Journal of Imaging Informatics in Medicine (2024) 37:778–800
https://doi.org/10.1007/s10278-023-00941-7

Enhancing Disease Classification with Deep Learning: a Two-Stage Optimization Approach for Monkeypox and Similar Skin Lesion Diseases

Serkan Savaş[1]

Received: 27 July 2023 / Revised: 26 September 2023 / Accepted: 17 October 2023 / Published online: 12 January 2024

Abstract
Monkeypox (MPox) is an infectious disease caused by the monkeypox virus, presenting challenges in accurate identification due to its resemblance to other diseases. This study introduces a deep learning-based method to distinguish visually similar diseases, specifically MPox, chickenpox, and measles, addressing the 2022 global MPox outbreak. A two-stage optimization approach was presented in the study. By analyzing pre-trained deep neural networks including 71 models, this study optimizes accuracy through transfer learning, fine-tuning, and ensemble learning techniques. ConvNeXtBase, Large, and XLarge models were identified achieving 97.5% accuracy in the first stage. Afterwards, some selection criteria were followed for the models identified in the first stage for use in ensemble learning technique within the optimization approach. The top-performing ensemble model, EM3 (composed of RegNetX160, ResNetRS101, and ResNet101), attains an AUC of 0.9971 in the second stage. Evaluation on unseen data ensures model robustness and enhances the study's overall validity and reliability. The design and implementation of the study have been optimized to address the limitations identified in the literature. This approach offers a rapid and highly accurate decision support system for timely MPox diagnosis, reducing human error, manual processes, and enhancing clinic efficiency. It aids in early MPox detection, addresses diverse disease challenges, and informs imaging device software development. The study's broad implications support global health efforts and showcase artificial intelligence potential in medical informatics for disease identification and diagnosis.

Keywords Monkeypox · Chickenpox · Measles · Transfer learning · Ensemble learning · Optimization

Introduction

Monkeypox (MPox) is an infectious disease caused by the monkeypox virus, which manifests as a painful rash, enlarged lymph nodes, and fever. Human transmission of MPox can occur through physical contact with an infected individual, contact with contaminated materials, or exposure to infected animals [1]. Historically, MPox outbreaks have occurred in 1970 (Central Africa, East Africa, and West Africa), 2003 (USA), and 2017 (Nigeria). However, in May 2022, a sudden outbreak of MPox emerged, rapidly spreading across Europe, the Americas, and all six World Health Organization (WHO) regions. A total of 110 countries reported approximately 87,000 cases and 112 deaths. The global MPox outbreak was subsequently declared a public health emergency of international concern, Public Health Emergency of International Concern (PHEIC), on July 23, 2022. In response, the WHO published a strategic preparedness and response plan for MPox along with a suite of technical guidance documents [1].

Accurate identification of MPox can be challenging due to its resemblance to other infections and conditions. It is crucial to differentiate MPox from conditions such as chickenpox (CPox), measles, bacterial skin infections, scabies, herpes, syphilis, and medication-associated allergies. It is also possible for individuals with MPox to have concurrent herpes infections, while children suspected of having MPox may actually have CPox. Therefore, timely testing is essential to ensure prompt treatment and prevent further transmission [1, 2]. Chickenpox, caused by the varicella-zoster virus, is a highly contagious disease characterized by an itchy, blister-like rash and other associated symptoms.

✉ Serkan Savaş
serkansavas@kku.edu.tr

[1] Department of Computer Engineering, Kırıkkale University, 71450 Kırıkkale, Turkey

The rash typically starts on the chest, back, and face before spreading to the entire body. Chickenpox can pose serious risks, particularly during pregnancy, in infants, adolescents, adults, and individuals with weakened immune systems [3]. Measles is an easily transmissible infection that can cause severe complications in certain individuals. It typically begins with cold-like symptoms and is followed by a rash a few days later. Some individuals may also develop small spots in their mouth. Initial symptoms of measles include high fever, runny or blocked nose, sneezing, cough, and red, sore, watery eyes [4].

Distinguishing MPox from other diseases, such as measles and CPox, poses challenges as their primary differentiating factor lies in the inflammation and rash on the body. These symptoms are difficult to detect visually, except through the polymerase chain reaction (PCR) test. Prompt and accurate diagnosis of MPox is challenging for healthcare professionals, further complicated by the scarcity of available PCR tests to detect the MPox virus [2]. PCR analysis involves a series of steps for deoxyribonucleic acid (DNA) synthesis. Initially, the template DNA is denatured, resulting in the separation of the double-stranded DNA into single strands. Subsequently, primers are annealed to each original strand, facilitating the synthesis of new DNA strands. These reactions can be performed with any DNA polymerase and lead to the production of specific segments of the original DNA sequence. However, to achieve multiple rounds of synthesis, the templates need to undergo denaturation again, necessitating high temperatures that can deactivate most enzymes. To overcome this challenge, initial attempts at cyclic DNA synthesis involved the addition of fresh polymerase after each denaturation step (first and second) [5]. Nonetheless, this process is time-consuming and susceptible to factors that can compromise the reliability of the analysis. Therefore, the investigation of false negative rates and sensitivity rates of PCR tests is crucial, and repeat testing is recommended for follow-up clinical evaluation [6, 7]. Additionally, the complexity of the target species and the similarity between different illnesses, along with the expertise of the performing clinics, can influence the results of clinical analysis. Furthermore, human factors such as fatigue or monotony resulting from repetitive tasks can also impact the quality of the analyses. Consequently, there is a strong desire to automate the analysis process to overcome these quality limitations associated with human factors and optimize the overall efficiency of the procedure. By automating the analysis task, it would be possible to mitigate these quality disadvantages while also streamlining the entire process in terms of time optimization.

Given this situation, how can we differentiate these visually similar diseases more easily using state-of-the-art techniques? In recent years, machine learning (ML) and deep learning (DL), disciplines that have successfully provided solutions to various problems, emerge as promising approaches. Solutions to these problems can vary from segmentation and region of interest studies for smart city applications [8] to biomedical acoustic analysis to remove communication barriers [9], and various machine learning studies. Medical image processing has been widely utilized for the past decade, particularly due to the success of deep neural networks (DNNs) in this field. Examples of DL usage in medical informatics include skin lesion detection [10], pneumonia detection [11], Covid-19 diagnosis [12, 13], skin cancer identification [14], automatic diagnosis of malaria parasites detection [15], and stroke classification [16]. Detailed information regarding the DL techniques employed in this research is provided in the methodology section (see "Material and Method"). The subsequent sections of the research are structured as follows: The second section provides a summary of the related literature. The third section describes the materials and methodology employed in the study. The fourth section presents the experimental results. The fifth section includes a comparison and discussion of the study's findings with previous works. Finally, the sixth section concludes the study.

In this context, although the identification of this health problem was first reported in 1958, it has become more prevalent over the past 20 years and evolved into a global epidemic in 2022. This study proposes a promising approach to address this health issue. The contributions of the proposed approach are as follows:

- Comprehensive analysis of pre-trained DNNs on monkeypox, chickenpox, and measles dataset
- An innovative solution proposal to a globally threatening disease
- A fast and highly accurate decision support solution to assist clinics
- A two-stage optimization methodology using deep transfer learning and ensemble learning
- Prevention of overfitting through data augmentation techniques

In addition to these contributions of the study, there are some other indirect contributions that can be specified. By automating the analysis task using DL techniques, the study mitigates the limitations associated with human factors and optimizes the overall efficiency of the procedure. The findings contribute to early diagnosis and prevention of MPox transmission, while also addressing the challenges posed by disease diversity, manual processes, human errors, and increased workload. Moreover, the study serves as a basis for future research on MPox and related diseases, offering valuable insights for software developers of imaging devices used in clinical processes. The comprehensive tests conducted in this study have significant implications

for global health efforts, particularly during outbreaks and epidemics. The accurate differentiation of visually similar diseases can aid healthcare professionals in making timely and informed decisions, ultimately improving patient outcomes. The proposed DL approach showcases the potential of artificial intelligence (AI) in medical informatics, providing a foundation for further advancements in disease identification and diagnosis.

Literature Review

The new outbreak (MPox) once again highlights the proximity of global health threats and reinforces the importance of developing mechanisms to address these threats. AI technologies, along with the contributions of ML and DL studies over the past 20 years, have been successful in various disciplines, with health being one of the prominent areas.

There are numerous opportunities for AI in clinical applications, which have the potential to significantly improve patient care, streamline healthcare processes, and advance medical research. Diagnostic assistance and early disease detection within clinical decision support systems, clinical trials optimization, healthcare workflow optimization, personalized treatment plans, quality assurance and compliance, telemedicine and remote monitoring, genomic medicine, population health management, resource allocation, and data analytics and research can be listed in key opportunities. AI can enhance diagnostic accuracy by analyzing medical images such as X-rays, MRIs, and CT scans, as well as pathology slides and diagnostic data. AI algorithms excel in detecting subtle abnormalities and complex patterns that may challenge human interpretation. Moreover, AI-powered predictive models leverage patient data to identify early disease indicators, enabling timely interventions and improved patient outcomes. Additionally, AI systems provide healthcare professionals with evidence-based recommendations, supporting informed decisions regarding treatment options, drug interactions, and patient care plans [17]. AI's impact extends to clinical trial management, where it efficiently identifies suitable trial participants, predicts trial outcomes, and optimizes recruitment processes. This contributes to accelerated drug development and more efficient healthcare resource allocation, including staffing, appointment scheduling, and insurance claims processing. Furthermore, AI facilitates personalized medicine by analyzing patients' medical histories, genetic information, and real-time data to tailor treatment plans and minimize side effects. AI's data analysis prowess assists in identifying population health trends, risk factors, and opportunities for preventive care, thereby improving public health interventions. Quality control systems driven by AI ensure healthcare standards and regulations are upheld, reducing errors, and enhancing patient safety [18].

These opportunities showcase the transformative potential of AI in clinical applications, from improving patient outcomes to driving healthcare innovation and efficiency. Realizing these benefits requires collaboration among healthcare providers, AI developers, regulators, and other stakeholders to ensure responsible and ethical implementation. However, there are several challenges, concerns, or issues related to the integration of AI in clinical settings and healthcare. These challenges, concerns, or issues include data quality and availability, scalability, clinical integration, interoperability, privacy and security, ethical dilemmas, bias and fairness, regulatory compliance, clinical validation, human-AI collaboration, cost and resource constraints, patient acceptance and trust, education and training, and long-term maintenance and updates.

Specifically for the classification problem of MPox, CPox, and measles, there are issues such as data availability and quality, scalability, clinical integration and validation, and interoperability. Rapid and early diagnosis of diseases such as MPox and other similar skin lesions is crucial. Therefore, studies focusing specifically on MPox have begun in response to the outbreak. Different studies in the literature which use AI techniques for this task are summarized in Table 1, including the models used, datasets, objectives, and their results. The limitations of the studies are also noted in the table.

Within the studies presented in Table 1, there are studies [19, 21, 30] that specifically perform classification and detection processes for MPox. These studies generally yield high accuracy rates. However, when other diseases with similar skin symptoms (CPox, measles, etc.) are included, the performance rates are negatively affected. When only MPox positive–negative studies are conducted on the datasets found in the literature, results reaching up to 100% accuracy can be achieved. However, this rate remains more of a classification study in the literature rather than being beneficial in clinical diagnosis processes. This is because the most significant problem in this disease group is the rapid and early diagnosis of which group the skin lesions belong to. In some of the studies presented in Table 1, despite the comparison of different numbers of pre-trained models [20, 22, 24, 27], lower performance results were obtained compared to the results obtained in this study. Similarly, there are studies that apply ensemble techniques but achieve lower performance [29] or do not include a stage for selecting pre-trained transfer models [26] in the literature. Other studies have been conducted that only present training and validation rates [23] or focus on mobile applications [28].

