ITC 2/52
Information Technology and Control
Vol. 52 / No. 2 / 2023
pp. 288-296
DOI 10.5755/j01.itc.52.2.32803

Monkeypox Disease Detection with Pretrained Deep Learning Models

Received 2022/11/18 | Accepted after revision 2022/12/27

HOW TO CITE: Ren, G. (2023). Monkeypox Disease Detection with Pretrained Deep Learning Models. *Information Technology and Control*, 52(2), 288-296. https://doi.org/10.5755/j01.itc.52.2.32803

Monkeypox Disease Detection with Pretrained Deep Learning Models

Guanyu Ren
School of Mechanical, Electrical and Information Engineering, Shandong University, Weihai 264209, China

Corresponding author: guanyuren@mail.sdu.edu.cn

Monkeypox has been recognized as the next global pandemic after Novel coronavirus and its potential damage cannot be neglected. Computer vision-based diagnosis and detection method with deep learning models have been proven effective during the Novel coronavirus period. However, with limited samples, the deep learning models are difficult to be full trained. In this paper, twelve models based on convolutional neural network, including VGG16, VGG19, ResNet152, DenseNet121, DenseNet201, EfficientNetB7, EfficientNetV2B3, EfficientNetV2M and InceptionV3, are used for monkeypox detection with limited skin pictures. Numerical results suggest that DenseNet201 achieves the best classification accuracy of 98.89% for binary classification, 100% for four-class classification and 99.94% for six-class classification over the rest models.

KEYWORDS: Monkeypox, Deep Learning, Convolutional Neural Network.

1. Introduction

In the last three months, we have seen outbreaks of monkeypox in Europe and the United States. The number of confirmed cases has increased at an alarming rate. More than 18,000 cases of monkeypox have now been reported to the WHO (World Health Organization) from 78 countries, with more than 70% of cases reported from the European region and 25% from the Americas [35]. On July 23, 2022, the WHO announced that the global monkeypox outbreak is a public health emergency of international concern [26]. Monkeypox is a viral zoonosis caused by the monkeypox virus. People with monkeypox may develop a rash on their hands, feet, chest, face, or mouth [6]. It may also be located on the genitals or near the anus.

Monkeypox virus is a type of orthopoxvirus that includes camelpox, cowpox, vaccinia, and variola viruses [25]. Clinical differentiation of the disease from smallpox and varicella is difficult. Poorly resourced endemic areas where monkeypox is found present challenges in clinical identification, diagnosis, and prevention. The current techniques used to diagnose monkeypox virus are mainly PCR (Polymerase Chain

Reaction), including real-time PCR. However, these tests require expensive equipment and reagents. It must be done in a major laboratory with skilled technicians [25]. In the case of monkeypox outbreaks such as this, cheaper and more convenient detection methods are in need, such as image-based methods based on smartphones.

Deep learning models have been proven effective for a series of problems, especially those image-based approaches. Therefore, deep learning based on detection techniques become crucial. However, there are only a few cases of using deep learning to detect monkeypox. In these cases, only a few models are used for monkeypox detection, and the accuracy is comparatively low. One of the reasons for these conditions is that there are few pictures of monkeypox patients who can be found on the Internet, so there are also few monkeypox datasets.

To address the limitations of the current study, we paid attention to the improvement in the detection of monkeypox. The purpose of this paper is to detect monkeypox patients from skin pictures of monkeypox and other similar diseases. We selected three datasets, including 2, 4, and 6 classes. Section 4.1 is Dataset Description.

Because the existing monkeypox datasets are small, transfer learning can achieve relatively better detection results. We used twelve pretrained models, VGG16 [29], VGG19 [29], ResNet152 [21], DenseNet121 [10], DenseNet201 [10], EfficientNetB7 [33], EfficientNetV2B3 [33], EfficientNetV2M [33], InceptionV3 [31], InceptionResNetV2 [30], Xception [7] and MobileNetV2 [28], to achieve the best accuracy for the three datasets. The best pretrained CNN (convolutional neural network) model is DenseNet201, which achieves the highest accuracy of 98.89% for binary classification, 100% for four-class classification and 99.94% for six-class classification. Our experimental results are better than other existing studies.

