[1] Jinn-Yi Yeh *
[2] Ching-Hung Yu

Monkeypox Detection Using Particle Swarm Optimization-Based Extreme Learning Machine

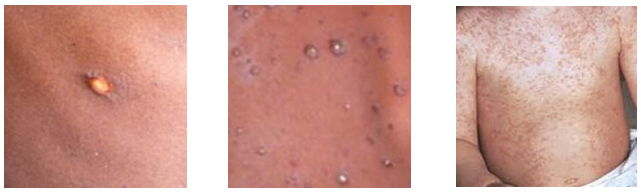
Journal of Electrical Systems

Abstract: - Monkeypox is caused by a virus that causes skin lesions and has an average mortality rate of 11% in unvaccinated patients. This study proposes a particle swarm optimization-based extreme learning machine (PSO-ELM) model to detect monkeypox disease. First, monkeypox images were read from two public databases. Secondly, image preprocessing is implemented, including noise removal, contrast enhancement, resizing, rotation and flipping. Third, a convolution neural network (CNN) model is then used to extract the required features from the processed images. Finally, the features extracted by the CNN were fed into PSO-ELM to distinguish monkeypox, chickenpox, and measles. A ten-fold cross-validation method is used to evaluate the performance of the proposed system. Experimental results show that the accuracy, precision, recall and F1-score of PSO-ELM are 0.961, 0.969, 0.961 and 0.964 respectively, which is better than the classic CNN. Moreover, the average training time of PSO-ELM is only about 16 seconds, which is significantly lower than classic CNN. This research result can be applied in clinical detection of monkeypox.

Keywords: Convolution neural network, Extreme learning machine, Monkeypox, Particle swarm optimization.

I. INTRODUCTION

Monkeypox disease is caused by a virus that is contagious to both humans and animals. Monkeypox virus belongs to the genus Orthopoxvirus, which also includes chickenpox virus and smallpox virus. Figure 1 shows example images of monkeypox, chickenpox, and measles [1]. According to the world health organization (WHO) report, 113 countries and regions have been affected by monkeypox from 2022 to 2023, with more than 86,000 confirmed cases and 280 deaths. The average death rate among unvaccinated patients is about 11%.



(1) Monkeypox (2) Chickenpox (3) Measles

Figure 1. Sample images from various medical datasets [1]

In order to detect monkeypox, there are certain requirements for hardware facilities and skilled technical personnel. However, monkeypox is usually prevalent in poorer areas. It is not easy to meet the above requirements. Therefore, a new detection method is needed to meet the low-cost and rapid detection. Extreme learning machine (ELM) is one of the candidate solutions. In ELM, hidden nodes are randomly started and then fixed without iterative adjustment. The only free parameters that need to be learned are the weights between the hidden and output layers. Compared with convolutional neural network (CNN), ELM has faster learning speed and higher versatility [2].
However, in ELM, the weights between the input layer and the hidden layer and the bias values of the hidden layer are random values and will not be changed iteratively. The weight values may not be optimal. Therefore, the metaheuristic algorithm can be used for finding the best solution. There is a category of metaheuristic algorithms that usually simulate the ecology or behavior in nature, such as particle swarm optimization (PSO), whale optimization algorithm (WOA), and artificial fish-swarm algorithm (AFSA).
Based on pilot studies, this study decided to use a PSO-based ELM model for monkeypox detection. The proposed system will combine image preprocessing, CNN models, PSO and ELM to detect different orthopoxvirus lesions, such as monkeypox, chickenpox, and measles. The remainder of this paper is organized as follows: Section 2

[1] Department of Management Information Systems, National Chiayi University, Chiayi City, Taiwan. jyeh@mail.ncyu.edu.tw

[2] Department of Management Information Systems, National Chiayi University, Chiayi City, Taiwan. a7376669@gmail.com

* Corresponding Author Email: jyeh@mail.ncyu.edu.tw

contains a literature review, Section 3 details the materials and research methods used in this study, Section 4 presents an analysis of the experimental results, and Section 5 concludes the paper.

II. LITERATURE REVIEW

A. Computer-Aided Detection of Monkeypox

Several computer-aided detection system for monkeypox are reviewed. Researchers [3] introduced the concept of transfer learning into three different pre-trained deep learning models of non-medical images for feature extraction, namely ResNet50, InceptionV3, and VGG16. As a result, monkeypox, chickenpox and measles were classified, and the accuracies of ResNet50, InceptionV3 and VGG16 were 81.48%, 82.96% and 74.07% respectively.