When considering recent studies on MPox, it can be generally observed that despite the findings of this

Table 1 Previous studies on MPox

Study	Model	Dataset	Aim	Result	Disadvantages
[19]	VGG16	Monkeypox2022	Detection MPox positive or negative	Acc: 97%	Detects only MPox not classification of others
[20]	DenseNet-201	MSID (All)	Classification MPox, CPox, measles and normal images	Acc: 93.19% (original) and 98.91% (augmented)	Usage of only a single pre-trained model and lower accuracy rate on original data
[21]	12 different pre-trained model	MSID (only MPox and normal)	Classification of MPox or normal images	Acc: 98.25%, Sens: 96.55%, Spe: 100.00%, F1-Score: 98.25%	Classification of only MPox not others
[22]	9 different pre-trained model	Monkeypox2022 + Kaggle MPox dataset	Classification of MPox vs normal and MPox vs others	Acc: between 93.6 and 96.4% for MPox vs normal, Acc: between 69 and 72% for multiclass	Lower accuracy rates on multi class problems
[23]	5 different pre-trained model	MPox Skin Lesion Dataset (MSLD)	Classification of monkeypox vs others	Acc: 83.89% (val) for Xception-CBAM-Dense	Result is provided for validation accuracy and lower accuracy rate
[24]	ResNet50, EfficientNetB3, and EfficientNetB7	Kaggle MPox dataset	Detection of MPox skin lesions	Acc: 87%, pre: 92%, rec: 87%, and F1-Score: 90% for EfficientNetB3	Only three different pre-trained model, one step approach and lower accuracy rates
[25]	Proposed CNN model	MPox skin lesion dataset + CPox Stock photos	Classification of MPox and CPox	Acc: 99.60%	Only classification of two different classes. Overfitting possibility
[26]	Ensemble learning with Inception V3, Xception, and DenseNet169	Kaggle MPox dataset	Classification of monkeypox vs others	Acc: 93.39%, pre:88.91%, rec: 96.78%, F1-Score: 92.35%	No definition step for defined pre-trained algorithm
[27]	MobileNetV2, VGG16, and VGG19	MSID dataset	Classification MPox, CPox, measles, and normal images	Acc: 91.38%, pre: 90.5%, rec: 86.75%, F1-scor: 88.25% for MobileNetV2	Using only transfer learning and also only author defined models
[28]	Six pre-trained model	Kaggle Monkeypox Skin Lesion Dataset (MSLD)	Android mobile app for MPox detection	Acc: 91.11%	Focus on mobile app. Therefore, not much attention paid on backend for model selection and training
[29]	9 different pre-trained model + ensemble technique with top-2 (Xception and DenseNet169)	Monkeypox2022	Detection MPox, CPox, measles and normal images	Acc: 87.13%, pre: 85.44%, rec: 85.47%, F1-score: 85.40%	Lower accuracy rates
[30]	MobileNetV3-s, EfficientNetV2, ResNet50, VGG19, DenseNet121, and Xception	Roboflow MonkeyPox Skin Dataset + Kaggle dataset	Detection MPox Positive or Negative	For MobileNetV3-s (best), F1-score:98%, AUC: 0.99, acc: 96%, Rec: 97%	Detects only MPox not classification of others

research, there are several limitations that can be generalized as follows:

- Some studies only address the classification and detection processes specifically for MPox. These studies may yield high accuracy rates; however, when similar skin symptoms of other diseases (CPox, measles, etc.) are included, the performance rates may be negatively affected.
- Several studies employ different numbers of pre-trained models, resulting in lower accuracy rates. Some studies solely rely on pre-trained models determined by the authors' preference. Without applying any selection criteria in these approaches, the correctness of the chosen models cannot be determined.
- In some studies, the reported results may appear high as they only reflect the performance during the training process. While the models may exhibit high performance in this context, they may not perform equally well on unseen data. Therefore, it is more reliable to evaluate the results using test data that was not used during training.

The design and implementation of this study have been optimized to address these limitations identified in the literature. Therefore, the following objectives were set in the study.

- To detect diseases with similar skin symptoms with high accuracy
- To reduce human error in diagnosis and diagnostic procedures and at the same time contribute to the reduction of workload
- Using multiple state-of-the-art (SOTA) techniques (transfer learning, fine-tuning, ensemble learning) in combination and optimization to ensure reliability and validity of the results
- Using SOTA techniques (data augmentation) to overcome data limitations
- Performing tests on unseen data to ensure the reliability of the results

Thus, studies have been carried out in this direction. To this end, the classification of different diseases with similar skin symptoms has been conducted. A stage has been planned to determine the preferred models, and through performed tests, the best models have been identified. An optimization process has been applied by combining these models as an ensemble. Additionally, test results have been evaluated using unseen data that was not utilized during training to prevent overfitting of the model. The evaluations of the studies are presented in the discussion section (see "Discussion").

Material and Method

This section provides information about the dataset and techniques used in the study. The block diagram of the research is given in Fig. 1.

The tests conducted in the study were performed on hardware consisting of a MacBook M2 Pro with 16 GB RAM, a 16-core GPU, and a 10-core CPU, with a 512 SSD. The experiments were carried out using the Anaconda platform and the JupyterLab environment. Python programming language version 3.8.16 was used in the study. In addition, version 2.12.0 of the TensorFlow library was included in the program. Version 3.7.1 of the Matplotlib library and version 1.2.2 of the Sklearn library were included and used. The deepstack module for ensemble learning was also used in the study, and version 0.0.9 of this library was included in the program.

Dataset

One of the major limitations for MPox disease is the dataset. Recent studies have been conducted on a limited number of specific datasets. One of these datasets is the Monkeypox Skin Images Dataset (MSID), which is openly available on Mendeley. This dataset consists of four classes: monkeypox, chickenpox, measles, and normal. All image classes are collected from internet-based health websites. The dataset was developed by two students, Diponkor Bala and Md. Shamim Hossain, from the Department of Computer Science and Engineering, Islamic University, Kushtia, Bangladesh, and the School of Computer Science and Technology, University of Science and Technology of China (USTC), Hefei, Anhui, China, respectively [31].

The dataset [31] contains 279 monkeypox images, 107 chickenpox images, 91 measles images, and 293 normal images. In the study, various operations were performed on these images. The first operation involved separating 30 images from each class for testing purposes (unseen data) to address one of the limitations mentioned in the literature review section (see "Literature Review") regarding the presentation of test data results. Since DL models require a large number of images for training, data augmentation techniques were applied to this dataset. Examples of images from the dataset are presented in Fig. 2.

Data Augmentation

Data augmentation is a vital technique in ML and DL models, enhancing performance by generating new samples through transformations applied to existing datasets. It aids generalization and combats overfitting by expanding dataset

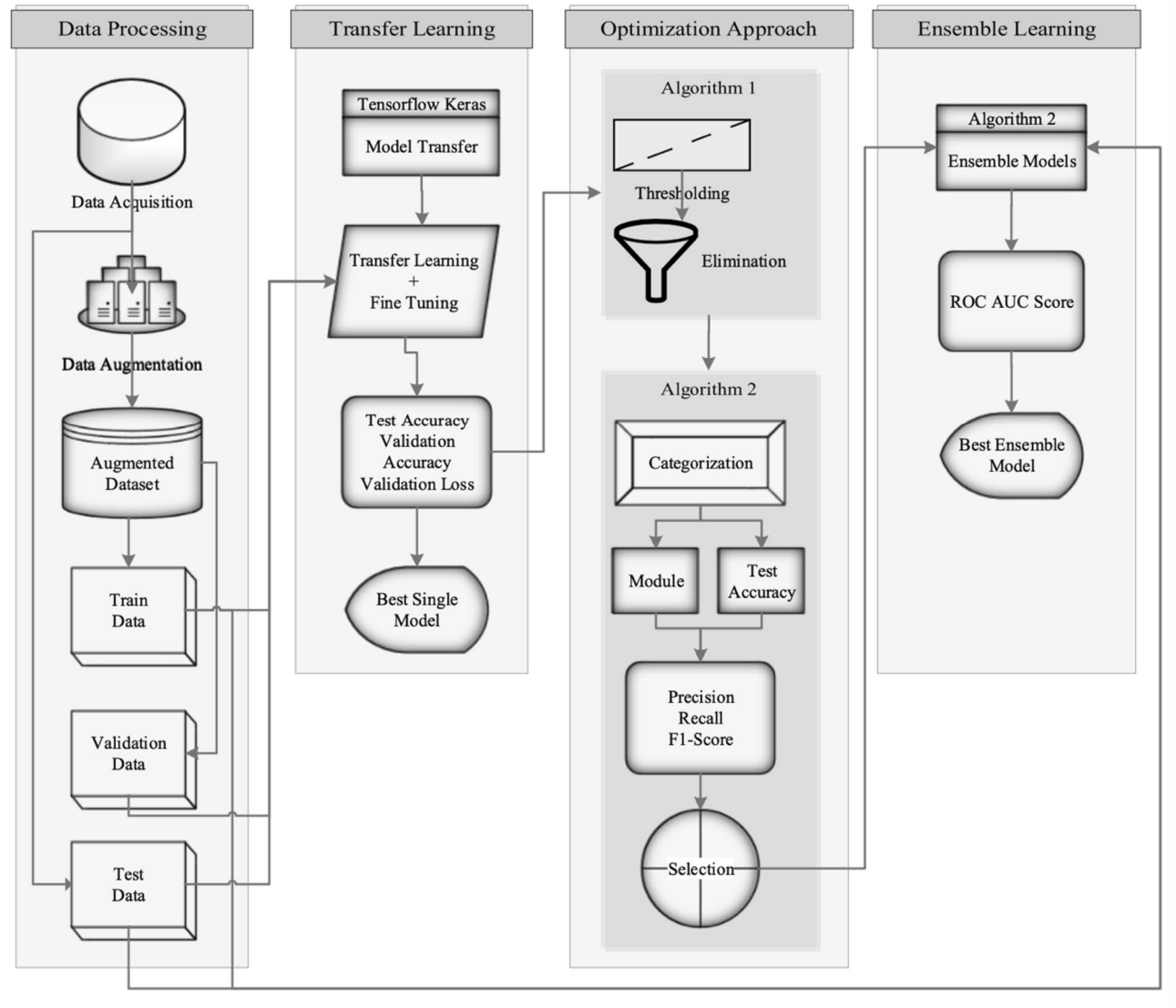


Fig. 1 Block diagram of the study

size. For image classification, it includes rotation, flipping, cropping, zooming, etc. This technique is especially effective with limited data and imbalanced classifications, where small datasets hinder generalization. Thus, data augmentation is a critical tool in ML and DL for improving performance, generalization, and handling limited data or imbalanced datasets [32, 33].

In this study, data augmentation technique was also used to overcome the limitation of limited data. The images in the dataset were augmented by applying rotation. To reduce the imbalance in the dataset, the images in classes such as MPox and normal which have higher counts were only rotated by 45°. The images in classes such as CPox and measles which have lower counts were not only rotated by

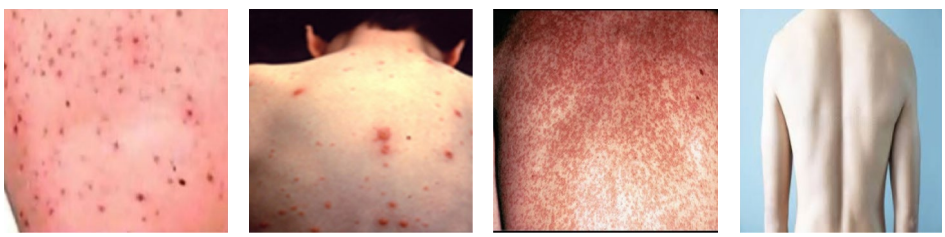


Fig. 2 Sample images from dataset **a** MPox, **b** CPox, **c** measles, and **d** normal

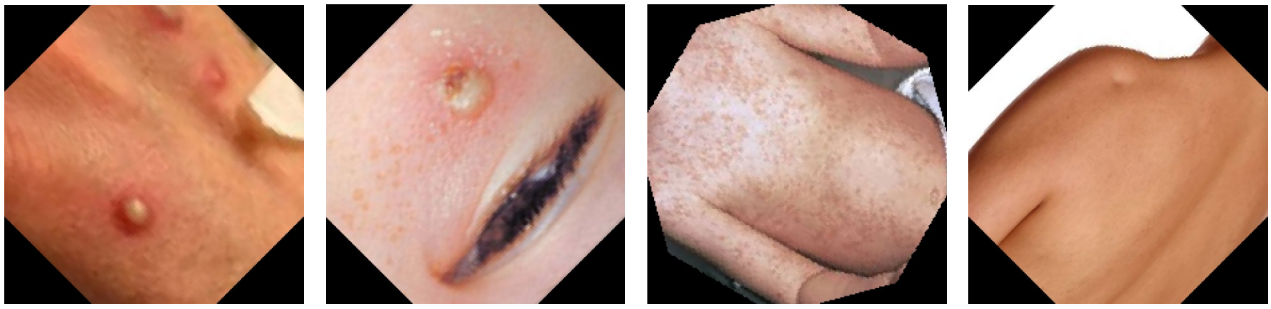


Fig. 3 Sample images from augmented dataset **a** MPox, **b** CPox, **c** measles, and **d** normal

45° but were further rotated by an additional 15°, resulting in a new augmented dataset. Examples of the augmented images after the data augmentation process are presented in Fig. 3. The resulting table after these operations is as follows (Table 2).

Transfer Learning

Transfer learning, widely used in DL research, involves leveraging knowledge and representations from one task to enhance performance on another. This method efficiently transfers data and computational resources, proving invaluable in data-limited or resource-constrained scenarios. Typically, it involves fine-tuning pre-trained DL models on a large-scale dataset for a new task, improving performance, generalization, and reducing training time and resources. Transfer learning finds success in domains like image processing, natural language processing, and speech recognition, effectively applying pre-trained models to new tasks and enabling rapid learning by starting with general representations that can be fine-tuned for task-specific features [34, 35].