The rest of this paper is organized as follows: In Section 2, we discuss the related work of our work. The methodology of deep transfer learning models is discussed in Section 3. The dataset, experimental settings, the results and discussions are described in Section 4. Finally, the conclusion, limitations, and future works are summarized in Section 5.

2. Related Work

2.1. Medical Image Processing

Traditional image processing is generally based on manual feature extraction techniques, which require the expert knowledge in specific domains. With the accumulation of data samples and the increase of computation resources, machine learning and deep learning methods have replaced the traditional methods in different fields, e.g., computer vision [19, 37], finance [13, 17, 34], transportation [14, 18, 20], communication networks [15, 16, 32], etc. During the COVID-19 (Novel coronavirus) pandemic period, deep learning models have been successfully applied in the diagnosis process [5, 23, 36].

Islam et al. [11] used a concatenation of a convolutional neural network and long short-term memory (LSTM) to detect COVID-19 from chest X-ray images. The experimental results show that their proposed system achieved an accuracy of 99.4%.

Rahimzadeh et al. [27] used a combination of Xception and ResNet50V2 to detect COVID-19 automatically from X-ray images. The average accuracy of the proposed network is 99.50%, and the overall average accuracy for all classes is 91.4%.

Loey et al. [24] used a GAN (Generative Adversarial Networks) with deep transfer learning to detect COVID-19 from chest X-ray images. In this paper, GoogleNet achieves 100% testing accuracy on 2-class classification.

Khan et al. [22] proposed CoroNet, based on Xception, for coronavirus detection in chest X-ray images. Their proposed model achieved an overall accuracy of 89.6% for 4-class cases and a classification accuracy of 95% for 3-class classification.

Hira et al. [1] used nine convolutional neural network-based architectures for coronavirus detection in chest X-ray images. The pretrained model Se-ResNeXt-50 achieves the highest classification accuracy of 99.32% for the binary class and 97.55% for the multiclass among all pretrained models.

2.2. Monkeypox Detection

While deep learning models have proven effective for medical image processing, especially under the COVID-19 pandemic period as discussed in the above

part. However, the application of monkey detection with deep learning is still in an early stage, which is the research motivation of this study.

Ali et al. [4] published a monkeypox image dataset that contained 1428 augmented images of monkeypox patients and 1764 augmented images of other people. They used ResNet50, VGG16 and the ensemble system with accuracies of 82.96 (±4.57%), 81.48 (±6.87%) and 79.26 (±1.05%), respectively.

Ahsan et al. [3] published another image dataset, which contains 1468 augmented images belonging to patients with monkeypox, chickenpox or measles or healthy people. They used VGG16 to detect monkeypox on this dataset. The highest accuracy in the paper was 83%.

Islam et al. [12] published a dataset containing Monkeypox, Chickenpox, Smallpox, Cowpox, and Measles infected skin as well as healthy skin images, 39396 in total.

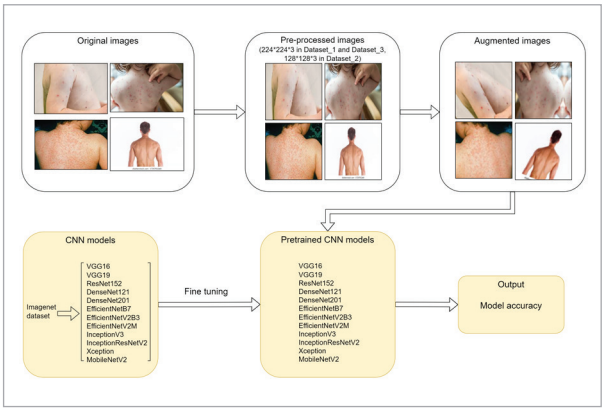
3. Methodology

3.1. Problem Description

The proposed approach of primary screening is encapsulated in Figure 1. The original pictures, including Monkeypox, Chickenpox, Smallpox, Cowpox, and Measles infected skin as well as healthy skin images, are preprocessed and augmented. After that, the pretrained models are further trained with the augmented images to screen the models. Then, we select the best pretrained model according to the accuracy and efficiency.