Another study [1] proposed a modified VGG16 deep learning model. Since the number of original images in the image set used was 161, image enhancement technology was used to increase the number of images. However, in order to avoid overfitting, the fully connected layer of the model is added to classify monkeypox, chickenpox, measles, and normal. The results were divided into experiment 1 (monkeypox, chickenpox) and experiment 2 (monkeypox, others), with accuracy of 83% and 78% respectively.

Thirteen pre-trained deep learning models are compared to determine the best model for detecting monkeypox virus [4]. The study also proposed an integration method that combines the two best-performing deep learning models (Xception, DenseNet-169) to improve the overall performance of monkeypox virus detection, changing the original 3-layer fully connected layer to global average pooling and a layer fully connected layer. By taking advantage of multiple models, this method has shown significant improvements in precision, recall, F1-score and accuracy of 85.44%, 85.47%, 85.40%, and 87.13%, respectively.

In another study [5], deep learning methods with transfer learning tools and hyperparameter optimization was combined together to detect monkeypox disease. The custom model included MobileNetV3-s, EfficientNetV2, ResNET50, Vgg19, DenseNet121, and Xception models. The optimized hybrid MobileNetV3-s model achieved the best score, with an average F1-score of 0.98, area under the curve (AUC) of 0.99, accuracy of 0.96, and recall of 0.97. Another study [6] proposed the MonkeyNet model, an improvement of the deep CNN model based on DenseNet-201, for classifying monkeypox from skin images. The proposed model correctly classified image classes in multi-class classification of original and enhanced datasets with accuracy of 93.19% and 98.91%, respectively.

B. Extreme Learning Machine for Computer-Aided Systems

Past studies were examined within the scope of deep learning models combined with ELM for medical image analysis. Researchers [7] combined ELM into deep learning networks. They used VGG16 and AlexNet to extract features and then fed them into ELM to detect brain cancer on brain MRI images. This study used three data sets: RIDER, Figshare, and REMBRANDT. The performance evaluation was cross-validated using different depth networks, and the accuracy of the three data sets were 99%, 97.64% and 98.46% respectively.

Another study [8] proposed a breast cancer detection method using deep feature fusion with convolutional neural networks and then using ELM for classification. They used a dataset of 400 female mammograms. The results show that the best accuracy for detecting benign and malignant breast classification based on the deep feature model is 86.5%. In another study [9], a hybrid improved water cycle algorithm-accelerated particle swarm optimization (IWCA-APSO) is proposed for weight optimization of the ensemble extreme learning machine (EELM) model for the classification and detection of breast cancer from mammogram. On the INbreast dataset, the IWCA-APSO based EELM classification showed sensitivity, specificity, accuracy, and calculation time of 99.67%, 99.71%, 99.36% and 23.88 seconds, respectively.

Another study [10] applied ELM and multivariate calibration regression model to analyze the results of Raman spectroscopy and determine the concentrations of components in blood samples, including glucose, cholesterol, and triglycerides. This study further applied the adaptive differential evolution artificial bee colony (SADEABC) algorithm to improve the accuracy and robustness of the model. The resulting coefficient of determination, calibration root mean square error, prediction root mean square error, and relative percent deviation were 0.9822, 0.3993, 0.3827, and 6.6679 for glucose, 0.9786, 0.2104, 0.2088 and 5.9533 for cholesterol, and 0.9921, 0.2744, 0.3433 and 10.5075 for triglyceride, respectively.

In another study [11], a COVID-19 detection system was proposed that utilized PSO-ELM as a classifier and Mel Frequency Cepstral Coefficients (MFCC) for feature extraction. In this study, respiratory sound samples were taken from the Corona Hack Respiratory Sound Dataset (CHRSD). Experimental results show that the PSO-ELM was

capable of attaining the highest accuracy, reaching 95.83%, 91.67%, 89.13%, 96.43%, 92.86%, 88.89%, 96.15%, 96.43%, 88.46%, 96.15%, 96.15%, 95.83%, and 82.89% for breath deep, breath shallow, all breath, cough heavy, cough shallow, all cough, count fast, count normal, all count, vowel a, vowel e, vowel o, and all vowel scenarios, respectively.

Based on the above literature review, it is found that the extreme learning machine based on particle swarm optimization is a machine learning algorithm that can be considered accurate and fast in the detection and classification process.