Considering the contributions of transfer learning, the study also employed the approach of transfer learning with fine-tuning. Instead of selecting a single or a specific number of pre-trained models, as indicated in the literature review section (see "Literature Review"), a total of 71 different pre-trained models (which had successful results in ImageNet competition and also applied many different areas with transfer learning and fine-tuning techniques) available in the TensorFlow Keras library [36, 37] were initially tested. The list of models and their modules given in a table in the supplementary file. The most successful models were then used in an ensemble learning technique to further optimize the performance. As seen in the table in the supplementary file, a comprehensive training and testing process was carried out with all the pre-trained models used in different studies in the literature. Thus, it is aimed to form a basis for further research. In the conclusion part of the study, this issue related to the recommendations was also stated (see "Conclusion").

Ensemble Learning

Ensemble learning, commonly used in both ML and DL, combines multiple models to enhance predictive performance and robustness. It capitalizes on the idea that diverse models can yield better predictions than a single one. In DL, ensemble methods improve performance by combining predictions from multiple neural networks. Model averaging averages predictions from independently trained networks, while model stacking uses network outputs as input features for a meta-learner. Ensemble learning boosts model generalization, mitigates overfitting, and improves accuracy in domains like computer vision, natural language processing, and speech recognition. It excels when individual models exhibit diverse strengths and weaknesses, capturing a wider range of patterns and enhancing resilience to data noise or

Table 2 Dataset information

Class	Number of original data	Number of augmented data	Number of train data	Number of validation data	Number of test data
MPox	279	2202	1982	220	30
CPox	107	1575	1418	157	30
Measles	91	1335	1202	133	30
Normal	293	2216	1995	221	30

outliers. However, it may require additional computational resources, necessitating careful method selection based on task, dataset, and available resources [38–40].

In this study, to overcome the aforementioned limitations mentioned in the literature review section, and to optimize the performance of the models, the ensemble learning technique was employed. The DeepStack module was utilized for implementation which is a Python module designed for constructing deep learning ensembles, initially built on the Keras framework, and distributed under the MIT license. Within this module, the Dirichlet Ensemble technique was applied. This technique involves weighting the ensemble members to optimize a specific metric or score based on a validation dataset. The optimization of ensemble weights is performed using a randomized search method based on the dirichlet distribution. The Dirichlet distribution, a significant multivariate continuous distribution in probability and statistics, serves as a natural extension of the Beta distribution. It plays a crucial role in modeling compositional data and proportion measurements [41]. The Dirichlet distribution of order k has k parameters $\alpha = (\alpha_1, \ldots, \alpha_k)$, with $\alpha_i > 0$. It is denoted by $Dir(\alpha)$ and is given as Eq. (1):

$$f(x) \equiv f(x|\alpha) = \frac{1}{B(\alpha)} \prod_{i=1}^{k} x_i^{\alpha_i - 1} \tag{1}$$

where for $i = 1, 2, ..., k, x_i \geq 0$ and $\sum_{i=1}^{k} x_i = 1$. Here, the normalizing constant $B(\alpha)$ is the multivariate beta function, which can be written in terms of the gamma function as Eq. (2):

$$B(\alpha) = \alpha \frac{\prod_{i=1}^{k} \Gamma(\alpha_i)}{\Gamma(\sum_{i=1}^{k} \alpha_i)} \tag{2}$$

The support or domain of the Dirichlet distribution is the set of k-dimensional vectors x whose components are real numbers in the interval [0, 1] such that $\sum_{i=1}^{k} x_i = 1$.

During the fitting process of the dirichlet ensemble model, the module calculates the ensemble weights by optimizing the AUC Binary Classification Metric. This optimization is achieved through randomized search utilizing the properties of the dirichlet distribution [42].

Optimization Approach

Fine-tuning in DL involves adapting a pre-trained model, usually trained on a large dataset, to a specific task using a smaller labeled dataset. This process capitalizes on the pre-trained model's knowledge and learned representations, enhancing them for the target task. It entails freezing lower layers that capture generic features and retraining upper layers with task-specific data through backpropagation. Fine-tuning is valuable when labeled data for the target task is limited, allowing the model to transfer knowledge from pre-training to improve performance with fewer samples. It is widely used in domains like computer vision, natural language processing, and speech recognition, enabling adaptation of powerful pre-trained models to specific applications with reduced data requirements [43, 44].

In this study, to take advantage of the benefits of fine-tuning, a test-and-retest method was initially employed to determine the standard parameters for the models while also identifying the most suitable parameters for the task at hand. Several considerations played a role in determining these parameters, including insights from the literature review, the researcher's past experiences, the selection of appropriate parameters for the models, the results obtained from testing, and innovative strategies. Taking these components into account, in the initial stage of the study, pre-trained models were used, and the following adjustments were made to the fully connected layers (classification) after the feature extraction layers in the CNN structure.

In this structure, the neural network architecture employed three dense layers, namely, with 1000, 512, and 256 neurons, respectively. The rectified linear unit (ReLU) activation function was utilized in these layers to introduce non-linearity. The ReLU function is calculated as Eq. (3):

$$f(x) = \begin{cases} 0 for x < 0 \\ x for x \geq 0 \end{cases} \tag{3}$$

To mitigate overfitting, three dropout layers with a rate of 0.5 were inserted between these dense layers. The dropout technique is a versatile approach that has demonstrated improved performance in neural networks across various application domains. Feed-forward NN with dropout is described as Eq. (4) [45]:

$$r_j^{(l)} \sim Bernoulli(p)$$

$$\tilde{y}^{(l)} = r^{(l)} * y^{(l)}$$

$$z_i^{(l+1)} = w_i^{(l+1)} \tilde{y}^l + b_i^{(l+1)}$$

$$y_i^{(l+1)} = f(z_i^{(l+1)}) \tag{4}$$

where $l \in (1, ..., L-1)$ index the hidden layers of the network, i hidden unit, $z^{(l)}$ denotes the vector of inputs into layer l, $y^{(l)}$ denotes the vector of outputs from layer, $W^{(l)}$ and $b^{(l)}$ are the weights and biases at layer l, and f is any activation function; $*$ denotes an element-wise product, $r^{(l)}$ is a vector of independent Bernoulli random variables each of which has probability p.

The final layer, consisting of four neurons, employed the Softmax activation function, enabling the classification of multiple classes. The Softmax function: $\sigma : \mathbb{R}^K \rightarrow [0,1]^K$ is calculated as Eq. (5) [46]:

$$\sigma(z)_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}} for\ i = 1, \ldots, K \text{ and } z = (z_1, \ldots, z_K) \epsilon \mathbb{R}^K \tag{5}$$

During model compilation, the Adam optimizer was utilized, and a learning rate of 1×10^{-4} was specified. The Adam optimizer converges much faster for multi-layer NNs or CNNs, than any other optimizer and is nowadays one of the most popular step size methods in the studies. It combines the benefits of two other optimization algorithms (Momentum and RMSProp). For optimization process, $n_t := \frac{n}{\sqrt{t}}$ as step size $\beta_1, \beta_2 \in (0,1)$ as decay rates for the moment estimates, $\beta_{1,t} := \beta_1 \lambda^{t-1}$ with $\lambda \in (0,1), \epsilon > 0, e(w(t))$ as a convex differentiable error function, and $w(0)$ as the initial weight vector. During the process, set $m_0 = 0$ as initial first moment vector, set $m_0 = 0$ as initial second moment vector, and set $t = 0$ as initial time stamp. Here, while $w(t)$ not converged, optimizer calculates as Eq. (6) and returns $w(t)$ [47, 48]:

$$t = t + 1$$

$$g_t = \nabla_w e(w(t-1))$$

$$m_t = \beta_{1,t} m_{t-1} + (1 - \beta_{1,t}) g_t$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2$$

$$\hat{m}_t = \frac{m_t}{(1 - \beta_1^t)}$$

$$\hat{v}_t = \frac{v_t}{(1 - \beta_2^t)}$$

$$\mathrm{w(t)} = \mathrm{w(t-1)} - n_t \frac{\hat{m}_t}{(\sqrt{\hat{v}_t} + \epsilon)} \tag{6}$$

The loss value is a measure that quantifies the discrepancy between the predicted outputs of a model and the true or expected outputs. It serves as a guide for the model to adjust its internal parameters during the training process to minimize the error and improve the overall performance [49]. By optimizing the loss function during the training process, the model can learn to make more accurate predictions and improve its overall performance on the given task. The choice of loss function in this study was categorical_crossentropy, which is suitable for multi-class classification tasks. Mathematically, the categorical cross-entropy loss for a single data point is defined as Eq. (7) [50]:

$$L(y, \hat{y}) = -\sum_{i=1}^{C} y_i \cdot \log(\hat{y}_i) \tag{7}$$

where $L(y, \hat{y})$ is the loss for a single data point, y_i is the ground truth probability that the data point belongs to class i, $\hat{y}_i$ is the predicted probability that the data point belongs to class i, and C is the total number of classes. In practice, y is a one-hot encoded vector, where only one element is 1 (indicating the true class), and all other elements are 0. The predicted probability distribution $\hat{y}$ is typically obtained from the softmax function applied to the raw scores or logits produced by the model. The softmax function ensures that $\hat{y}$ sums to 1, turning the raw scores into class probabilities [50].

The batch size, defined as 8, was employed during the training process, and each test was conducted for a total of 30 epochs to optimize the model's performance and convergence.

Training process in first stage demonstrated that some models have achieved significantly lower performance compared to other pre-trained models. Therefore, a process of elimination was applied by setting certain threshold values. The steps followed for elimination are as follows:

1. Models with a test accuracy below 80% were removed.
2. Models with a validation loss above 2 were removed.
3. Models with a validation accuracy of 75% or below were removed.

One important limitation identified during the literature review is the evaluation of the performance of pre-trained models directly (see "Literature Review"). However, the optimization studies that bring together successful results have been conducted in a limited number. In these studies, only a specific number of models were combined. To overcome all these limitations, an ensemble learning study consisting of combinations of different models was conducted in this research. For this process, a two-step approach was applied. In the first step, the performances of all pre-trained models were compared. In the second step, successful models were combined with different combinations, and an optimization approach was applied. The following selection criteria (SC) were followed for the models identified for use in ensemble learning technique in the optimization approach. To increase model diversity during the selection process, a subsequent model that meets the criteria was chosen instead of the models selected in previous criteria.

- SC1: Three models with the highest test accuracy from different module groups were selected (in case of a tie, the lowest loss value was considered).

- SC2: For each category based on test accuracy, three models with the lowest validation loss rate from different module groups were included.
- SC3: Three models with the highest F1-Score for the MPox class were selected from different module groups (in case of a tie, higher test accuracy was considered).
- SC4: Three models with the highest F1-Score based on the F1-Score ratio for the CPox class were selected from different module groups (in case of a tie, higher test accuracy was considered).
- SC5: Three models with the highest F1-Score for the measles class were selected from different module groups (in case of a tie, higher test accuracy was considered).
- SC6: Three models with the highest F1-Score for the normal class were selected from different module groups (in case of a tie, higher test accuracy was considered).

The algorithm for the elimination steps is presented in algorithm 1, and the selection criteria and the algorithm applied for the selected models are shown in algorithm 2. Algorithms are provided in supplementary file.

Evaluation Metrics

The proper selection and use of evaluation metrics play a vital role in assessing the performance of a study. One of the limitations identified during the literature review (see "Literature Review") is the use of only accuracy as the evaluation metric in some studies. However, especially in healthcare research, in addition to accuracy, metrics such as precision, recall (sensitivity), and f1-score are of great importance. In addition, AUC values were obtained in this study, and the performance of the study was evaluated based on these metrics. The AUC is a scalar metric derived from the ROC curve. It quantifies the overall performance of a model. During the performance evaluation of the models, true positive (TP), true negative (TN), false positive (FP), and false negative (FN) values for each class were used by calculating their ratios to each other.

The accuracy rate represents the ratio of correct predictions to all examples, and it is calculated as shown in Eq. (8). Precision is used to measure the accuracy of positive predictions and quantifies the proportion of correctly predicted positive instances out of all instances predicted as positive (Eq. 9). Recall represents the model's ability to avoid false negatives and capture all positive instances (Eq. 10). The F1-score combines precision and recall into a single value and is calculated by Eq. (11):

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{8}$$

$$Precision = \frac{TP}{TP + FP} \tag{9}$$

$$Recall\ (Sensitivity) = \frac{TP}{TP + FN} \tag{10}$$

$$F1 - score = 2x\frac{Precision \times Recall}{Precision + Recall} \tag{11}$$

Results

In this section, the comprehensive analysis results obtained in the study are presented, and the classification outcomes achieved through the optimization approach are described.

Results of the Initial Evaluation

To mitigate the potential misleading nature of training accuracy values, as noted in the literature, the validation accuracy, validation loss, and test accuracy values of the 71 models used in the study were recorded. The obtained results are presented in Table 3. The models are listed in the following order: first, based on the highest test accuracy; second, based on the highest validation accuracy; and finally, based on the lowest loss value.

As mentioned in the "Material and Method" section, it is not sufficient to rely solely on accuracy rates to evaluate the performance of the models. Therefore, various metrics and categorizations have been utilized to assess the performance of these models. In the initial training stage, Table 3 shows that certain models exhibited significantly inferior performance relative to others. Consequently, a process of elimination ensued, guided by the establishment of specific threshold criteria. The elimination process resulted in 40 pre-trained models successfully passing the threshold criteria (see Algorithm 1 in supplementary file). Following the elimination, the accuracy graph of the remaining models is presented in Fig. 4a, while the distribution of the loss values is shown in Fig. 4b.

