A HYBRID REXCEPTION NETWORK FOR COVID-19 CLASSIFICATION FROM CHEST X-RAY IMAGES

Nour Aburaed[*], Mina Al-Saad[†], Alavikunhu Panthakkan[‡], Saeed Al Mansoori[§], Hussain Al-Ahmad[¶] and Stephen Marshall[‖]
[*†‡¶]College of Engineering and IT, University of Dubai, UAE
[§] Mohammed Bin Rashid Space Centre, UAE
[*‖] Department of Electronic and Electrical Engineering, University of Strathclyde, UK
Email: [*]nour.aburaed,[†]minaalsaad@ieee.org

***Abstract*—Nowadays, with the rapid spread of Coronavirus disease (COVID-19) across the globe, the necessity to develop an intelligent system for early diagnosis and detection the COVID-19 infectious disease increases. In recent researches, Chest X-ray (CXR) of individual lungs became a common method to identify COVID-19 virus. Manual interpretation of the CXR images can be a lengthy process and subjective to human errors. In this paper, a hybrid Deep Learning model called ReXception is implemented, trained, and evaluated using two types of datasets; Mutliclass and Binary. The network is evaluated based on its overall accuracy, loss, precision, and recall, in addition to the running time and network size. The results show positive indications of the network's performance, especially when compared to its original constituent networks.**

***Index Terms*—COVID-19, Chest X-ray, Deep Learning, Classification, Convolutional Neural Network**

I. Introduction

In December 2019, COVID-19, an infectious disease that is mainly caused by Severe Acute Respiratory Syndrome Corona Virus 2 (SARS-CoV-2), was initially identified in Wuhan, China [1]. According to the World Health Organization (WHO) reports on the 12th of August 2021, this newly known disease has impacted the whole world with 205,338,159 confirmed COVID-19 cases, and 4,333,094 deaths [2]. One of the traditional methods for detecting and diagnosing COVID-19 infection is based on performing genetic testing that is known as Reverse Transcription Polymerase Chain Reaction (RT-PCR). This method is considered costly and time-consuming [3]. Another example of traditional methods is Lateral Flow Testing (LFT), which aims to detect particular proteins found in COVID-19 virus. However, this method is not reliable [4]. Therefore, different research studies have discussed the possibility of utilizing radiological imaging techniques, including Chest X-ray (CXR) and Computed Tomography (CT) scan, to identify COVID-19 disease. The manual interpretation of these images is considered both time-consuming and subjective to human error [5], [6]. Nowadays, Artificial Intelligence (AI), especially Deep Learning techniques, play a vital role in the analysis of CXR and CT scan images with less human intervention [7]–[10]. Particularly, Deep Convolutional Neural Networks (DCNNs) have proven efficiency in this field of research. Some of the most prominent DCNNs that have been utilized for the purpose of classifying COVID-19 infected lungs from CXR and CT scans include ResNet-50 and Extreme version of Inception (Xcpetion) and their extensions. For example, in [11], the authors utilize pre-trained ResNet-50 over ImageNet dataset, and introduce modifications to its architecture for the purpose of classifying patients as COVID-19 infected or not. These modifications include adding 3 extra layers to boost the robustness of the classification, which are convolution, batch normalization, and finally a ReLU activation function. Another example of utilizing ResNet-50 can be seen in [12]. The proposed framework consists of three modules; the first one is ResNet-50 with transfer learning, the second one, is a pool of frequency and texture-based hand-picked features that are further reduced using PCA, and the last one is a Feed Forward Neural Network that concatenates the features obtained from first and second modules. Finally, these features are passed to a dense layer followed by a softmax activation function that outputs the classification result.