III. Materials And Methods

The general diagram of the proposed monkeypox detection system using the PSO-ELM model is illustrated in Figure 2. The diagram consists of various stages that are used to create the monkeypox detection system. The first stage refers to read monkeypox images from two public databases. The second stage consists of the image pre-processing including noise removal, contrast enhancement, resizing, rotation and flipping. In the third stage, the CNN method is then utilized to extract the needed features from the processed images. Last, in the fourth stage, the CNN extracted features are fed into the PSO-ELM model to detect monkeypox. These main stages of the proposed monkeypox detection system are described in the following subsections.

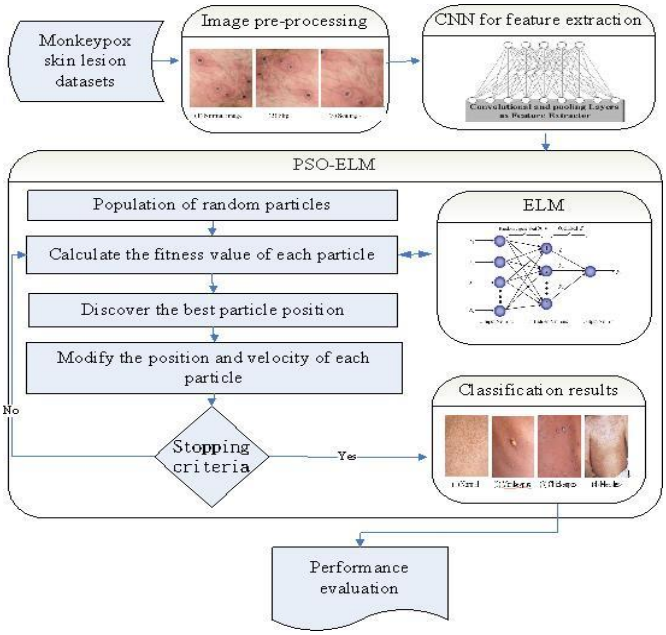


Figure 2. Block diagram of the proposed monkeypox detection system

C. Datasets

In this study, monkeypox images were acquired from two public databases the Monkeypox-dataset-2022 [1] and Monkeypox Skin Lesion Dataset (MSLD) [3]. The original images in the formal dataset include 43 monkeypox images, 47 chickenpox images, 17 measles images and 66 normal images, for a total of 173 images. After image preprocessing was used to expand the images, there were 587 monkeypox images, 641 chickenpox images, 232 measles images, and 900 normal images, for a total of 2360 images.

The latter dataset has been verified by dermatologist experts. The original images include 279 monkeypox images, 107 chickenpox images, 91 measles images, and 293 normal images, for a total of 770 images. After augmentation, there were 2642 images of monkeypox, 1777 images of chickenpox, 1502 images of measles, and 2768 normal images, for a total of 8689 images. Table 1 lists the number of sample images used in this study.

Table 1. Dataset information

Image classes	Monkeypox-dataset		MSLD	
	Original	Augmented	Original	Augmented
Monkeypox	43	587	279	2642
Chickenpox	47	641	107	1777
Measles	17	232	91	1502

Normal	66	900	273	2768
Total	173	2360	770	8689

D. CNN for Feature Extraction

For feature extraction, this study uses DenseNet [12], a deep learning architecture known for its dense connection structure. The architecture of DenseNet mainly includes dense block, transition block and global average pooling layer. Dense blocks are the basic module of DenseNet, where each layer is directly connected to all previous layers. This means that the input of layer X is the concatenation of stacked feature maps from layers 0 to x-1. In order to control the size of the feature map, a transition layer is inserted between dense blocks, which usually includes batch normalization, 1x1 convolution and pooling operations. Transition layers help reduce the computational burden while maintaining the parametric efficiency of the network. The global average pooling layer is at the end of the network. The spatial dimension of the feature map is reduced to 1x1 through global average pooling, and then connected to the final classification layer. This helps reduce the number of parameters, prevents overfitting, and improves the generalization ability of the model. Overall, the DenseNet architecture achieves more effective parameter sharing and feature reuse through design strategies.

E. Extreme Learning Machines

The extreme learning machines (ELM) is a neural network with one hidden layer, which connects the input and output layers to each other [2]. In ELM, the input weight is calculated between the input node and the hidden neuron. While hidden biases are used to interconnect all hidden neurons, output weights are used to interconnect hidden neurons to nodes in the output layer. Figure 3 shows the extreme learning machine architecture.