Figure 4 demonstrates that specific groups of models exhibit notable accuracy performance on the disease images used in the study. Among these model groups, the "convnext," "regnet_rs," and "efficientnet_v2" groups are the most represented. Pre-trained models consist of hundreds of layers. After performing feature extraction within these layers, the models have been fine-tuned for the classification problem used in the study (see "Material and Method").

Table 3 Accuracy rates of pre-trained models

Model	Module	Train acc.	Val. acc.	Test acc.	Model	Module	Train acc.	Val. acc.	Test acc.
ConvNeXtXLarge	convnext	99.12	85.77	97.50	**RegNetY064**	regnet	98.54	76.61	86.67
ConvNeXtLarge	convnext	99.24	84.68	97.50	**MobileNet**	mobilenet	99.24	76.33	86.67
ConvNeXtBase	convnext	99.18	83.86	97.50	**RegNetY008**	regnet	98.92	75.37	86.67
EfficientNetV2S	efficientnet_v2	98.85	84.54	95.83	**EfficientNetB5**	efficientnet	97.14	72.09	86.67
ResNet152	resnet	98.76	80.57	95.83	**ResNet50**	resnet	98.45	77.02	85.83
ResNetRS200	resnet_rs	99.18	84.95	95.00	**InceptionV3**	inception_v3	95.48	76.61	85.83
ResNetRS420	resnet_rs	98.76	83.58	95.00	**NASNetLarge**	nasnet	98.45	71.55	85.83
ResNetRS101	resnet_rs	99.39	83.31	95.00	**DenseNet169**	densenet	98.66	75.38	85.00
RegNetY160	regnet	98.61	82.63	95.00	**MobileNetV2**	mobilenet_v2	98.92	68.40	85.00
ResNetRS270	resnet_rs	99.20	87.27	94.17	**EfficientNetB0**	efficientnet	98.91	77.56	84.17
EfficientNetV2M	efficientnet_v2	98.76	86.18	94.17	**EfficientNetV2B3**	efficientnet_v2	98.58	73.60	84.17
RegNetX160	regnet	98.92	80.57	94.17	**RegNetY006**	regnet	98.03	72.78	84.17
RegNetX120	regnet	98.67	78.93	94.17	**EfficientNetV2B0**	efficientnet_v2	98.80	70.45	84.17
EfficientNetV2L	efficientnet_v2	98.60	85.77	93.33	**ResNet50V2**	resnet_v2	99.01	69.08	84.17
ResNetRS50	resnet_rs	99.44	80.85	93.33	**RegNetX032**	regnet	97.86	76.61	83.33
RegNetX064	regnet	98.64	81.40	92.50	**EfficientNetV2B2**	efficientnet_v2	98.77	75.51	83.33
RegNetX040	regnet	98.59	81.26	92.50	**EfficientNetB6**	efficientnet	97.21	75.10	83.33
ResNetRS350	resnet_rs	98.83	87.82	91.67	**RegNetX006**	regnet	97.62	73.46	83.33
ResNetRS152	resnet_rs	98.58	83.72	91.67	**Xception**	xception	97.82	72.50	83.33
RegNetX016	regnet	98.80	80.98	91.67	**DenseNet121**	densenet	98.44	72.23	83.33
RegNetY320	regnet	98.73	82.76	90.83	**EfficientNetV2B1**	efficientnet_v2	99.08	72.23	83.33
DenseNet201	densenet	98.33	82.22	90.83	**RegNetY032**	regnet	98.51	69.77	83.33
ResNet101	resnet	98.89	81.67	90.00	**VGG19**	vgg19	98.68	75.78	82.50
RegNetY080	regnet	98.67	78.52	90.00	**RegNetX002**	regnet	97.33	74.69	82.50
EfficientNetB3	efficientnet	98.53	77.98	89.17	**ConvNeXtSmall**	convnext	99.26	74.97	81.67
EfficientNetB2	efficientnet	98.64	74.69	89.17	**VGG16**	vgg16	98.67	74.56	81.67
RegNetX320	regnet	98.56	79.34	88.33	**RegNetY004**	regnet	99.50	73.73	81.67
RegNetY040	regnet	98.42	78.11	88.33	**RegNetX004**	regnet	97.41	67.17	81.67
EfficientNetB1	efficientnet	98.86	79.62	88.00	**RegNetY120**	regnet	98.92	74.56	80.83
RegNetY016	regnet	98.36	79.07	87.50	**InceptionResNetV2**	inception_resnet_v2	96.56	70.86	79.17
ResNet152V2	resnet_v2	99.14	78.11	87.50	**RegNetY002**	regnet	99.24	67.58	79.17
RegNetX080	regnet	98.80	81.53	86.67	**ConvNeXtTiny**	convnext	99.35	67.85	76.67
EfficientNetB7	efficientnet	98.56	81.12	86.67	**NASNetMobile**	nasnet	97.51	69.63	75.00
EfficientNetB4	efficientnet	98.15	79.07	86.67	**MobileNetV3Large**	mobilenet_v3	81.28	19.84	45.00
RegNetX008	regnet	98.15	78.39	86.67	**MobileNetV3Small**	mobilenet_v3	62.45	19.70	36.67
ResNet101V2	resnet_v2	98.97	78.25	86.67					

Figure 5 showcases the output images at different layers of a selected pre-trained model during the feature extraction process for each class of images in the study.

As depicted in Fig. 5, as the input image progresses through the hundreds of layers of the pre-trained models, different features are extracted from different channels (0, 1, 2: RGB). These features amount to millions of parameters in the models. Following these processes, the models were categorized to facilitate the application of the optimization approach employed in the study. These categories were established based on the model groups and were subsequently utilized in the second stage of the optimization approach, namely ensemble learning.

In addition to the categorization process, for a more in-depth analysis, the confusion matrix results of the models shown in Table 3 were also obtained. The obtained results are presented in Table 4.

Table 4 presents the test accuracy (test acc), precision (pre), recall (rec), F1-score (f1), and support (sup) values. The table demonstrates the successful results obtained by 10 different modules in the categorization process. Among these modules, the convnext model stands out as the most successful. Figure 6 provides graphical representations of precision, recall, and F1-score values for each class, based on the models listed in Table 4.

Figure 6 depicts the precision, recall, and F1-score values for the MPox, CPox, measles, and normal classes. These

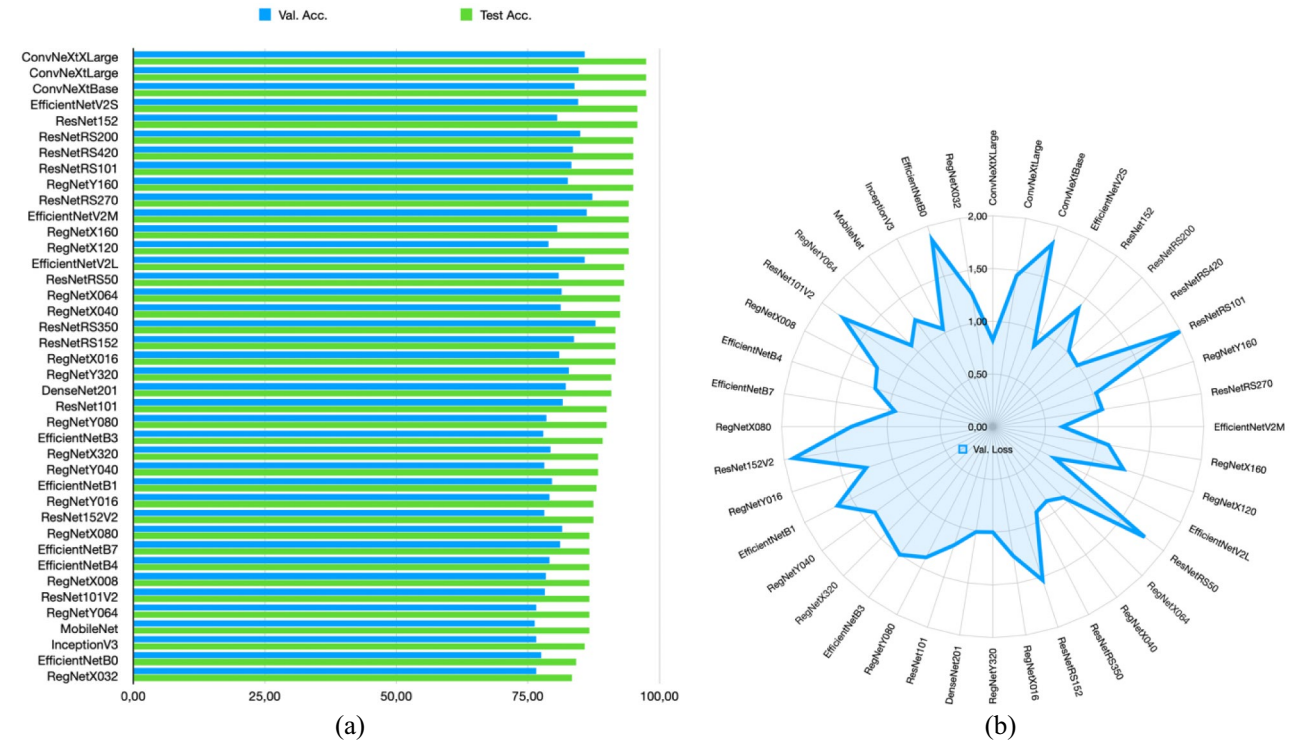


Fig. 4 After the elimination process, **a** accuracy rates and **b** loss rates

metrics provide insights into the performance of the models for each specific class. Figure 6 clearly demonstrates that the models exhibit higher precision values for the MPox and normal classes, while the recall values are higher for the chickenpox and measles classes.

The categories established in Table 4 were ranked based on F1-score values for each group. In the ranking process, priority was given to the MPox F1-score, which is the focus of the study, followed by CPox, measles, and normal F1-scores, respectively. F1-score, being the harmonic mean of precision and recall values, provides a more reliable evaluation of the classifier's performance by considering both false positives and false negatives. A higher F1-score indicates a better trade-off between precision and recall, suggesting that the model has achieved a good balance in correctly predicting positive instances while minimizing false positives and false negatives. The comparative graph of F1-scores based on the models' confusion matrix results is presented in Fig. 7.

The F1-score values obtained from the tests conducted in the first step of the optimization approach allow for the exploration of different model combinations in the second step. Comparative graphs of precision and recall (sensitivity) values for each disease group and the normal class are presented in Fig. 8.

In the second step of the study, informed by the outcomes of the initial step, the selection of ensemble learning participants was undertaken. A set of predetermined criteria were adhered to for the selection of models to be incorporated into the ensemble learning technique within the optimization approach (see Sect. 3). The foremost criterion involved the exclusion of models from the convnext module group that had exhibited remarkably high accuracy surpassing 97.5%. This was done to prevent overfitting. These models in the module group have also achieved high accuracy rates when used individually. After excluding the models from the convnext module group, ensemble learning technique was applied with the remaining models.

After following the steps in Algorithm 2 (see supplementary file), a total of 18 different models were identified under six separate SC. The confusion matrices of the selected models, represented as a heatmap, are presented in Fig. 9 in sequential order.