Figure 1

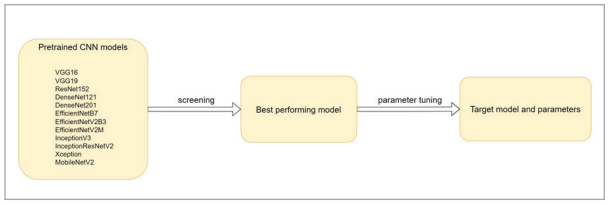
Flow chart for primary screening for monkeypox detection



We attempt to tune various parameters of the preliminary screening model to determine whether it will perform better. Finally, the best parameter combination for this model is obtained. The process of tuning the parameters to achieve the best performance is shown in Figure 2.

Figure 2

Flow chart of tuning the parameters of the best model

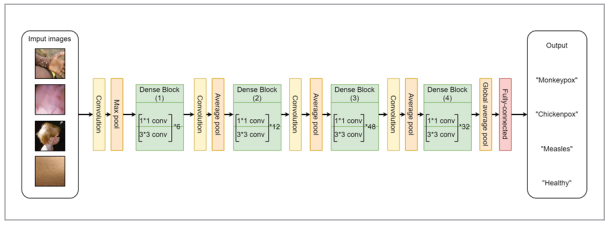


3.2. Model Description

Among the 12 models, the best one is Densenet201 in primary screening. The model structure of Densenet201 [10] is shown in Figure 3. The pictures are labeled by category. After passing through multiple convolution layers, pooling layers and dense blocks [10], they are classified by the fully-connected layer and output labels, and the predicted labels are output.

Figure 3

Structure diagram of Densenet201



4. Experiment

4.1. Dataset Description

Due to the small number of existing monkeypox datasets, the range of options is small. For the purpose of the experiments, we selected the following three datasets. They are all properly classified and preprocessed, which makes it easier for us to train.

First, a dataset [4] including two classes on Kaggle was selected. It contains 1428 augmented images of

monkeypox patients and 1764 augmented images of other people. This dataset is referred to as Dataset_1.

Second, to further evaluate the classification performance, a dataset [2] including four classes (monkeypox, measles, chickenpox and normal images) on Github was selected. A problem is that some pictures were labeled incorrectly but we fixed it. It contains 171 original images (monkeypox: 43, chickenpox: 47, measles: 27, normal: 54) and 1468 augmented images (monkeypox: 301, chickenpox: 329, measles: 286, normal: 552) after our correction. This dataset is referred to as Dataset_2.

Third, a dataset [12] including 6 classes was selected. It contains 804 original images (monkeypox: 117, chickenpox: 178, smallpox: 358, cowpox: 54, measles: 47, health: 50) and 39396 augmented images (monkeypox: 5733, chickenpox: 8722, smallpox: 17542, cowpox: 2646, measles: 2303, health: 2450). This dataset is referred to as Dataset_3.

The URLs of the three datasets are shown in Table 1.

Table 1
URL of each dataset

Dataset	URL
Dataset_1 [4]	https://www.kaggle.com/datasets/nafin59/monkeypox-skin-lesion-dataset
Dataset_2 [2]	https://github.com/mahsan2/Monkeypox-dataset-2022
Dataset_3 [12]	https://www.heywhale.com/mw/dataset/62eb75d6fef0903951b1f199/content

4.2. Experiment Settings

We use the Windows10 system with CUDA11.1, cuDNN 8.1.0, Python 3.9.12, and TensorFlow 2.9.0. All images we used in the three datasets have been augmented. For each dataset, we randomly divided the augmented images into test sets and training sets at a ratio of 1:4.

In terms of choosing an optimizer. SGDM causes slower convergence. Adagrad will make it easy to end training early. Adadelta causes the training to repeatedly shake around the local minimum at a later stage. RMSProp causes parameters to fluctuate too much when updating. However, Adam combines Adagrad's ability to handle sparse gradients and RMSprop's ability to handle non-stationary targets, and enables training to converge quickly. Therefore, we chose Adam as the optimizer in our study.

4.3. Results and Discussions

To achieve the best performance in monkeypox detection, we used these 12 models for prediction on each dataset for primary selection. On each dataset, different models used the same parameters. The parameter settings of the models are shown in Table 2.