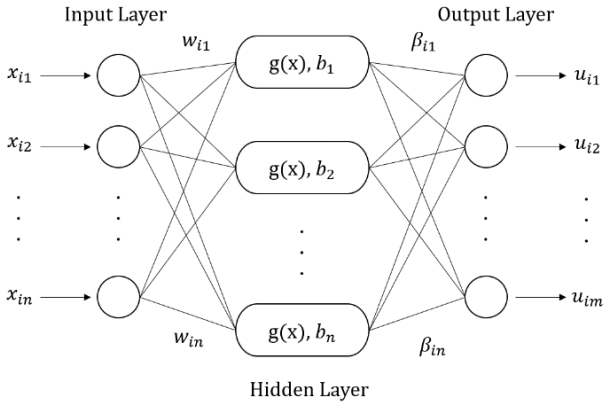


Figure 3. Extreme learning machine architecture [2]

The following is the formula definition of ELM：

$$\sum_{i=1}^{L} \beta_i * g(w_i, b_i, x_i) = u_i \quad (1)$$

where w_i is the weight of the input node and the hidden node, g(x) represents the activation function used (using the sigmoid function), b_i represents the bias value of the hidden node, β_i is the weight between the hidden node and the output. ELM is a powerful and simple machine learning algorithm suitable for applications that pursue high performance and fast training.

F. Particle Swarm Optimization

The core concept of PSO is derived from the observation of group behavior in nature [13]. This algorithm simulates the social behavior of a flock of birds or a school of fish, in which each individual (particle) adjusts its movement direction based on its own state and experience, as well as information from other individuals in the group. All these particles have a current position vector $x_i = (x_{i1} + x_{i2} + \ldots x_{iN})$. There is an optimal vector $p_i = (p_{i1} + p_{i2} + \ldots p_{iN})$ and a current speed vector $v_i = (v_{i1} + v_{i2} + \ldots v_{iN})$, where walk represents the i^{th} particle in N-dimensional space. Initially, PSO starts with a population of random particles and then uses generational updates to search for optimal values. Each particle in the PSO will be updated with two "personal best" values in each iteration, namely P_{bset} and G_{bset}. The local optimal solution P_{bset} (personal best) represents the extent to which the particle reaches the optimal solution in terms of its position vector and therefore the fitness

value will be saved. The global best position, G_{bset}, represents the best position reached so far by any particle in the swarm. When the fitness function can be expressed as *f*, the value of P_{bset} for any particle updated in iterations falcon can be given as:

$$p_i(t+1)\begin{cases} p_i(t) & if\ f(x_i(t+1) \geq \mathrm{f}\,(p_i(\mathrm{t}\ +1))) \\ x_i(t) & if\ f(x_i(t+1) < \mathrm{f}\,(p_i(\mathrm{t}\ +1))) \end{cases} \quad (2)$$

The value of G_{bset} is updated based on the value of P_{bset} of all particles as:

$$G_{bset}\ \in \{p_0(t), p_1(t), \ldots p_m(t)\} \quad (3)$$

To update the motion vector, the performance can be accomplished using:

$$v_i(t+1) = wv_i(\mathrm{t}) + c_1r_1(p_i(\mathrm{t})\text{-}x_i(\mathrm{t})) + c_2r_2\left(G_{best} - x_i(t)\right) \quad (4)$$

where *w* represents the inertia coefficient, which is used to control the influence of the previous velocity vector on the newer velocity, *v* represents the velocity vector, *c* represents the learning constant of individuals and groups, *r* is a random number between 0 and1, and x_i represents the current location. To update the position, the function can be expressed by:

$$x_i(\mathrm{t+1}) = x_i(\mathrm{t}) + v_i(t+1) \quad (5)$$

Finally, the iterative process is repeated until the stopping condition is met.

G. Performance Evaluation

In this study, the performance evaluation uses confusion matrices and commonly used criteria in the field of image recognition such as accuracy, precision, recall, and F1-Score. A confusion matrix is a visualization tool commonly used in machine learning and statistical classification problems. Each column represents an example prediction of a category, and each row represents an actual example of the category. The results of the classification can be represented in Table 2.