When Fig. 9 is analyzed, the EfficientNetV2S model, one of the models determined with SC1, incorrectly predicted four of the images belonging to the MPox class (three CPox, one measles). While two images belonging to the normal class were incorrectly predicted as MPox, one image each from the measles and CPox classes were incorrectly predicted. When the ResNet152 model is analyzed, it is seen that five images belonging to the MPox class were misclassified (two CPox, two normal, 1 measles), two images from the CPox class were predicted as normal, and one image from the normal class was predicted as CPox. The entire

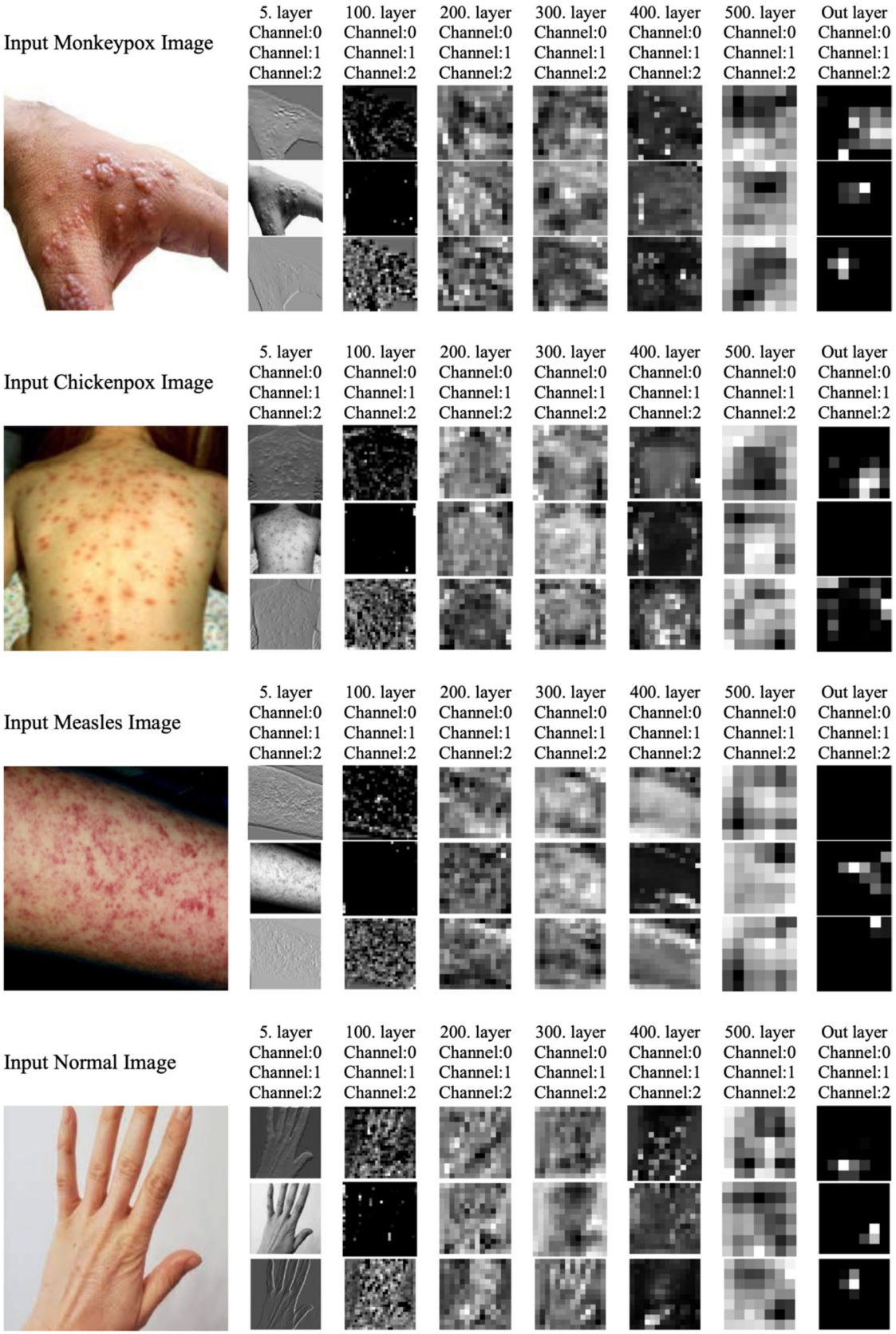


Fig. 5 Feature extraction in different layers

Table 4 Categorization and confusion matrix results

Module	Model	Test Acc	MonkeyPox			Chickenpox			Measles			Normal			Sup
			Pre	Rec	F1	Pre	Rec	F1	Pre	Rec	F1	Pre	Rec	F1	
convnext	ConvNeXtLarge	97.50	1.00	0.97	0.98	0.97	1.00	0.98	0.94	1.00	0.97	1.00	0.93	0.97	120
	ConvNeXtBase	97.50	1.00	0.93	0.97	0.94	1.00	0.97	0.97	1.00	0.98	1.00	0.97	0.98	120
	ConvNeXtXLarge	97.50	1.00	0.90	0.95	0.94	1.00	0.97	0.97	1.00	0.98	1.00	1.00	1.00	120
densenet	DenseNet201	90.83	1.00	0.90	0.95	0.83	1.00	0.91	0.86	1.00	0.92	1.00	0.73	0.85	120
efficientnet	EfficientNetB3	89.17	1.00	0.77	0.87	0.83	1.00	0.91	0.86	1.00	0.92	0.92	0.80	0.86	120
	EfficientNetB1	88.00	1.00	0.73	0.85	0.83	0.97	0.89	0.81	1.00	0.90	0.96	0.83	0.89	120
	EfficientNetB4	86.67	1.00	0.70	0.82	0.75	0.90	0.82	0.81	1.00	0.90	1.00	0.87	0.93	120
	EfficientNetB7	86.67	0.92	0.73	0.81	0.76	0.87	0.81	0.86	1.00	0.92	0.96	0.87	0.91	120
	EfficientNetB0	84.17	1.00	0.63	0.78	0.72	0.93	0.81	0.85	0.97	0.91	0.89	0.83	0.86	120
efficientnet_v2	EfficientNetV2S	95.83	1.00	0.90	0.95	0.94	0.97	0.95	0.94	1.00	0.97	0.97	0.97	0.97	120
	EfficientNetV2M	94.17	0.96	0.90	0.93	0.93	0.87	0.90	0.91	1.00	0.95	0.97	1.00	0.98	120
	EfficientNetV2L	93.33	0.96	0.87	0.91	0.88	0.93	0.90	0.91	1.00	0.95	1.00	0.93	0.97	120
inception_v3	InceptionV3	85.83	0.96	0.77	0.85	0.87	0.87	0.87	0.74	0.93	0.82	0.93	0.87	0.90	120
mobilenet	MobileNet	86.67	0.96	0.77	0.85	0.86	1.00	0.92	0.79	1.00	0.88	0.91	0.70	0.79	120
regnet	RegNetX160	94.17	1.00	0.93	0.97	0.94	1.00	0.97	0.88	1.00	0.97	0.96	0.83	0.89	120
	RegNetY160	95.00	1.00	0.90	0.95	0.97	1.00	0.98	0.86	1.00	0.92	1.00	0.90	0.95	120
	RegNetX016	91.67	0.97	0.93	0.95	0.88	0.97	0.92	0.86	1.00	0.92	1.00	0.77	0.87	120
	RegNetX120	94.17	1.00	0.90	0.95	0.83	1.00	0.91	0.97	1.00	0.98	1.00	0.87	0.93	120
	RegNetX040	92.50	1.00	0.87	0.93	0.86	1.00	0.92	0.88	1.00	0.94	1.00	0.83	0.91	120
	RegNetX064	92.50	0.93	0.93	0.93	0.83	0.97	0.89	0.97	1.00	0.98	1.00	0.80	0.89	120
	RegNetX008	86.67	1.00	0.87	0.93	0.79	0.90	0.84	0.77	1.00	0.87	1.00	0.70	0.82	120
	RegNetX320	88.33	1.00	0.83	0.91	0.87	0.90	0.89	0.77	1.00	0.87	0.96	0.80	0.87	120
	RegNetY080	90.00	0.96	0.87	0.91	0.79	1.00	0.88	0.91	1.00	0.95	1.00	0.73	0.85	120
	RegNetY016	87.50	1.00	0.80	0.89	0.77	1.00	0.87	0.83	1.00	0.91	1.00	0.70	0.82	120
	RegNetY040	88.33	1.00	0.73	0.85	0.77	1.00	0.87	0.86	1.00	0.92	1.00	0.80	0.89	120
	RegNetY064	86.67	1.00	0.73	0.85	0.75	1.00	0.86	0.83	1.00	0.91	1.00	0.73	0.85	120
	RegNetY320	90.83	1.00	0.70	0.82	0.88	1.00	0.94	0.83	1.00	0.91	0.97	0.93	0.95	120
	RegNetX080	86.67	1.00	0.67	0.80	0.78	0.97	0.87	0.86	1.00	0.92	0.89	0.83	0.86	120
	RegNetX032	83.33	1.00	0.53	0.70	0.71	0.97	0.82	0.79	1.00	0.88	1.00	0.83	0.91	120
resnet	ResNet152	95.83	0.94	1.00	0.97	0.97	0.93	0.95	0.97	1.00	0.98	0.96	0.90	0.93	120
	ResNet101	90.00	1.00	0.90	0.95	0.75	1.00	0.86	0.93	0.90	0.92	1.00	0.80	0.89	120
resnet_rs	ResNetRS50	93.33	1.00	0.90	0.95	1.00	1.00	1.00	0.79	1.00	0.88	1.00	0.83	0.91	120
	ResNetRS101	95.00	1.00	0.90	0.95	0.90	0.93	0.92	0.91	1.00	0.95	1.00	0.97	0.98	120
	ResNetRS200	95.00	1.00	0.90	0.95	0.88	0.93	0.90	0.94	1.00	0.97	1.00	0.97	0.98	120
	ResNetRS270	94.17	0.97	0.93	0.95	0.90	0.87	0.88	0.91	1.00	0.95	1.00	0.97	0.98	120
	ResNetRS152	91.67	1.00	0.87	0.93	0.88	0.97	0.92	0.97	1.00	0.98	1.00	1.00	1.00	120
	ResNetRS350	91.67	0.93	0.90	0.92	0.87	0.90	0.89	0.88	1.00	0.94	1.00	0.87	0.93	120
	ResNetRS420	95.00	1.00	0.83	0.91	0.88	0.97	0.92	0.94	1.00	0.97	1.00	1.00	1.00	120
resnet_v2	ResNet152V2	87.50	0.96	0.80	0.87	0.87	0.90	0.89	0.77	1.00	0.87	0.96	0.80	0.87	120
	ResNet101V2	86.67	0.95	0.67	0.78	0.84	0.90	0.87	0.81	1.00	0.90	0.90	0.90	0.90	120

measles class was correctly predicted. The last model in this group, ResNetRS420, incorrectly predicted four images from the MPox class (two CPox, two measles). It incorrectly predicted two images from the CPox class (as MPox) and one image each from the measles and normal classes. ResNetRS200, one of the models determined with SC2, correctly predicted the entire measles class. It predicted three images of the MPox class and one image of the normal class as CPox. It also incorrectly predicted two images of the CPox class as measles. EfficientNetV2M model incorrectly predicted five images of MPox class as CPox. It incorrectly predicted two images of the CPox class as MPox and one image each as measles and normal. It incorrectly predicted three images each from measles and normal classes. While

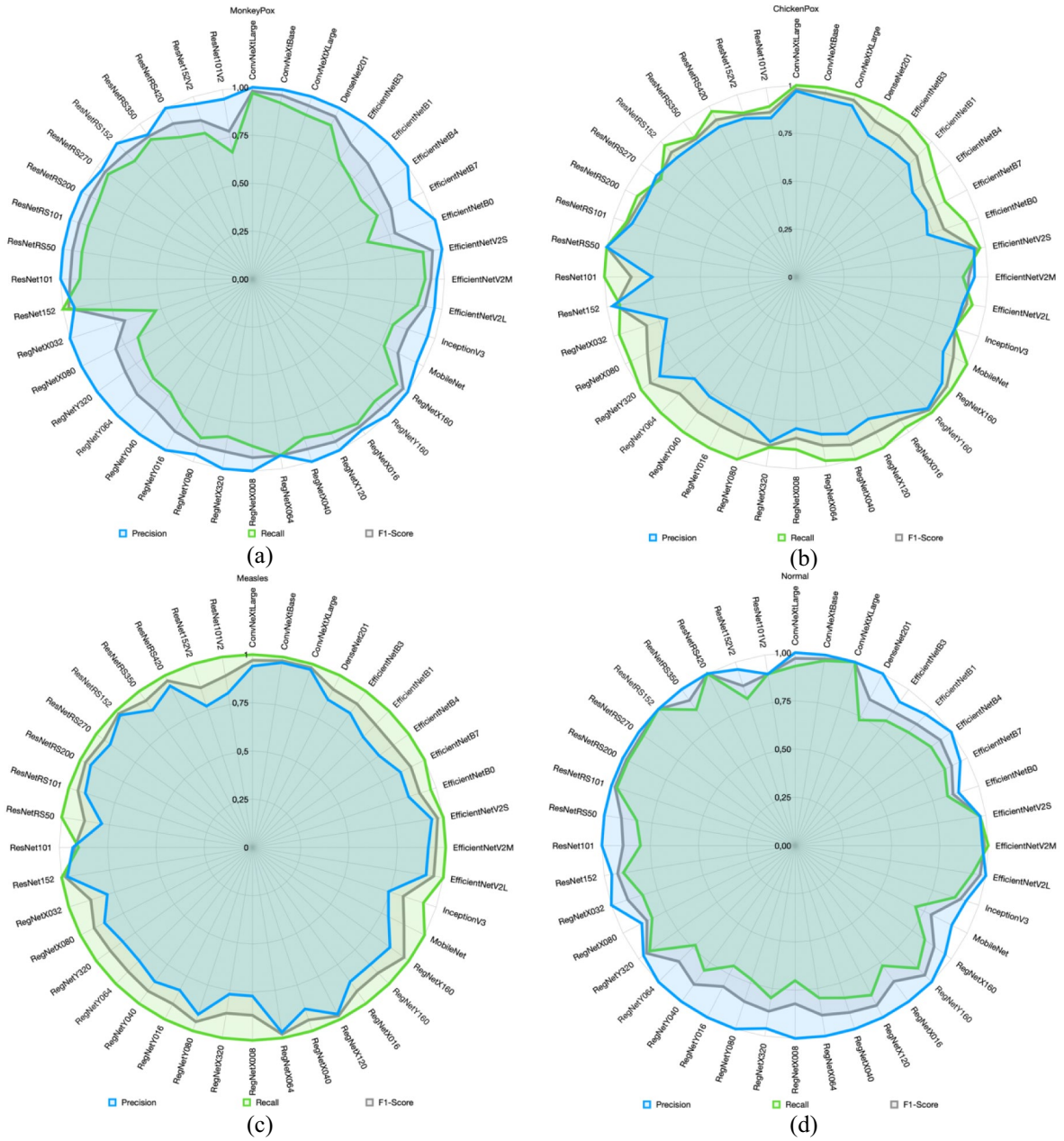


Fig. 6 Precision, recall, and f1-score values of **a** MPox, **b** CPox, **c** measles, and **d** normal classes

the RegNetX040 model correctly predicted the images belonging to the CPox and measles classes, it incorrectly predicted four images from the MPox class (three CPox, one measles) and five images from the normal class (three measles, two CPox). RegNetX160 model, one of the models determined with SC3, correctly predicted all of the CPox and measles classes. It incorrectly predicted two images of the MPox class (one CPox, one normal) and five images of the normal class (four measles, one CPox). The ResNetRS101 model correctly predicted all images belonging to the measles class. Two images belonging to the CPox class were classified as measles, three images belonging to the MPox class, and one image belonging to the normal class were incorrectly predicted. The ResNet101 model correctly predicted the entire CPox class. It incorrectly predicted three images of the measles class as CPox. It incorrectly predicted two images of the MPox class as CPox and 1 as measles. In the normal class, six images were incorrectly predicted (five CPox, one measles).

