaseen, Z. M., Farooque, A. A., Deo, R. C., Zhang, J., & Tao, X. (2021). Streamflow prediction using an integrated methodology based on convolutional neural network and long short-term memory networks. Scientific Reports, 11(1), 1–26. doi:10.1038/s41598-021-96751-4.

[49] Mittal, A., Kumar, D., Mittal, M., Saba, T., Abunadi, I., Rehman, A., & Roy, S. (2020). Detecting pneumonia using convolutions and dynamic capsule routing for chest X-ray images. Sensors (Switzerland), 20(4), 1068. doi:10.3390/s20041068.

[50] Ruiz, L., Gama, F., & Ribeiro, A. (2020). Gated Graph Recurrent Neural Networks. IEEE Transactions on Signal Processing, 68, 6303–6318. doi:10.1109/TSP.2020.3033962.

[51] Bibi, I., Akhunzada, A., Malik, J., Iqbal, J., Mussaddiq, A., & Kim, S. (2020). A Dynamic DL-Driven Architecture to Combat Sophisticated Android Malware. IEEE Access, 8, 129600–129612. doi:10.1109/ACCESS.2020.3009819.

[52] Gao, S., Zheng, Y., & Guo, X. (2020). Gated recurrent unit-based heart sound analysis for heart failure screening. BioMedical Engineering Online, 19(1), 1–17. doi:10.1186/s12938-020-0747-x.

[53] Zhang, C., Ji, C., Hua, L., Ma, H., Nazir, M. S., & Peng, T. (2022). Evolutionary quantile regression gated recurrent unit network based on variational mode decomposition, improved whale optimization algorithm for probabilistic short-term wind speed prediction. Renewable Energy, 197, 668-682. doi:10.1016/j.renene.2022.07.123.

[54] Tian, L., Li, X., Ye, Y., Xie, P., & Li, Y. (2020). A Generative Adversarial Gated Recurrent Unit Model for Precipitation Nowcasting. IEEE Geoscience and Remote Sensing Letters, 17(4), 601–605. doi:10.1109/LGRS.2019.2926776.

[55] Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. Neural Computation, 9(8), 1735–1780. doi:10.1162/neco.1997.9.8.1735.

[56] Vidal, A., & Kristjanpoller, W. (2020). Gold volatility prediction using a CNN-LSTM approach. Expert Systems with Applications, 157, 113481. doi:10.1016/j.eswa.2020.113481.

[57] Boulila, W., Ghandorh, H., Khan, M. A., Ahmed, F., & Ahmad, J. (2021). A novel CNN-LSTM-based approach to predict urban expansion. Ecological Informatics, 64, 101325. doi:10.1016/j.ecoinf.2021.101325.

[58] Wang, D., Bai, Y., Wu, C., Li, Y., Shang, C., & Shen, Q. (2022). Convolutional LSTM-Based Hierarchical Feature Fusion for Multispectral Pan-Sharpening. IEEE Transactions on Geoscience and Remote Sensing, 60, 1–16. doi:10.1109/TGRS.2021.3104221.

[59] Han, M., Chen, W., & Moges, A. D. (2019). Fast image captioning using LSTM. Cluster Computing, 22, 6143–6155. doi:10.1007/s10586-018-1885-9.

[60] Amin, J., Sharif, M., Raza, M., Saba, T., Sial, R., & Shad, S. A. (2020). Brain tumor detection: a long short-term memory (LSTM)-based learning model. Neural Computing and Applications, 32, 15965-15973. doi:10.1007/s00521-019-04650-7.

[61] Islam, M. Z., Islam, M. M., & Asraf, A. (2020). A combined deep CNN-LSTM network for the detection of novel coronavirus (COVID-19) using X-ray images. Informatics in Medicine Unlocked, 20, 100412. doi:10.1016/j.imu.2020.100412.

[62] Demir, F. (2021). DeepCoroNet: A deep LSTM approach for automated detection of COVID-19 cases from chest X-ray images. Applied Soft Computing, 103, 107160. doi:10.1016/j.asoc.2021.107160.

[63] Yu, Y., Si, X., Hu, C., & Zhang, J. (2019). A review of recurrent neural networks: LSTM cells and network architectures. Neural Computation, 31(7), 1235–1270. doi:10.1162/neco_a_01199.

[64] Sahoo, B. B., Jha, R., Singh, A., & Kumar, D. (2019). Long short-term memory (LSTM) recurrent neural network for low-flow hydrological time series forecasting. Acta Geophysica, 67(5), 1471–1481. doi:10.1007/s11600-019-00330-1.

[65] Zhou, D., Zuo, X., & Zhao, Z. (2022). Constructing a Large-Scale Urban Land Subsidence Prediction Method Based on Neural Network Algorithm from the Perspective of Multiple Factors. Remote Sensing, 14(8), 1803. doi:10.3390/rs14081803.

[66] Siami-Namini, S., Tavakoli, N., & Namin, A. S. (2019). The Performance of LSTM and BiLSTM in Forecasting Time Series. Proceedings - IEEE International Conference on Big Data 2019, 3285–3292. doi:10.1109/BigData47090.2019.9005997.