Table 2: Confusion Matrix

		Predicted class	
		Class 1	Class 2
Actual class	Class 1	A:T+	C:F-
	Class 2	B:F+	D:T-

The definitions of these four symbols are as follows: True positive (A: T+), which is actually class 1 and predicted to be class 1; False positive (B: F+), which is actually class 2 but predicted to be class 1; False negative (C: F-), which is actually class 1 but predicted to be class 2; and True negative (D: T-), actually class 2 and predicted to be class 2. The four criteria are defined as follows:

$$Accuracy = \frac{(\mathrm{A}:\mathrm{T}+)+(\mathrm{D}:T-)}{(A:T+)+(B:F+)+(C:F-)+(D:T-)} \quad (6)$$

$$Precision = \frac{(A:T+)}{(A:T+)+(C:F-)} \quad (7)$$

$$Recall = \frac{(A:T+)}{(A:T+)+(B:F+)} \quad (8)$$

$$\text{F1-Score} = 2 \times \frac{\text{Precision}\times\text{Recall}}{\text{Precision}+\text{Recall}} \quad (9)$$

IV. Experimental Results

The experiments in this study were done on a PC with 32GB RAM, INTEL i5-13400 CPU and NVDIA 3060 GPU. This study uses Anaconda in Windows 10 to build different environments with different versions of programming languages and packages. Python 3.7.6 is used as the development tool, and the related packages include tensoflow 1.14.0, scikit-image 0.10.2, keras 2.3.1, and opensoflow 2.3.1.

H. Image Pre-processing Results

This study uses rotation, scaling, horizontal flipping, vertical flipping, cropping, and Gabor filter and histogram equalization to improve the contrast of the image. Figure 4 shows the images after flipping and scaling. Figure 5 shows images with different tones. The approach is to scale the pixel values to between 0 and 1 to better adapt to the training of deep neural networks.

(1) Normal image (2) Flip (3) Scaling

Figure 4. Images of flipping and scaling

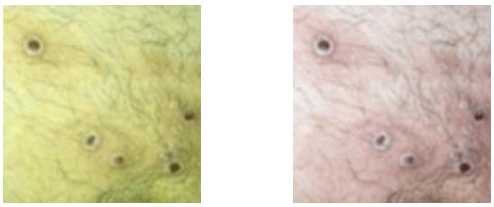
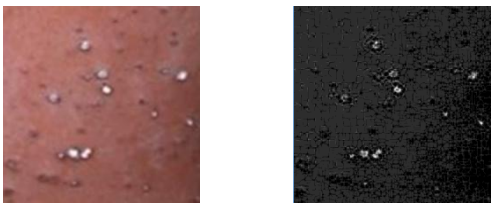


Figure 5. Images of different tones

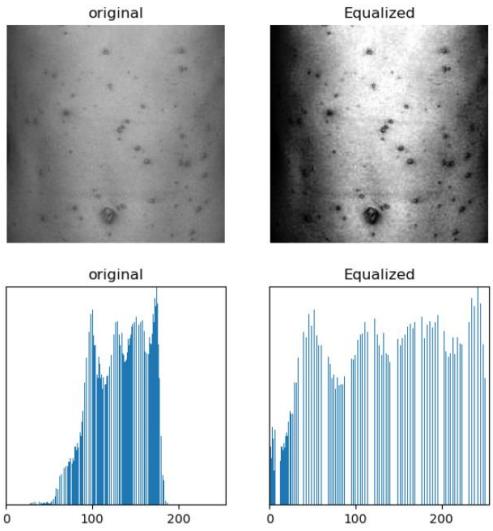
The Gabor filter is similar to the human visual system in the representation of frequency and direction, and is suitable for texture expression and recognition. It is mainly used for texture analysis, edge detection, and feature extraction. That is, it passes frequencies in a specific band and attenuates other frequencies outside that band. Figure 6 shows the image after Gabor filter operation.



(1) Normal image (2) Gabor filter result

Figure 6. Image after Gabor filter operation

In this study, histogram equalization was used to improve image contrast. Histogram equalization is an image processing technology that can change the intensity distribution of the image to make the histogram distribution of the image more uniform, thereby improving the contrast of the image. Figure 7 shows the image after histogram equalization operation.



(1) Normal image (2) Histogram equalization result

Figure 7. Image after histogram equalization operation

I. Feature Extraction Results

This study uses DenseNet to extract features for further experiments, including 50 features and 100 features. Table 3 lists part of the extracted 50 features.

Table 3. Part of 50 features extracted by DenseNet

Classes	Feature_1	Feature_2	…	Feature_50
Monkeypox	0.0010	-0.0183	…	-0.0194
Monkeypox	-0.0049	-0.0101	…	-0.0274
Chickenpox	0.0013	0.0260	…	0.0015
Chickenpox	0.0138	0.0088	…	0.0006
Measles	-0.0175	0.0768	…	0.0163
Measles	-0.0362	0.0044	…	0.0013
Normal	0.0189	0.0055	…	0.0178
Normal	0.0207	0.0175	…	-0.0617

J. ELM Results

The first experiment is to input the numerical features on the basic ELM model for monkeypox detection. Some examples of basic ELM prediction results are shown in Figure 8. The confusion matrix that classifies the predictions into four categories such as monkeypox, chickenpox, measles normal is shown in Figure 9.