Among the models determined with SC4, the ResNetRS50 model correctly predicted all of the CPox and measles classes. three images from the MPox class and five images

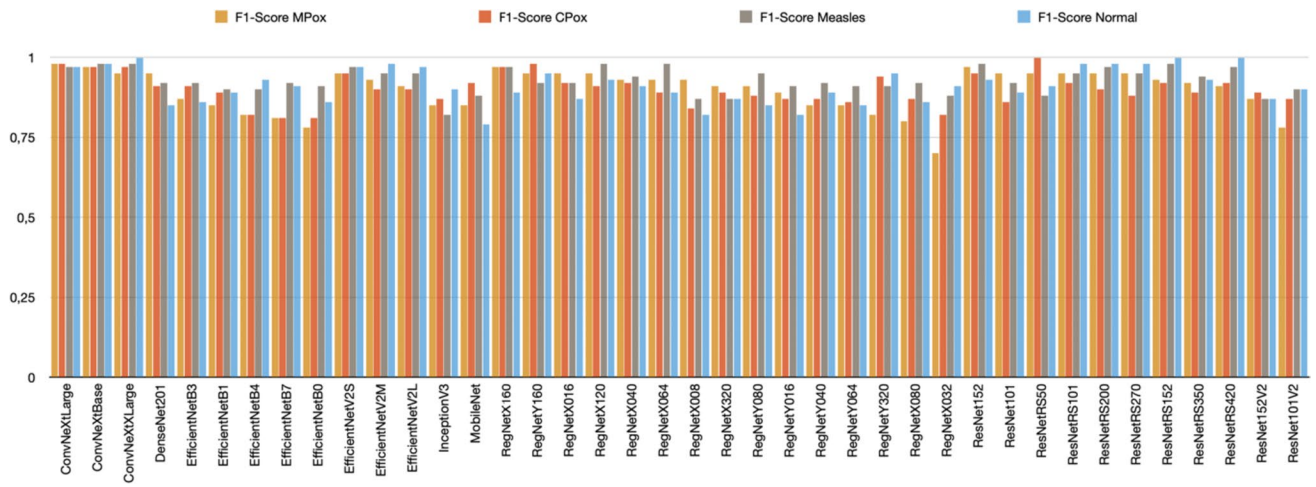


Fig. 7 Comparison of models' F1-Score values

from the normal class were incorrectly predicted as measles. The RegNetY320 model also correctly predicted all CPox and measles classes. Nine images from the MPox class were incorrectly predicted. Two images were incorrectly predicted from the normal class. The MobileNet model also correctly predicted all of the CPox and measles classes. It incorrectly predicted seven images from the MPox class and nine images from the normal class. Among the models determined with SC5, ResNetRS152 model correctly predicted all measles and normal classes, but incorrectly predicted one image in CPox and four images in MPox. The RegNetX120 model correctly predicted the CPox and measles classes, but incorrectly predicted three images in MPox and four images in normal. The EfficientNetV2L model correctly predicted measles, incorrectly predicted two images in CPox and normal classes and four images in MPox. Finally, the ResNetRS270 model, one of the models determined with SC6, correctly predicted the entire measles class, while predicting two images from the MPox class as CPox. It also predicted one image from the normal class as CPox. It incorrectly predicted four images from the CPox class. The RegNetY160 model correctly predicted all of the CPox and measles classes, but incorrectly predicted three images each from the MPox and normal classes. The EfficientNetB4 model correctly predicted the measles class, but incorrectly predicted three, four, and nine images from the CPox, normal, and MPox classes, respectively.

Second-step Ensemble Learning Results

The ensemble learning models determined based on the comprehensive analyses conducted in the first step of the study, and the results of the created models are presented in Table 5.

Among the six different ensemble learning models presented in Table 5, the most successful ROC AUC score was achieved by the EM3 model (0.9971). The models included in this ensemble model are RegNetX160, ResNetRS101, and ResNet101. These models, created according to SC3 criterion, were identified as the most successful model among all ensemble models. A very close result to this success was obtained by the EM1 model (0.9948). The models used in this model are EfficientNetV2S, ResNet152, and ResNetRS420. Based on these results, the results that produce higher performance than the highest performance ranking were achieved with the models determined according to the MPox selection criterion.

After conducting the tests in the study, the DNN models that achieved the most successful results among individual models and ensemble models were determined. The ConvNeXtBase, ConvNeXtLarge, and ConvNeXtXLarge models belonging to the convnext module were found to successfully classify MPox, CPox, measles, and normal images. All of these models achieved high accuracy rates (97.5%). Among the models, the ConvNeXtLarge model has the lowest loss rate (0.82). The evaluation of the ensemble learning models was conducted based on the ROC AUC score. Among these models, the EM3 model consisting of RegNetX160, ResNetRS101, and ResNet101 was identified as the model with the most successful score.

Discussion

In this section, the results obtained in this study are discussed and evaluated in comparison with the results of similar studies found in the literature review. The limitations of

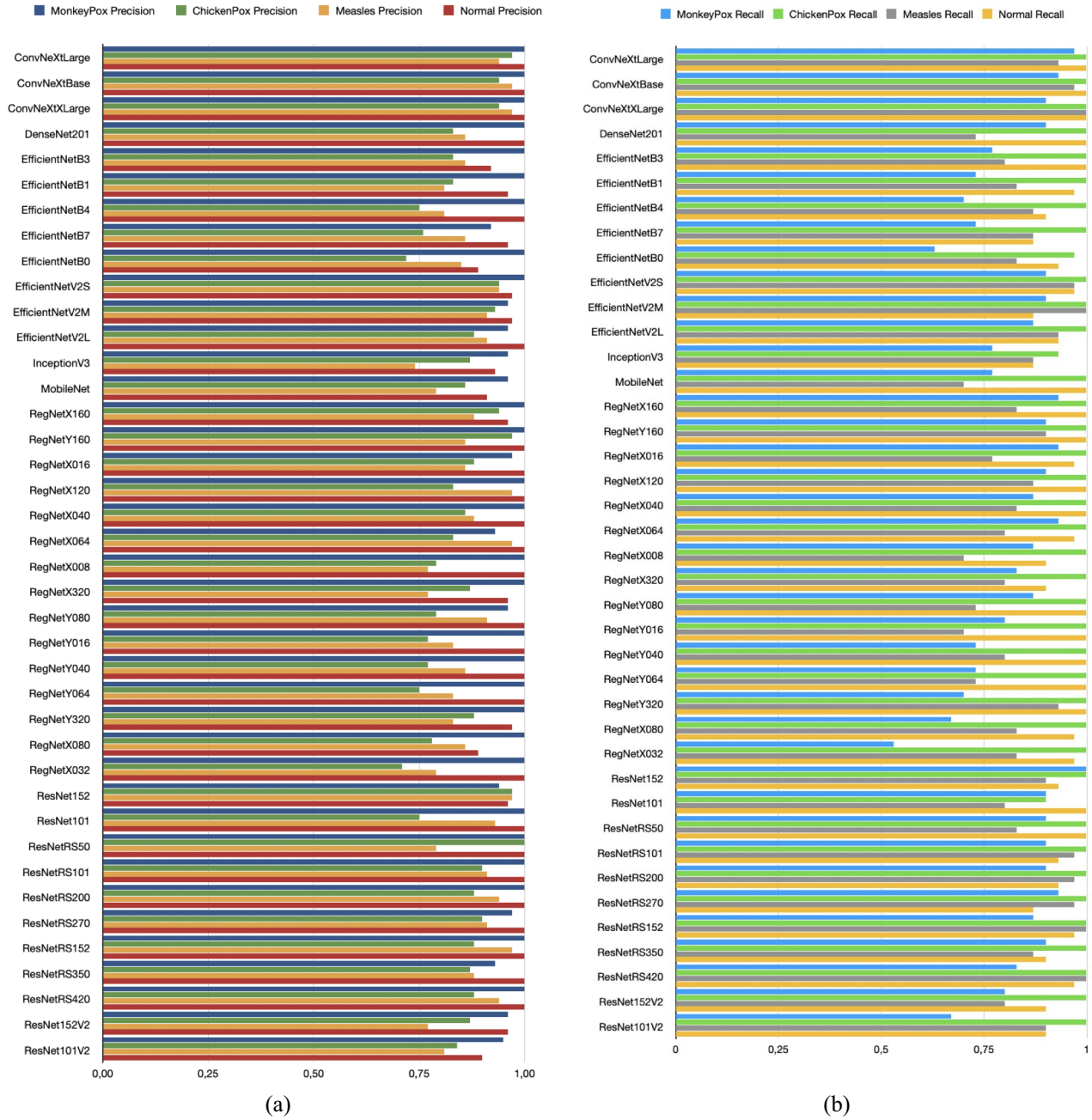
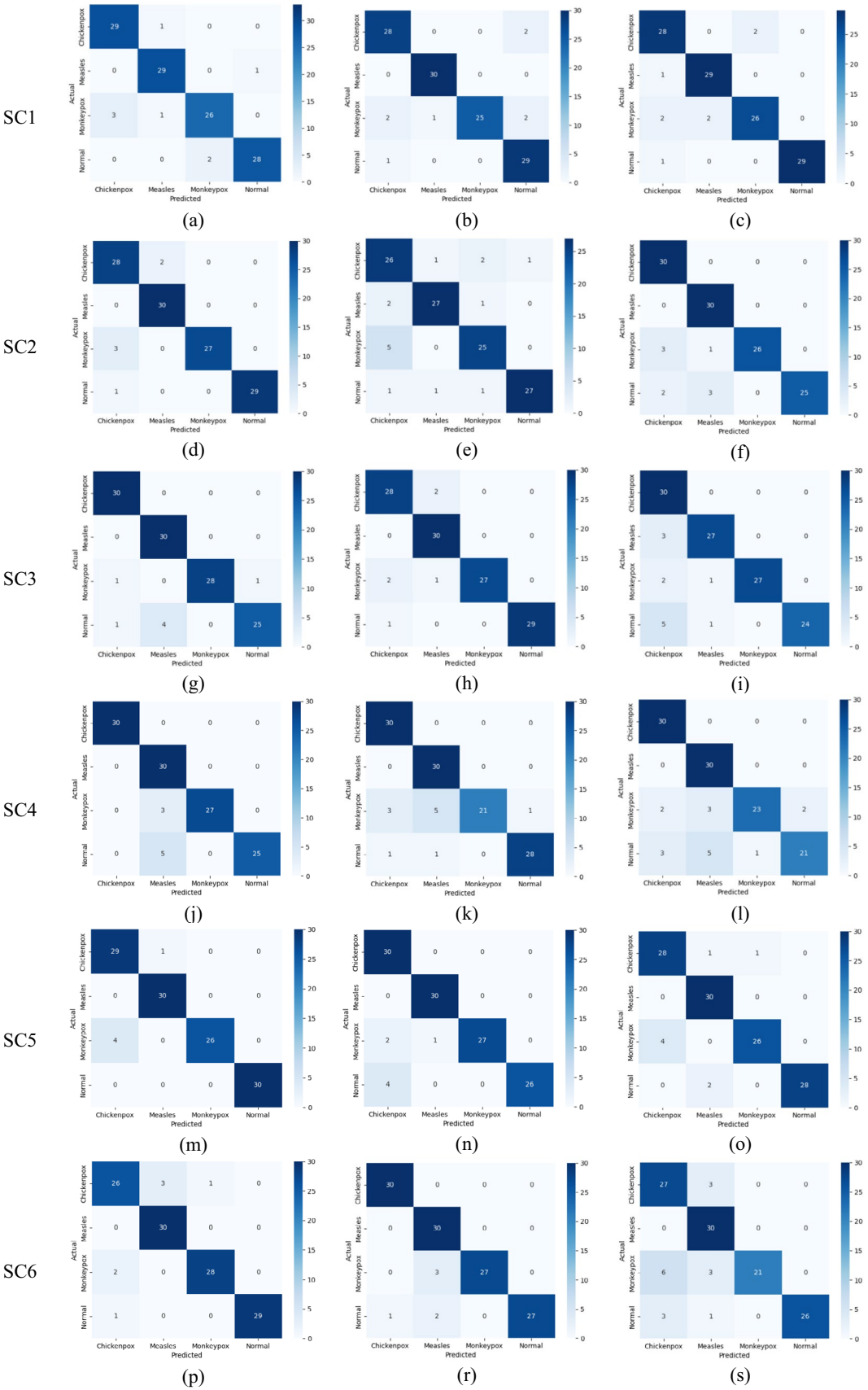


Fig. 8 Comparison of models' **a** precision, **b** recall values

the studies in the literature and the advantages of the results of this study over other studies are discussed.