Table 2
Parameter settings of models for primary selection

Models	Optimizer	learning rate	Batch size	decay	Dropout rate	No of neurons in the fully connected layer	No of epochs on Dataset_1	No of epochs on Dataset_2	No of epochs on Dataset_3
VGG16	ADAM	0.0001	8	0.001	0.5	256	40	50	10
VGG19	ADAM	0.0001	8	0.001	0.5	256	40	50	10
ResNet152	ADAM	0.0001	8	0.001	0.5	256	40	50	10
DenseNet121	ADAM	0.0001	8	0.001	0.5	256	40	50	10
DenseNet201	ADAM	0.0001	8	0.001	0.5	256	40	50	10
EfficientNetB7	ADAM	0.0001	8	0.001	0.5	256	40	50	10
EfficientNetV2B3	ADAM	0.0001	8	0.001	0.5	256	40	50	10
EfficientNetV2M	ADAM	0.0001	8	0.001	0.5	256	40	50	10
InceptionV3	ADAM	0.0001	8	0.001	0.5	256	40	50	10
InceptionResNetV2	ADAM	0.0001	8	0.001	0.5	256	40	50	10
Xception	ADAM	0.0001	8	0.001	0.5	256	40	50	10
MobileNetV2	ADAM	0.0001	8	0.001	0.5	256	40	50	10

The results for primary screening are shown in Table 3. These accuracies indicate that most models perform well. However, considering the accuracy of the three datasets and the total number of parameters for each model, we can conclude that Densenet201 is the best model in our research. Therefore, we will tune the parameters of DenseNet201 to obtain better performance.

First, we used different decays on the Adam optimizer to observe the changes in model accuracy.

The results of DenseNet201 when using different decays are shown in Table 4. We can see from that setting decay to 0.001 is the best. Therefore, we continue to set decay to 0.001. Then we used different dropout rates to observe the changes in model accuracy.

The results of DenseNet201 when using different dropout rates are shown in Table 5. There are slight differences in the accuracy of the approaching settings. However, we can still find that setting the dropout rate to 0.5 is the best. Therefore, we continue to set the dropout rate to 0.5.

Then, we used the different numbers of neurons in the fully connected layer to observe the changes in model accuracy.

Table 3
The accuracy on each dataset for primary selection

Models	Total No of parameters	Accuracy on Dataset_1	Accuracy on Dataset_2	Accuracy on Dataset_3	Average accuracy
VGG16	21M	0.9509	0.9688	0.9957	0.9718
VGG19	26M	0.9525	0.9792	0.9766	0.9694
ResNet152	84M	0.9763	0.9722	0.9982	0.9822
DenseNet121	20M	0.9652	0.9931	0.9972	0.9852
DenseNet201	42M	0.9810	1.0000	0.9986	0.9932
EfficientNetB7	96M	0.9842	0.9722	0.9985	0.9850
EfficientNetV2B3	32M	0.9494	0.9757	0.9953	0.9735
EfficientNetV2M	69M	0.9620	0.9757	0.9990	0.9789
InceptionV3	35M	0.9794	0.9931	0.9958	0.9894
InceptionResNetV2	64M	0.9715	0.9583	0.9709	0.9669
Xception	47M	0.9842	0.9722	0.9977	0.9847
MobileNetV2	18M	0.9430	0.9861	0.9876	0.9722

Table 4
The accuracy on each dataset when different decays are used (Model: DenseNet201, learning rate: 0.0001, batch size: 8, dropout rate: 0.5, number of neurons in the fully connected layer: 256)

Decay	Accuracy on Dataset_1	Accuracy on Dataset_2	Accuracy on Dataset_3	Average accuracy
0.1	0.9383	0.9514	0.8112	0.9003
0.01	0.9826	0.9965	0.9924	0.9905
0.005	0.9842	0.9931	0.9967	0.9913
0.003	0.9842	0.9965	0.9973	0.9927
0.001	0.9810	1.0000	0.9986	0.9932
0.0001	0.9794	0.9965	0.9972	0.9910
0.00001	0.9525	0.9757	0.9947	0.9743

Table 6 shows the results when using the different numbers of neurons in the fully connected layer. Setting the numbers of neurons in the fully connected layer to 128 performs best on each dataset. It is better than the original setting, which sets the numbers of neurons in the fully connected layer to 256. Therefore, we set that parameter to 128. Then we used the different numbers of epochs on each dataset to achieve the highest accuracy.