Figure 8. Some examples of basic ELM prediction results

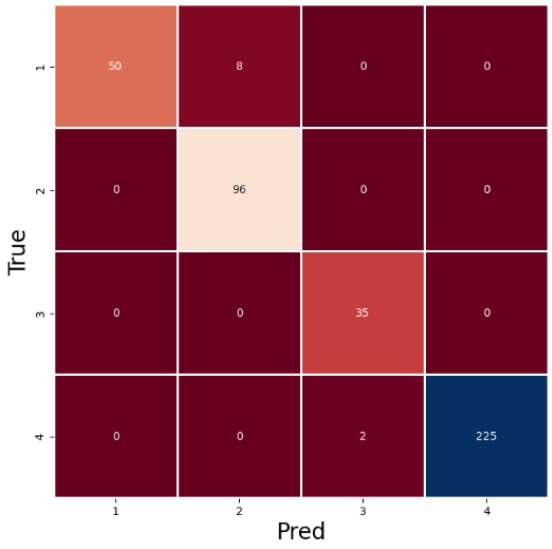


Figure 9. Confusion matrix of the basic ELM

K. PSO-ELM Results

The second experiment is to input the numerical features on the PSO-ELM model for monkeypox detection. Figure 10 shows that the error will gradually level off after about 20 epochs of training. Some examples of PSO-ELM

prediction results are shown in Figure 11. The confusion matrix that classifies the predictions into four categories is shown in Figure 12.