One of the significant studies contributing to the field of MPox disease is conducted by Ahsan et al. [19]. They proposed a modified VGG16 model with an accuracy of $97 \pm 1.8\%$ on the study dataset. The main limitation of this study is that it only differentiates MPox disease from normal images. Since there are other visually similar diseases, one of the key challenges in clinical diagnoses is distinguishing these diseases from each other. Another important study in this field is conducted by Bala et al. [20]. They proposed a modified DenseNet-201 model, achieving accuracies of 93.19% and 98.91% on the original and augmented datasets, respectively. The limitation of this study is that it focuses on a single model instead of testing different pre-training models.

Akın et al. [21] proposed a decision support system using a dataset consisting of 572 images in two classes, monkeypox and normal. They evaluated 12 different CNN models, and MobileNetV2 achieved the best performance with an


SC1
SC2
SC3
SC4
SC5
SC6
Chickenpox
Measles
Monkeypox
Normal
Actual
Predicted
(a)
(b)
(c)
(d)
(e)
(f)
(g)
(h)
(i)
(j)
(k)
(l)
(m)
(n)
(o)
(p)
(r)
(s)

◂**Fig. 9** Confusion matrices of the models selected for Ensemble learning: **a** EfficientNetV2S, **b** ResNet152, **c** ResNetRS420, **d** ResNetRS200, **e** EfficientNetV2M, **f** RegNetX040, **g** RegNetX160, **h** ResNetRS101, **i** ResNet101, **j** ResNetRS50, **k** RegNetY320, **l** MobileNet, **m** ResNetRS152, **n** RegNetX120, **o** EfficientNetV2L, **p** ResNetRS270, **r** RegNetY160, **s**EfficientNetB4

accuracy of 98.25%, sensitivity of 96.55%, specificity of 100.00%, and F1-Score of 98.25%. The main limitation of this study is that it only differentiates MPox disease from normal images and does not include other similar diseases in the dataset. Another study that classifies different diseases, including monkeypox and measles, using transfer learning with nine different DNN models, was conducted by Yaşar [22]. DarkNet-53, DenseNet-201, InceptionV3, and Xception stood out as the most successful CNN architectures in the study. The study achieved high performance rates (93.6 to 96.4%) when comparing monkeypox and normal, but when other diseases were included, the performance rates ranged from 69 to 72%. Low performance rates were identified as the main limitation of this study.

In the study conducted by Haque et al. [23] on the MPox Skin Lesion Dataset (MSLD), VGG19, Xception, DenseNet121, EfficientNetB3, and MobileNetV2, along with integrated channel and spatial attention mechanisms, were used. The study classified monkeypox and other diseases with a validation accuracy of 83.89%. The major limitations of this study are the reliance on validation accuracy and the observed low accuracy rate. In another study, ResNet50, EfficientNetB3, and EfficientNetB7 were trained on the Kaggle Monkeypox dataset, and EfficientNetB3 outperformed with an 87% accuracy in the early detection of monkeypox skin lesions [24]. The main limitations of this study are the selection of only three models for comparison from numerous pre-trained models and the performance rate. Another study that distinguishes between MPox and CPox skin lesions was conducted by Uzun Ozsahin et al., [25]. They proposed a nine-layer CNN model for this classification problem, which outperformed all DL models with a test accuracy of 99.60%. The main limitation of this study is that it focuses only on two classes.

A study utilizing ensemble learning on MPox skin lesions was conducted by Pramanik et al. [26]. The researchers first considered three pre-trained base learners (InceptionV3, Xception, and DenseNet169) on the Kaggle Monkeypox dataset. They then extracted probabilities from these deep models to feed into the ensemble framework. Their model achieved average accuracy, precision, recall, and F1 scores of 93.39%, 88.91%, 96.78%, and 92.35%, respectively. The identified limitation of this study is that the models selected in the initial stage were not subjected to further refinement. The models were chosen by the researchers. Another ensemble learning method was proposed by [29]. In the study, 13 different pre-trained models were initially compared, and then the top two models were combined using the ensemble method. This technique, created using Xception

Table 5 Ensemble learning results

Criteria	Models	Ensemble model name	Result
SC1	EfficientNetV2S ResNet152 ResNetRS420	EM1	model1—weight: 0.2170—roc_auc_score: 0.7579 model2—weight: 0.6306—roc_auc_score: 0.9941 model3—weight: 0.1524—roc_auc_score: 0.8919 DirichletEnsemble roc_auc_score: 0.9948
SC2	ResNetRS200 EfficientNetV2M RegNetX040	EM2	model1—weight: 0.3427—roc_auc_score: 0.9165 model2—weight: 0.2276—roc_auc_score: 0.8933 model3—weight: 0.4297—roc_auc_score: 0.9493 DirichletEnsemble roc_auc_score: 0.9794
SC3	RegNetX160 ResNetRS101 ResNet101	EM3	model1—weight: 0.0292—roc_auc_score: 0.9509 model2—weight: 0.4344—roc_auc_score: 0.9377 model3—weight: 0.5363—roc_auc_score: 0.9880 DirichletEnsemble roc_auc_score: 0.9971
SC4	ResNetRS50 RegNetY320 MobileNet	EM4	model1—weight: 0.4475—roc_auc_score: 0.9319 model2—weight: 0.5395—roc_auc_score: 0.9294 model3—weight: 0.0129—roc_auc_score: 0.6650 DirichletEnsemble roc_auc_score: 0.9556
SC5	ResNetRS152 RegNetX120 EfficientNetV2L	EM5	model1—weight: 0.3382—roc_auc_score: 0.9622 model2—weight: 0.2596—roc_auc_score: 0.9081 model3—weight: 0.4022—roc_auc_score: 0.9572 DirichletEnsemble roc_auc_score: 0.9806
SC6	ResNetRS270 RegNetY160 EfficientNetB4	EM6	model1—weight: 0.1712—roc_auc_score: 0.9001 model2—weight: 0.8237—roc_auc_score: 0.9637 model3—weight: 0.0050—roc_auc_score: 0.7662 DirichletEnsemble roc_auc_score: 0.9746

and DenseNet169, yielded precision, recall, F1-score, and accuracy values of 85.44%, 85.47%, 85.40%, and 87.13%, respectively. Although the method applied in the study is innovative, the achieved performance rate stands out as a limitation. Singular models in different studies have also reached similar rates.

A transfer learning methodology was applied by Irmak et al. [27]. In their study, MobileNetV2, VGG16, and VGG19 models were employed on the MSID dataset. The highest performance scores were achieved by the MobileNetV2 model, with an accuracy of 91.38%, precision of 90.5%, recall of 86.75%, and an F1-score of 88.25%. The VGG16 model attained an accuracy of 83.62%, while the VGG19 model achieved 78.45% accuracy. One notable limitation of the study is its reliance solely on transfer learning methods. In the study by Altun et al. [30], the custom model MobileNetV3-s, EfficientNetV2, ResNet50, VGG19, DenseNet121, and Xception models were implemented. The optimized hybrid MobileNetV3-s model obtained the highest score, with an average F1-score of 0.98, AUC of 0.99, accuracy of 0.96, and recall of 0.97. Transfer learning and optimization techniques were applied in the study. However, a limitation of the study is that it only focused on the binary classification of MPox, without considering other diseases.

State-of-the-art (SOTA) methods for medical image processing include transfer learning, ensemble learning, fine-tuning, data augmentation, attention mechanism, generative adversarial networks (GANs), recurrent neural networks (RNNs), and long short-term memory (LSTM) networks and graph neural networks (GNNs). These methods are increasing day by day with scientific advances. Techniques such as transfer learning, ensemble learning, fine-tuning, and data augmentation are discussed in detail in "Material and Method" as they are used in this study. Attention mechanisms, like those used in Transformer models, were applied to focus on relevant regions within images, making the models more interpretable and effective. GNNs were used to analyze medical data with inherent graph structures, such as brain connectivity networks or molecular graphs in drug discovery. RNNs and LSTMs were used for tasks involving sequential medical data, like time series analysis of vital signs or ECG data. Considering all these aspects, the most suitable SOTA methods that can be used in this study have been brought together. Thus, robustness and generalization for addressing issues of model robustness, generalization to diverse patient populations, and data bias to ensure that AI models perform reliably across different clinical scenarios were provided. It is possible to apply different network applications such as RNN, LSTM, and GNN to different types of data. However, the algorithm suitable for the data set used in this study is the CNN algorithm. Attention mechanisms are among the studies that have been increasing especially in recent years. In future studies, vision transformers can be used for this disease group, and the results can be compared with this study.

The superior aspects of the results obtained in this study compared to the other studies in the literature can be summarized as follows:

- Improved performance: The results of this study demonstrate higher performance compared to the other studies in terms of accuracy, precision, recall, specificity, and F1-score. The fine-tuned transfer learning models (convnext module group) and the developed ensemble learning models in this study show better capabilities in classifying and detecting MPox and other similar diseases accurately.
- Ensemble learning approach: Unlike some studies in Table 1 that focus on individual models, this study employs an ensemble learning approach by combining different models. The ensemble technique used in this study enhances the overall performance by leveraging the strengths of multiple models and reducing the negative impact of other diseases with similar symptoms.
- Evaluation metrics: In addition to accuracy, this study evaluates the performance using metrics such as precision, recall, F1-score, and AUC. This comprehensive evaluation provides a more comprehensive understanding of the model's performance in different aspects.
- Dataset considerations: This study considers datasets that include MPox, CPox, and measles cases, allowing for a more realistic evaluation of the model's performance in real-world scenarios. By including a broader range of cases and considering potential confounding diseases, the developed models in this study demonstrate robustness and generalizability.

Overall, the results obtained in this study surpass the mentioned studies in the literature by achieving higher performance, following an optimization approach, utilizing an ensemble learning technique, considering comprehensive evaluation metrics, and using appropriate datasets for a more realistic assessment.

Conclusion

After the long-lasting effects of the Covid-19 pandemic worldwide, another outbreak was announced by the WHO in 2022. The Director-General of WHO, Tedros Adhanom Ghebreyesus, Ph.D., declared the current MPox outbreak as a PHEIC [51]. This decision was justified due to the increasing number of cases in over 70 countries, most of which are non-endemic, and the presence of milder nonspecific clinical symptoms without clear epidemiological links [52].

In this study, a methodology has been developed that successfully classifies three different types of diseases with similar skin symptoms, as well as normal skin types, using SOTA AI techniques such as optimization approaches, transfer learning, fine-tuning, and ensemble learning. Based on this, extensive tests on 71 different models from existing pre-trained model libraries have been conducted in this study, and the test results have been evaluated using different metrics. Through elimination and filtering methods, models that fell below the threshold value were eliminated, and a selection process was applied to the remaining models based on different criteria. Among the individual models, the ConvNeXtBase, large, and XLarge models in the convnext module group were the most successful models with an accuracy of 97.5%. Among the ensemble models created based on different criteria from the remaining models, the best result was achieved by the EM3 model, created based on the F1-Score performances for the MPox class (0.9971). This model includes the RegNetX160, ResNetRS101, and ResNet101 models.

The comprehensive tests conducted in the study are expected to contribute to the early diagnosis of these significant diseases that threaten global health during and before epidemics. It is also expected to serve as a basis for future research on this group of diseases. Additionally, this study has contributions to software-developing organizations for imaging devices used in clinical processes. Software developers can make decisions regarding the models they will use based on the results of this study. Researchers can save time by experimenting with the results of these models on different datasets of similar disease groups in the future. In addition, as a result of the increase in data sets of these disease groups in the coming years, different data sets can be tested using the hold-out technique in both stages applied in this study, and the results of the study can be explained comparatively.

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/s10278-023-00941-7.

Author Contribution Serkan Savaş: conceptualization, methodology, investigation, data acquisition, testing, validation, visualization, writing, original draft, and editing.

Data Availability The dataset used in this article is online available at https://doi.org/10.17632/R9BFPNVYXR.6.

Declarations

Ethics Approval Only data coming from publicly available datasets were used and ethics approval not applicable.