Table 7 shows the results when using the different numbers of epochs on each dataset. Considering both

Table 5

The accuracy on each dataset when different dropout rates are used (Model: DenseNet201, learning rate: 0.0001, batch size: 8, decay: 0.001, number of neurons in the fully connected layer: 256)

Dropout Rate	Accuracy on Dataset_1	Accuracy on Dataset_2	Accuracy on Dataset_3	Average accuracy
0.3	0.9794	0.9896	0.9982	0.9891
0.4	0.9778	0.9965	0.9991	0.9911
0.5	0.9842	1.0000	0.9986	0.9943
0.6	0.9810	1.0000	0.9985	0.9932
0.7	0.9873	0.9931	0.9977	0.9927

Table 6

The accuracy on each dataset when the numbers of neurons in the fully connected layer are different (Model: DenseNet201, learning rate: 0.0001, batch size: 8, decay: 0.001, dropout rate: 0.5)

The numbers of neurons in the fully connected layer	Accuracy on Dataset_1	Accuracy on Dataset_2	Accuracy on Dataset_3	Average accuracy
1024	0.9794	0.9931	0.9980	0.9902
512	0.9858	1.0000	0.9980	0.9946
256	0.9810	1.0000	0.9986	0.9932
128	0.9889	1.0000	0.9990	0.9960
64	0.9858	0.9931	0.9976	0.9922
32	0.9810	0.9965	0.9981	0.9919
16	0.9778	0.9931	0.7160	0.8956