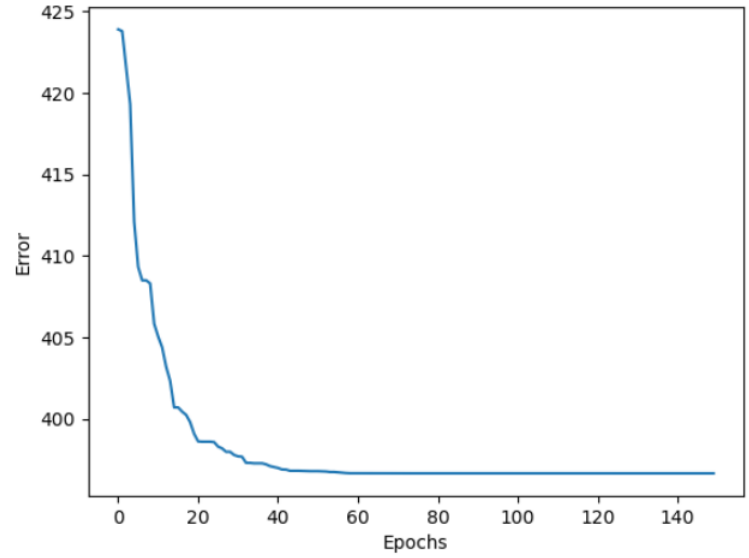


Figure 10. PSO-ELM training error vs. epochs

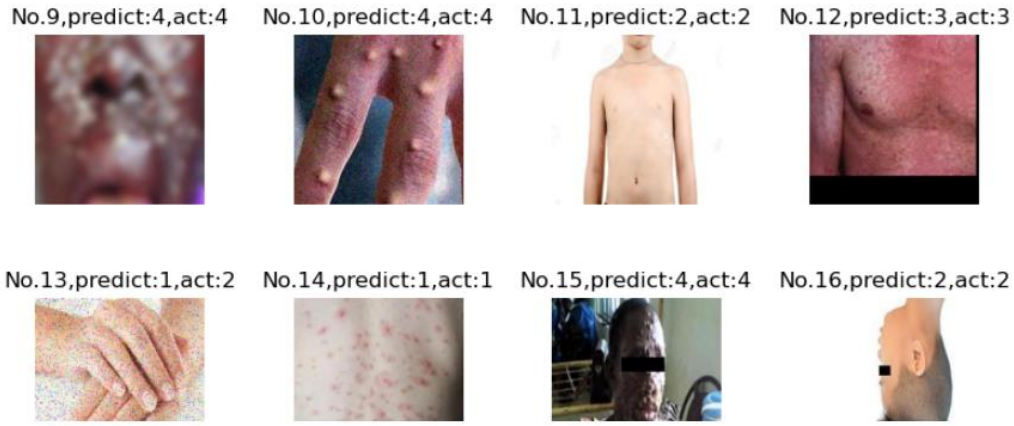


Figure 11. Some examples of PSO-ELM prediction results

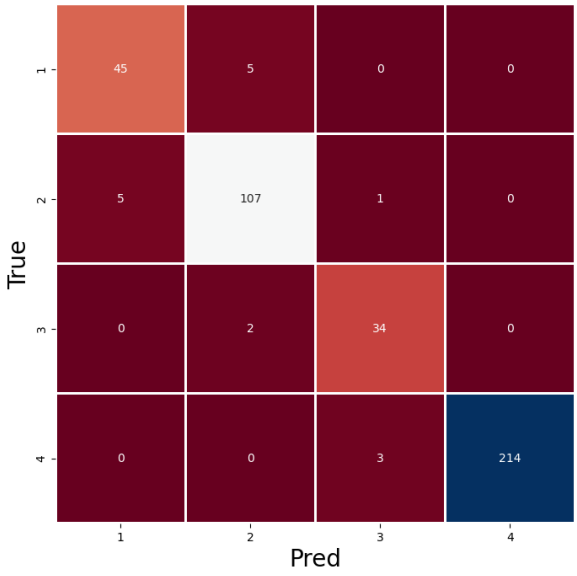


Figure 12. Confusion matrix of the PSO-ELM

The basic ELM model on 50 feature datasets are evaluated using the ten-fold cross-validation method to obtain an average accuracy rate of 0.948 and an average training time of 0.002 seconds, as shown in Table 4.

Table 4. Results of basic ELM model on 50 features

Fold	Accuracy	Precision	Recall	F1-score	Time
1	0.969	0.978	0.969	0.971	0.003
2	0.962	0.976	0.962	0.966	0.002
3	0.957	0.969	0.957	0.959	0.003
4	0.943	0.954	0.943	0.944	0.002
5	0.947	0.965	0.947	0.953	0.002
6	0.940	0.957	0.940	0.942	0.002
7	0.909	0.935	0.909	0.912	0.003
8	0.950	0.958	0.950	0.950	0.003
9	0.945	0.959	0.945	0.948	0.002
10	0.959	0.971	0.959	0.962	0.003
Mean	0.948	0.962	0.948	0.951	0.002
std	0.016	0.013	0.016	0.017	0.001

Table 5 shows the evaluation of the PSO-ELM model on 100 feature datasets using the ten-fold cross-validation method, obtaining an average accuracy of 0.961 and an average training time of 15.922 seconds.

Table 5. Results of PSO-ELM model on 100 features

Fold	Accuracy	Precision	Recall	F1-score	Time
1	0.995	0.995	0.995	0.993	15.411
2	0.971	0.975	0.971	0.973	17.244
3	0.981	0.981	0.981	0.982	16.537
4	0.933	0.955	0.933	0.939	15.325
5	0.962	0.970	0.962	0.966	15.575
6	0.930	0.939	0.930	0.934	15.764
7	0.995	0.995	0.995	0.995	16.115
8	0.964	0.975	0.964	0.969	15.506
9	0.923	0.936	0.923	0.929	15.626
10	0.959	0.967	0.959	0.963	16.112
Mean	0.961	0.969	0.961	0.964	15.922
std	0.026	0.020	0.026	0.024	0.598

The same experimental methods were used for the ELM on 100 features and PSO-ELM on 50 features. The results are shown in Table 6. The best model is PSO-ELM on 100 features, with accuracy, precision, recall and F1 score of 0.961±0.026, 0.969±0.02, 0.961±0.026 and 0.964±0.024 respectively.

Table 6. Comparison of ELM and PSO-ELM

	Accuracy	Precision	Recall	F1	Time(s)
ELM on 50 features	0.948 (0.016)	0.962 (0.013)	0.948 (0.016)	0.951 (0.017)	**0.002** **(0.001)**
ELM on100 features	0.949 (0.015)	0.962 (0.010)	0.949 (0.015)	0.952 (0.015)	0.003 (0.001)
PSO-ELM on 50 features	0.953	0.962	0.953	0.958	15.756

	(0.027)	(0.018)	(0.027)	(0.022)	(0.336)
PSO-ELM on 100 features	**0.961**	**0.969**	**0.961**	**0.964**	15.922
	(0.026)	**(0.020)**	**(0.026)**	**(0.024)**	(0.598)

Table 7 shows the comparison of PSO-ELM with classic CNNs, including Xception, DenseNet121, VGG19, ResNET50, and EffcientNetV2 [5]. Among them, the data shows that PSO-ELM has better performance.