[67] Graves, A., & Schmidhuber, J. (2005). Framewise phoneme classification with bidirectional LSTM and other neural network architectures. Neural Networks, 18(5–6), 602–610. doi:10.1016/j.neunet.2005.06.042.

[68] Liang, D., Liang, H., Yu, Z., & Zhang, Y. (2020). Deep convolutional BiLSTM fusion network for facial expression recognition. Visual Computer, 36(3), 499–508. doi:10.1007/s00371-019-01636-3.

[69] Yang, L., & Zhao, Q. (2022). A BiLSTM Based Pipeline Leak Detection and Disturbance Assisted Localization Method. IEEE Sensors Journal, 22(1), 611–620. doi:10.1109/JSEN.2021.3128816.

[70] Zhang, P., Yang, Y., & Yin, Z.-Y. (2021). BiLSTM-Based Soil–Structure Interface Modeling. International Journal of Geomechanics, 21(7), 4021096. doi:10.1061/(asce)gm.1943-5622.0002058.

[71] Li, Y. H., Harfiya, L. N., Purwandari, K., & Lin, Y. Der. (2020). Real-time cuffless continuous blood pressure estimation using deep learning model. Sensors (Switzerland), 20(19), 1–19. doi:10.3390/s20195606.

[72] Liu, W., Jing, W., & Li, Y. (2020). Incorporating feature representation into BiLSTM for deceptive review detection. Computing, 102(3), 701–715. doi:10.1007/s00607-019-00763-y.

[73] Jiao, Y., & Du, P. (2016). Performance measures in evaluating machine learning based bioinformatics predictors for classifications. Quantitative Biology, 4(4), 320–330. doi:10.1007/s40484-016-0081-2.

[74] Pereira, R. M., Bertolini, D., Teixeira, L. O., Silla, C. N., & Costa, Y. M. G. (2020). COVID-19 identification in chest X-ray images on flat and hierarchical classification scenarios. Computer Methods and Programs in Biomedicine, 194, 105532. doi:10.1016/j.cmpb.2020.105532.

[75] Luque, A., Carrasco, A., Martín, A., & de las Heras, A. (2019). The impact of class imbalance in classification performance metrics based on the binary confusion matrix. Pattern Recognition, 91, 216–231. doi:10.1016/j.patcog.2019.02.023.

[76] Chicco, D., & Jurman, G. (2020). The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. BMC Genomics, 21(1), 1–13. doi:10.1186/s12864-019-6413-7.

[77] Tharwat, A. (2018). Classification assessment methods. Applied Computing and Informatics, 17(1), 168–192. doi:10.1016/j.aci.2018.08.003.

[78] Caton, S., & Haas, C. (2024). Fairness in Machine Learning: A Survey. ACM Computing Surveys 56(7), 1-38. doi:10.1145/3616865.

[79] 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), 1–74. doi:10.1186/s40537-021-00444-8.

[80] Zeiler, M. D., & Fergus, R. (2014). Visualizing and understanding convolutional networks. Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 8689 LNCS (Part 1), 818–833. doi:10.1007/978-3-319-10590-1_53.

[81] Simonyan, K., Vedaldi, A., & Zisserman, A. (2014). Deep inside convolutional networks: Visualising image classification models and saliency maps. 2nd International Conference on Learning Representations, ICLR 2014 - Workshop Track Proceedings, Oxford, United Kingdom.

[82] Koh, P. W., & Liang, P. (2017). Understanding black-box predictions via influence functions. 34th International Conference on Machine Learning, PMLR: 70, 1885-1894.

Appendix I: List of Acronyms

Acronyms	Descriptions	Acronyms	Descriptions
ANN	Artificial Neural Network	MLPs	Multi-Layer Perceptrons
AUC	Area Under the ROC Curve	UK	United Kingdom
BiLSTM	Bidirectional Long Short Term Memory	PCR	Polymerase Chain Reaction
CNN	Convolutional Neural Network	PHE	Public Health Emergency
Covid-19	Coronavirus Disease	PHEIC	Public Health Emergency Of International Concern
DNA	Deoxyribonucleic Acid	ReLU	Rectified Linear Unit
DRC	Democratic Republic of the Congo	RGB	Red, Green, Blue
DT	Decision Tree	RNN	Recurrent Neural Network
FNs	False Negatives	ROC	Receiver Operating Characteristics
FPs	False Positives	SVM	Support Vector Machine
GPU	Graphics Processing Units	TNs	True Negatives
Grad-CAM	Gradient-Weighted Class Activation Mapping	TPR	True Positive Rate
GRU	Gated Recurrent Unit	TPs	True Positives
IDE	Integrated Development Environment	WHO	World Health Organization
K-NN	K-Nearest Neighbors	XAI	Explainable Artificial Intelligence
LIME	Local Interpretable Model-Agnostic Explanations	LSTM	Long Short Term Memory