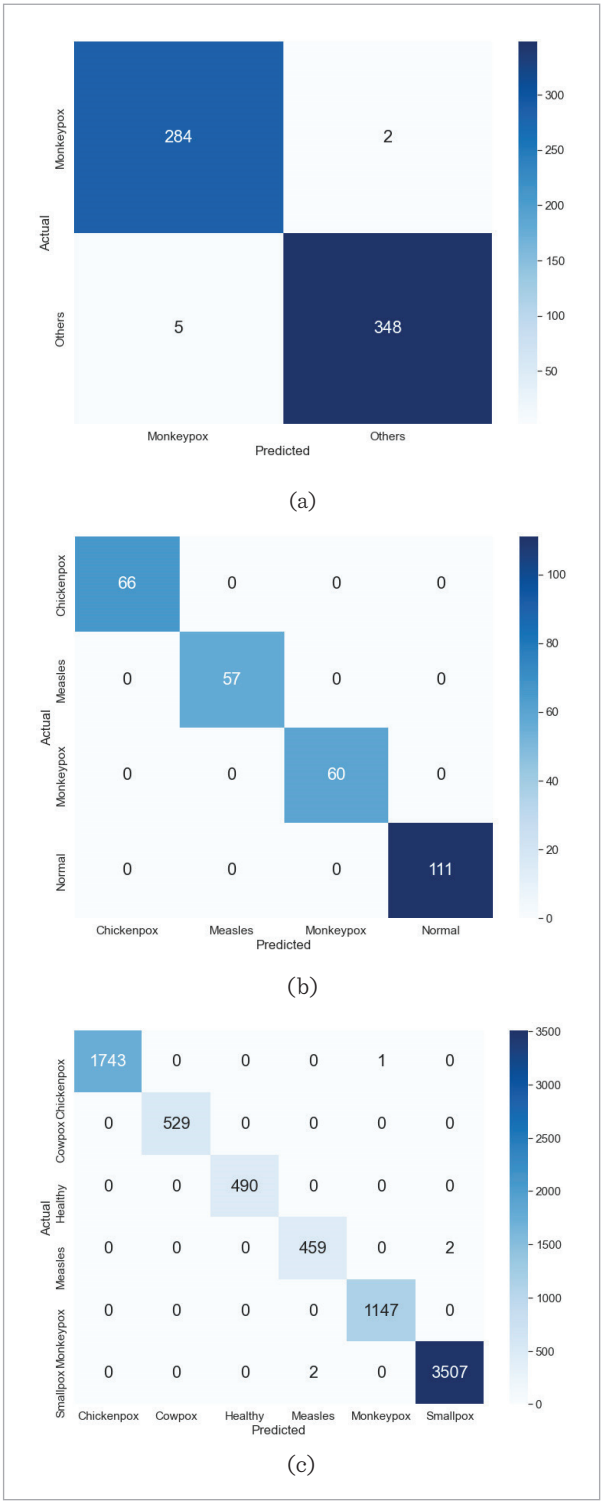
Table 7

The accuracy on each dataset when the numbers of epochs are different (Model: DenseNet201, learning rate: 0.0001, batch size: 8, decay: 0.001, dropout rate: 0.5, number of neurons in the fully connected layer: 128)

Dataset	No of epochs	Accuracy
Dataset_1	10	0.9684
	20	0.9826
	30	0.9778
	40	0.9889
	50	0.9763
Dataset_2	30	0.9896
	40	1.0000
	50	1.0000

Dataset	No of epochs	Accuracy
Dataset_2	60	0.9965
	70	1.0000
Dataset_3	5	0.9976
	10	0.9990
	15	0.9986
	20	0.9994
	25	0.9991
	30	0.9989

Figure 4
Confusion matrices for DenseNet201 on each dataset: (a) Dataset_1, (b) Dataset_2, (c) Dataset_3



accuracy and efficiency, we can deduce that DenseNet201 performs best when setting the numbers of epochs to 40 on Dataset_1 and Dataset_2 and 20 on Dataset_3. The confusion matrices in these parameter settings are shown in Figure 4.

5. Discussion

We used DenseNet201 with classification accuracy of 98.89% for binary classification. Ali et al. [4] used the ensemble system with accuracies of 79.26% (±1.05%) on the same dataset. We used DenseNet201 with classification accuracy of 100% for four-class classification. Ahsan et al. [3] used VGG16 to detect monkeypox on this dataset. The highest accuracy in the paper was 83% on the same dataset with us. Our results are better than Ali et al. [4] and Ahsan et al. [3].

6. Conclusion

The results in this study prove that transfer deep learning may have a significant impact on the automatic detection of skin images, which are related to the diagnosis of monkeypox.

In this paper, we used twelve CNN-based models, including VGG16, VGG19, ResNet152, DenseNet121, DenseNet201, EfficientNetB7, EfficientNetV2B3, EfficientNetV2M, and InceptionV3, to differentiate monkeypox patients and people with other diseases or healthy people from skin pictures. The results suggest that DenseNet201 achieves the best classification accuracy of 98.89% for binary classification, 100% for four-class classification and 99.94% for six-class classification over the rest models.

A more in-depth discussion requires much more image data. The current challenge is that the datasets are small. The future goal is to collect a larger dataset to improve the monkeypox detection ability. With a larger dataset in the future, the CNN model may perform worse, so we need to further explore more and more powerful models, such as Transformer [8] and ViG [9].

Data Sharing Agreement

The datasets used and/or analyzed during the current study are available from the corresponding author on reasonable request.

Declaration of Conflicting Interests

The author(s) declared no potential conflicts of interest with respect to the research, author-ship, and/or publication of this article.

Funding

The author(s) received no financial support for the research, authorship, and/or publication of this article.