Table 7. Comparison of PSO-ELM and classic CNNs

	Accuracy	Recall	F1
Xception	0.852	0.852	0.866
DenseNet121	0.895	0.895	0.895
VGG19	0.924	0.924	0.946
ResNET50	0.935	0.935	0.958
EffcientNetV2	0.955	0.955	**0.973**
PSO-ELM	**0.961**	**0.961**	0.964

V. CONCLUSION

This study provides valuable insights into the rapid diagnosis and classification of monkeypox and demonstrates the potential of deep learning techniques in medical image analysis. This approach not only improves the accuracy of diagnosis, but also reduces the time and cost required for diagnosis compared to classical deep learning models, which is important for the control and prevention of monkeypox epidemics. Future research can explore more metaheuristic algorithms to improve the performance and efficiency of the model.

Although this study proposes a model architecture for extreme learning machines based on the particle swarm optimization algorithms, there are still many directions that deserve to be investigated and improved. These improvements can effectively help the model to improve accuracy, training time, and applicability for better utilization in areas with weak infrastructure.

ACKNOWLEDGEMENT

We are grateful to the Ministry of Science and Technology of Taiwan for providing a research grant that supported this study (MOST 110-2221-E-415-014).

REFERENCES

[1] M. M. Ahsan, M. R. Uddin, M. Farjana, A. N. Sakib, K. A. Momin, and S. A. Luna, "Image Data collection and implementation of deep learning-based model in detecting Monkeypox disease using modified VGG16," *arXiv* preprint arXiv:2206.01862, 2022.

[2] G. B. Huang, Q. Y. Zhu, and C. K. Siew, "Extreme learning machine: a new learning scheme of feedforward neural networks," In 2004 IEEE International Joint Conference on Neural Networks, vol. 2, pp. 985-990, 2004.

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

[4] C. Sitaula and T. B. Shahi, "Monkeypox virus detection using pre-trained deep learning-based approaches," Journal of Medical Systems, vol. 46, no. 78, 2022.

[5] M. Altun, H. Gürüler, O. Özkaraca, F. Khan, J. Khan, and Y. Lee, "Monkeypox detection using CNN with transfer learning," Sensors, vol. 23, no. 1783, 2023.

[6] D. Bala, M. S. Hossain, M. A. Hossain, M. I. Abdullah, M. M. Rahman, B. Manavalan, and Z. Huang, "MonkeyNet: A robust deep convolutional neural network for monkeypox disease detection and classification," Neural Networks, vol. 161, pp. 757-775, 2023.

[7] A. Arı, O. F. Alcin, and D. Hanbay, "Brain MR image classification based on deep features by using extreme learning machines," Biomedical Journal of Scientific and Technical Research, vol. 25, no3, 2020.

[8] Z. Wang, M. Li, H. Wang, H. Jiang, Y. Yao, H. Zhang, and J. Xin, "Breast cancer detection using extreme learning machine based on feature fusion with CNN deep features," IEEE Access, vol. 7, pp. 105146-105158, 2019.

[9] R. K. Pattnaik, M. Siddique, S. Mishra, D. J. Gelmecha, R. S. Singh, and S. Satapathy, "Breast cancer detection and classification using metaheuristic optimized ensemble extreme learning machine," International Journal of Information Technology, vol. 15, no. 8, pp. 4551-4563, 2023.

[10] Q. Wang, S. Song, L. Li, D. Wen, P. Shan, Z. Li, and Y. Fu, “An extreme learning machine optimized by differential evolution and artificial bee colony for predicting the concentration of whole blood with Fourier Transform Raman spectroscopy,” Spectrochimica Acta Part A: Molecular and Biomolecular Spectroscopy, vol. 292, no. 122423, 2023.

[11] M.A.A. Albadr, S. Tiun, M. Ayob, and F. T. AL-Dhief, “Particle swarm optimization-based extreme learning machine for COVID-19 detection,” Cognitive Computation, vol. 16, pp. 1858–1873, 2024.

[12] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, “Densely connected convolutional networks,” In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4700-4708, 2017.

[13] J Kennedy and R. Eberhart, “Particle swarm optimization,” In Proceedings of ICNN'95-International Conference on Neural Networks, vol. 4, pp. 1942-1948, Nov. 1995.