Consent to Participate Not applicable.

Consent for Publication Not applicable.

Competing Interests The author declares no competing interests.

References

1. WHO. (2023). *Mpox (monkeypox)*. Mpox (Monkeypox). https://www.who.int/news-room/fact-sheets/detail/monkeypox
2. Haque, Md. E., Ahmed, Md. R., Nila, R. S., & Islam, S. (2022a). *Classification of Human Monkeypox Disease Using Deep Learning Models and Attention Mechanisms*. https://arxiv.org/abs/2211.15459v1
3. CDC. (2022). *About Chickenpox*. About Chickenpox. https://www.cdc.gov/chickenpox/about/index.html#
4. NHS. (2022). *Measles*. Measles. https://www.nhs.uk/conditions/measles/
5. Delidow, B. C., Lynch, J. P., Peluso, J. J., & White, B. A. (1993). Polymerase Chain Reaction. In B. A. White (Ed.), *PCR Protocols: Current Methods and Applications* (pp. 1–29). Humana Press. https://doi.org/10.1385/0-89603-244-2:1
6. Binny, R. N., Priest, P., French, N. P., Parry, M., Lustig, A., Hendy, S. C., Maclaren, O. J., Ridings, K. M., Steyn, N., Vattiato, G., & Plank, M. J. (2023). Sensitivity of Reverse Transcription Polymerase Chain Reaction Tests for Severe Acute Respiratory Syndrome Coronavirus 2 Through Time. *The Journal of Infectious Diseases*, *227*(1), 9–17. https://doi.org/10.1093/infdis/jiac317
7. Kanji, J. N., Zelyas, N., MacDonald, C., Pabbaraju, K., Khan, M. N., Prasad, A., Hu, J., Diggle, M., Berenger, B. M., & Tipples, G. (2021). False negative rate of COVID-19 PCR testing: a discordant testing analysis. *Virology Journal*, *18*(1), 13. https://doi.org/10.1186/s12985-021-01489-0
8. Aggarwal, A., Rani, A., & Kumar, M. (2020). A robust method to authenticate car license plates using segmentation and ROI based approach. *Smart and Sustainable Built Environment*, *9*(4), 737–747. https://doi.org/10.1108/SASBE-07-2019-0083
9. Aggarwal, G., Jhajharia, K., Izhar, J., Kumar, M., & Abualigah, L. (2023). A Machine Learning Approach to Classify Biomedical Acoustic Features for Baby Cries. *Journal of Voice*. https://doi.org/10.1016/J.JVOICE.2023.06.014
10. Alhudhaif, A., Almaslukh, B., Aseeri, A. O., Guler, O., & Polat, K. (2023). A novel nonlinear automated multi-class skin lesion detection system using soft-attention based convolutional neural networks. *Chaos, Solitons & Fractals*, *170*, 113409. https://doi.org/10.1016/j.chaos.2023.113409
11. Güler, O., & Polat, K. (2022). Classification Performance of Deep Transfer Learning Methods for Pneumonia Detection from Chest X-Ray Images. *Journal of Artificial Intelligence and Systems*, *4*(1), 107–126. https://doi.org/10.33969/AIS.2022040107
12. Bütüner, R., & Calp, M. H. (2022). Diagnosis and Detection of COVID-19 from Lung Tomography Images Using Deep Learning and Machine Learning Methods. *International Journal of Intelligent Systems and Applications in Engineering*, *10*(2), 190–200. https://ijisae.org/index.php/IJISAE/article/view/1843
13. Raheja, S., Kasturia, S., Cheng, X., & Kumar, M. (2023). Machine learning-based diffusion model for prediction of coronavirus-19 outbreak. *Neural Computing and Applications*, *35*(19), 13755–13774. https://doi.org/10.1007/s00521-021-06376-x
14. Al-Saedi, D. K. A., & Savaş, S. (2022). Classification of Skin Cancer with Deep Transfer Learning Method. *Computer Science*, *IDAP-2022*(International Artificial Intelligence and Data Processing Symposium), 202–210. https://doi.org/10.53070/BBD.1172782
15. Madhu, G., Govardhan, A., Ravi, V., Kautish, S., Srinivas, B. S., Chaudhary, T., & Kumar, M. (2022). DSCN-net: a deep Siamese capsule neural network model for automatic diagnosis of malaria parasites detection. *Multimedia Tools and Applications*, *81*(23), 34105–34127. https://doi.org/10.1007/s11042-022-13008-6
16. Alhatemi, R. A. J., & Savaş, S. (2022). Transfer Learning-Based Classification Comparison of Stroke. *Computer Science*, *IDAP*

2022:(International Artificial Intelligence and Data Processing Symposium), 192–201. https://doi.org/10.53070/BBD.1172807
17. Chen, H., & Sung, J. J. Y. (2021). Potentials of AI in medical image analysis in Gastroenterology and Hepatology. *Journal of Gastroenterology and Hepatology*, *36*(1), 31–38. https://doi.org/10.1111/JGH.15327
18. Kolla, L., Gruber, F. K., Khalid, O., Hill, C., & Parikh, R. B. (2021). The case for AI-driven cancer clinical trials – The efficacy arm in silico. *Biochimica et Biophysica Acta (BBA) - Reviews on Cancer*, *1876*(1), 188572. https://doi.org/10.1016/J.BBCAN.2021.188572
19. Ahsan, M. M., Uddin, M. R., Farjana, M., Sakib, A. N., Momin, K. Al, & Luna, S. A. (2022). *Image Data collection and implementation of deep learning-based model in detecting Monkeypox disease using modified VGG16*. https://arxiv.org/abs/2206.01862v1
20. Bala, D., Hossain, M. S., Hossain, M. A., Abdullah, M. I., Rahman, M. M., Manavalan, B., Gu, N., Islam, M. S., & Huang, Z. (2023). MonkeyNet: A robust deep convolutional neural network for monkeypox disease detection and classification. *Neural Networks*, *161*, 757–775. https://doi.org/10.1016/J.NEUNET.2023.02.022
21. Akın, K. D., Gürkan, Ç., 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*, *40*, 106–110.
22. Yaşar, H. (2022). Transfer Derin Öğrenme Kullanılarak Maymun Çiçeği Hastalığının İki Sınıflı ve Çok Sınıflı Sınıflandırılması Üzerine Kapsamlı Bir Çalışma. *ELECO 2022 - Elektrik-Elektronik ve Biyomedikal Mühendisliği Konferansı*, 1–5.
23. Haque, Md. E., Ahmed, Md. R., Nila, R. S., & Islam, S. (2022b). Classification of Human Monkeypox Disease Using Deep Learning Models and Attention Mechanisms. *ArXiv*. https://arxiv.org/abs/2211.15459v1
24. Dwivedi, M., Tiwari, R. G., & Ujjwal, N. (2022). Deep Learning Methods for Early Detection of Monkeypox Skin Lesion. *2022 8th International Conference on Signal Processing and Communication, ICSC 2022*, 343–348. https://doi.org/10.1109/ICSC56524.2022.10009571
25. Uzun Ozsahin, D., Mustapha, M. T., Uzun, B., Duwa, B., & Ozsahin, I. (2023). Computer-Aided Detection and Classification of Monkeypox and Chickenpox Lesion in Human Subjects Using Deep Learning Framework. *Diagnostics*, *13*(2). https://doi.org/10.3390/diagnostics13020292
26. Pramanik, R., Banerjee, B., Efimenko, G., Kaplun, D., & Sarkar, R. (2023). Monkeypox detection from skin lesion images using an amalgamation of CNN models aided with Beta function-based normalization scheme. *PLOS ONE*, *18*(4), e0281815. https://doi.org/10.1371/JOURNAL.PONE.0281815
27. Irmak, M. C., Aydin, T., & Yağanoğlu, M. (2022). Monkeypox Skin Lesion Detection with MobileNetV2 and VGGNet Models. *2022 Medical Technologies Congress (TIPTEKNO)*, 1–4. https://doi.org/10.1109/TIPTEKNO56568.2022.9960194
28. 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), 79. https://doi.org/10.1007/s10916-022-01863-7
29. Sitaula, C., & Shahi, T. B. (2022). Monkeypox Virus Detection Using Pre-trained Deep Learning-based Approaches. *Journal of Medical Systems*, *46*(11), 78. https://doi.org/10.1007/s10916-022-01868-2
30. Altun, M., Gürüler, H., Özkaraca, O., Khan, F., Khan, J., & Lee, Y. (2023). Monkeypox Detection Using CNN with Transfer Learning. *Sensors*, *23*(4). https://doi.org/10.3390/s23041783
31. Bala, D., & Hossain, M. S. (2023). *Monkeypox Skin Images Dataset (MSID)*. *6*. https://doi.org/10.17632/R9BFPNVYXR.6
32. Cubuk, E. D., Zoph, B., Shlens, J., & Le, Q. V. (2019). RandAugment: Practical automated data augmentation with a reduced search space. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, *2020-June*, 3008–3017. https://doi.org/10.1109/CVPRW50498.2020.00359
33. Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on Image Data Augmentation for Deep Learning. *Journal of Big Data*, *6*(1), 1–48. https://doi.org/10.1186/S40537-019-0197-0/FIGURES/33
34. Pan, S. J., & Yang, Q. (2010). A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, *22*(10), 1345–1359. https://doi.org/10.1109/TKDE.2009.191
35. Weiss, K., Khoshgoftaar, T. M., & Wang, D. D. (2016). A survey of transfer learning. *Journal of Big Data*, *3*(1), 1–40. https://doi.org/10.1186/S40537-016-0043-6/TABLES/6
36. Keras. (2023). *Keras Applications*. Keras Applications. https://keras.io/api/applications/
37. TensorFlow. (2023). *Module: tf.keras.applications | TensorFlow v2.12.0*. Module: Tf.Keras.Applications | TensorFlow v2.12.0. https://www.tensorflow.org/api_docs/python/tf/keras/applications
38. Brown, G. (2010). Ensemble Learning. In G. I. Sammut Claude and Webb (Ed.), *Encyclopedia of Machine Learning* (pp. 312–320). Springer US. https://doi.org/10.1007/978-0-387-30164-8_252
39. Deng, L., & Yu, D. (2013). Deep learning: Methods and applications. *Foundations and Trends in Signal Processing*, *7*(3–4), 197–387. https://doi.org/10.1561/2000000039
40. Polikar, R. (2006). Ensemble based systems in decision making. *IEEE Circuits and Systems Magazine*, *6*(3), 21–44. https://doi.org/10.1109/MCAS.2006.1688199
41. Ng, K. W., Tian, G. L., & Tang, M. L. (2011). Dirichlet and Related Distributions: Theory, Methods and Applications. In *Dirichlet and Related Distributions: Theory, Methods and Applications*. John Wiley & Sons, Ltd. https://doi.org/10.1002/9781119995784
42. Borges, J. (2019). *DeepStack: Ensembles for Deep Learning*. https://github.com/jcborges/DeepStack
43. Howard, J., & Ruder, S. (2018). Universal Language Model Fine-tuning for Text Classification. *ACL 2018 - 56th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference (Long Papers)*, *1*, 328–339. https://doi.org/10.18653/v1/p18-1031
44. Yosinski, J., Clune, J., Bengio, Y., & Lipson, H. (2014). How transferable are features in deep neural networks? *Advances in Neural Information Processing Systems*, *4*(January), 3320–3328. https://arxiv.org/abs/1411.1792v1
45. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research*, *15*(56), 1929–1958. http://jmlr.org/papers/v15/srivastava14a.html
46. Gao, B., & Pavel, L. (2017). *On the Properties of the Softmax Function with Application in Game Theory and Reinforcement Learning*. https://arxiv.org/abs/1704.00805v4
47. Bock, S., Goppold, J., & Weiß, M. (2018). *An improvement of the convergence proof of the ADAM-Optimizer*. https://arxiv.org/abs/1804.10587v1
48. Kingma, D. P., & Ba, J. L. (2014). Adam: A Method for Stochastic Optimization. *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*. https://arxiv.org/abs/1412.6980v9
49. Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT press.
50. Gómez, R. (2018). *Understanding Categorical Cross-Entropy Loss, Binary Cross-Entropy Loss, Softmax Loss, Logistic Loss, Focal Loss and all those confusing names*. Github. https://gombru.github.io/2018/05/23/cross_entropy_loss/
51. WHO. (2022). *Second meeting of the International Health Regulations (2005) (IHR) Emergency Committee regarding the multi-country outbreak of monkeypox*. Second Meeting of the International Health Regulations (2005) (IHR) Emergency Committee Regarding

the Multi-Country Outbreak of Monkeypox. https://www.who.int/news/item/23-07-2022-second-meeting-of-the-international-health-regulations-(2005)-(ihr)-emergency-committee-regarding-the-multi-country-outbreak-of-monkeypox

52. Nuzzo, J. B., Borio, L. L., & Gostin, L. O. (2022). The WHO Declaration of Monkeypox as a Global Public Health Emergency. *JAMA*, *328*(7), 615–617. https://doi.org/10.1001/JAMA.2022.12513