References

1. Abbas, A., Abdelsamea, M. M., Gaber, M. M. Classification of COVID-19 in Chest X-ray Images Using DeTraC Deep Convolutional Neural Network. Applied Intelligence, 2021, 51(2), 854-864. https://doi.org/10.1007/s10489-020-01829-7
2. Ahsan, M. M., Uddin, M. R., Luna, S. A. Monkeypox Image Data Collection. ArXiv preprint, 2022. arXiv:2206.01774.
3. Ahsan, M. M., Uddin, M. R., Farjana, M., Sakib, A. N., Momin, K. A., Luna, S. A. Image Data Collection and Implementation of Deep Learning-based Model in Detecting Monkeypox Disease Using Modified VGG16. ArXiv preprint, 2022. arXiv:2206.01862.
4. Ali, S. N., Ahmed, M., Paul, J., Jahan, T., Sani, S. M., Noor, N., Hasan, T. Monkeypox Skin Lesion Detection Using Deep Learning Models: A Feasibility Study. ArXiv preprint, 2022. arXiv:2207.03342.
5. Alsunaidi, S. J., Almuhaideb, A. M., Ibrahim, N. M., Shaikh, F. S., Alqudaihi, K. S., Alhaidari, F. A., Khan, I. U., Aslam, N., Alshahrani, M. S. Applications of Big Data Analytics to Control COVID-19 Pandemic. Sensors, 2021, 21(7), 2282. https://doi.org/10.3390/s21072282
6. Centers for Disease Control and Prevention. (2022) Monkeypox signs and symptoms. Online: https://www.cdc.gov/poxvirus/monkeypox/symptoms/index.html. Accessed on 2022/9/5.
7. Chollet, F. Xception: Deep Learning with Depthwise Separable Convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, 1251-1258. https://doi.org/10.1109/CVPR.2017.195
8. Han, K., Wang, Y., Chen, H., Chen, X., Guo, J., Liu, Z., Tang, Y., Xiao, A., Xu, C., Xu, Y., Yang, Z., Zhang, Y., Tao, D. A Survey on Vision Transformer. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2023, 45(1), 87-110. https://doi.org/10.1109/TPAMI.2022.3152247
9. Han, K., Wang, Y., Guo, J., Tang, Y., Wu, E. Vision GNN: An Image is Worth Graph of Nodes. ArXiv preprint, 2022. arXiv:2206.00272.
10. Huang, G., Liu, Z., Van Der Maaten, L., Weinberger, K. Q. Densely Connected Convolutional Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, 4700-4708. https://doi.org/10.1109/CVPR.2017.243
11. Islam, M. Z., Islam, M. M., Asraf, A. A Combined Deep CNN-LSTM Network for the Detection of Novel Coronavirus (COVID-19) Using X-ray Images. Informatics in Medicine Unlocked, 2020, 20, 100412. https://doi.org/10.1016/j.imu.2020.100412
12. Islam, T., Hussain, M. A., Chowdhury, F. U. H., Islam, B. R. A Web-scrapped Skin Image Database of Monkeypox, Chickenpox, Smallpox, Cowpox, and Measles. bioRxiv 2022.08.01.502199. https://doi.org/10.1101/2022.08.01.502199
13. Jiang, W. Applications of Deep Learning in Stock Market Prediction: Recent Progress. Expert Systems with Applications, 2021, 184, 115537. https://doi.org/10.1016/j.eswa.2021.115537
14. Jiang, W. Bike Sharing Usage Prediction with Deep Learning: A Survey. Neural Computing and Applications, 2022, 15369-15385. https://doi.org/10.1007/s00521-022-07380-5
15. Jiang, W. Cellular Traffic Prediction with Machine Learning: A Survey. Expert Systems with Applications, 2022, 117163. https://doi.org/10.1016/j.eswa.2022.117163
16. Jiang, W. Graph-based Deep Learning for Communication Networks: A Survey. Computer Communications, 2022, 185, 40-54. https://doi.org/10.1016/j.comcom.2021.12.015
17. Jiang, W. Loan Default Prediction with Deep Learning and Muddling Label Regularization. IEICE Transactions on Information and Systems, 2022, 105(7), 1340-1342. https://doi.org/10.1587/transinf.2022EDL8003
18. Jiang, W., Luo, J. Graph Neural Network for Traffic Forecasting: A Survey. Expert Systems with Applications, 2022, 117921. https://doi.org/10.1016/j.eswa.2022.117921
19. Jiang, W., Zhang, L. Edge-siamnet and Edge-triplenet: New Deep Learning Models for Handwritten Numeral Recognition. IEICE Transactions on Information and Systems, 2020, 103(3), 720-723. https://doi.org/10.1587/transinf.2019EDL8199

20. Jiang, W., Zhang, L. Geospatial Data to Images: A Deep-learning Framework for Traffic Forecasting. Tsinghua Science and Technology, 2018, 24(1), 52-64. https://doi.org/10.26599/TST.2018.9010033

21. Kaiming, H., Zhang, X., Ren, S., Sun, J. Deep Residual Learning for Image Recognition. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, 770-778. https://doi.org/10.1109/CVPR.2016.90

22. Khan, A. I., Shah, J. L., Bhat, M. M. CoroNet: A Deep Neural Network for Detection and Diagnosis of COVID-19 from Chest X-ray Images. Computer Methods and Programs in Biomedicine, 2020, 196, 105581. https://doi.org/10.1016/j.cmpb.2020.105581

23. Lo, S. H., Yin, Y. An Interaction-Based Convolutional Neural Network (ICNN) Toward a Better Understanding of COVID-19 X-ray Images. Algorithms, 2021, 14(11), 337. https://doi.org/10.3390/a14110337

24. Loey, M., Smarandache, F., M. Khalifa, N. E. Within the Lack of Chest COVID-19 X-ray Dataset: A Novel Detection Model Based on GAN and Deep Transfer Learning. Symmetry, 2020, 12(4), 651. https://doi.org/10.3390/sym12040651

25. McCollum, A. M., Damon, I. K. Human Monkeypox. Clinical Infectious Diseases, 2014, 58(2), 260-267. https://doi.org/10.1093/cid/cit703

26. Monkeypox Virtual Press Conference Transcript - 23 July 2022. Online: https://www.who.int/publications/m/item/monkeypox-virtual-press-conference-transcript---23-july-2022. Accessed on 2022/9/5.

27. Rahimzadeh, M., Attar, A. A Modified Deep Convolutional Neural Network for Detecting COVID-19 and Pneumonia from Chest X-ray Images Based on the Concatenation of Xception and ResNet50V2. Informatics in Medicine Unlocked, 2020, 19, 100360. https://doi.org/10.1016/j.imu.2020.100360

28. Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L. C. Mobilenetv2: Inverted Residuals and Linear Bottlenecks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, 4510-4520. https://doi.org/10.1109/CVPR.2018.00474

29. Simonyan, K., Zisserman, A. Very Deep Convolutional Networks for Large-scale Image Recognition. ArXiv preprint, 2014. arXiv:1409.1556.

30. Szegedy, C., Ioffe, S., Vanhoucke, V., Alemi, A. A. Inception-v4, Inception-resnet and the Impact of Residual Connections on Learning. In Thirty-first AAAI Conference on Artificial Intelligence, 2017, 4278-4284. https://doi.org/10.1609/aaai.v31i1.11231

31. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z. Rethinking the Inception Architecture for Computer Vision. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, 2818-2826. https://doi.org/10.1109/CVPR.2016.308

32. Tan, J., Guan, W. Resource Allocation of Fog Radio Access Network Based on Deep Reinforcement Learning. Engineering Reports, 2022, 4(5), e12497. https://doi.org/10.1002/eng2.12497

33. Tan, M., Le, Q. Efficientnet: Rethinking Model Scaling for Convolutional Neural Networks. In International Conference on Machine Learning, 2019, 6105-6114.

34. Wang, Q., Jiang, W. Beat Wash-Sale Tax with Multi-graph Convolutional Neural Networks Based Trading Strategy. Security and Communication Networks, 2022, 3598285. https://doi.org/10.1155/2022/3598285

35. WHO Press Conference on Monkeypox, COVID-19 and Other Global Health Issues - 27 July 2022. Online: https://www.who.int/multi-media/details/who-press-conference-on-monkeypox--covid-19-and-other-global-health-issues---27-july-2022. Accessed on 2022/9/5.

36. Yang, D., Yurtsever, E., Renganathan, V., Redmill, K. A., Özgüner, Ü. A Vision-based Social Distancing and Critical Density Detection System for COVID-19. Sensors, 2021, 21(13), 4608. https://doi.org/10.3390/s21134608

37. Zheng, Y., Jiang, W. Evaluation of Vision Transformers for Traffic Sign Classification. Wireless Communications and Mobile Computing, 2022, 3041117. https://doi.org/10.1155/2022/3041117