It has been reported in [13] that Xception's performance on COVID-19 CXR images is superior compared to Inception [14] and ResNet [15]. Therefore, some studies extend Xception in order to further improve its classification performance. In [16], the authors use integrated stack technique to combine Xception with various other DCNNs, such as ResNet-101, Inception-v3, MobileNetv2, and NASNet. The final classification is performed by combining the results from all networks. Another example that utilizes Xception integration can be seen in [17], where the authors stack Xception with ResNet-50v2. Even though ResNet-50 and Xception show superior performances, they suffer from major drawbacks, especially when integrated together with other techniques. As the size of the network grows larger, its memory footprint grows larger as well and it takes a long period to train that can extend to weeks, which is not practical for real-life applications. Additionally, the classification of COVID-19 is a crucial process; false positive and false negative cases must be avoided. Rather than stacking two DCNNs together, the aforementioned drawbacks can be avoided by utilizing different components from DCNNs into one hybrid network. The main contribution of this paper is developing a new hybrid DCNN model for COVID-19 classification called ReXception. This hybrid DCNN is created by fusing Xception and ResNet-50 components in a way that boosts the performance without compromising execution time or memory footprint. ReXception is trained, tested, and evaluated using two different datasets, and the performance is assessed based on the overall accuracy, loss, precision, recall, and sensitivity at specificity, in addition to the running time and network size. The rest of the paper is organized as follows: Section II illustrates the datasets used to train and test the network. Section III discusses ReXception and the various components used to implement it. Section IV demonstrates and analyzes the results. Finally, Section V summarizes and concludes the paper.

II. Dataset and Pre-processing

For this study, two datasets are used for training and testing purposes. The first one, called Multi-class dataset, is collected from two different sources [18], [19]. It contains CXR images of varying sizes that are labeled by radiologists into three classes: 1000 COVID-19, 1000 pneumonia, and 1000 Normal. There are equal samples of each category in order to avoid data bias. Samples of the dataset can be seen in Figure 1. The second dataset is COVID-Xray-5k dataset provided by Minaee et al. [20], which will be referred to as Binary dataset throughout this paper. This dataset contains images of two classes: COVID-19 and Non-COVID-19, as shown in Figure 2. The dataset suffers from high imbalance, as it originally has 84 COVID-19 images and 2000 Non-COVID-19 images of varying sizes. Therefore, data augmentation is applied on COVID-19 classes in order to avoid any potential data bias or overfitting. The augmentation was done by randomly applying combinations of zoom, rotation, horizontal and vertical flips, brightness adjustment, and contrast adjustment operations in various degrees over 84 images to produce 1916 augmented images. After augmentation, both classes in the dataset have 2000 images. Table I summarizes both datasets.

TABLE I
Datasets description summary.

Description	Multi-class dataset	Binary dataset
Training	1920	2560
Validation	480	640
Test	600	800
Num. of classes	3	2
Augmentation	No	Yes

III. Algorithm

This study combines components from two powerful networks to produce a hybrid network capable of efficiently classifying COVID-19 CT scans. The first network is Xception, and the second one is ResNet-50. Both networks utilize skip connections, a vital feature that boosts the network's accuracy and ability to learn at deeper levels. The next subsections provide an overview of Xception, ResNet-50, and the proposed ReXception Network.

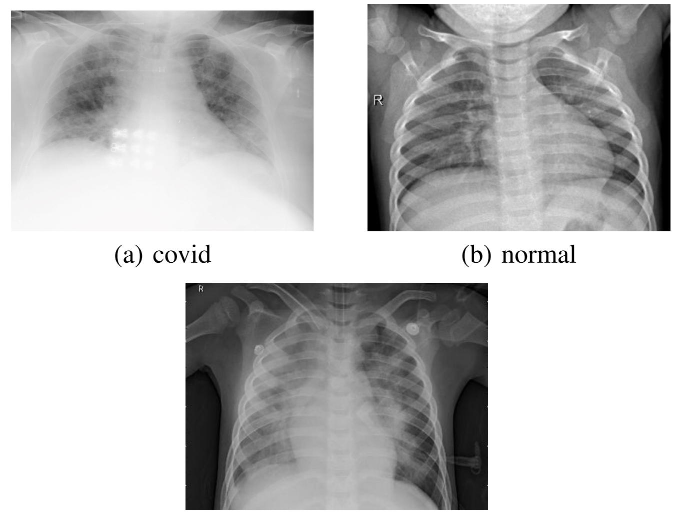
(a) covid (b) normal (c) pneumonia

Fig. 1. Samples from Multiclass dataset

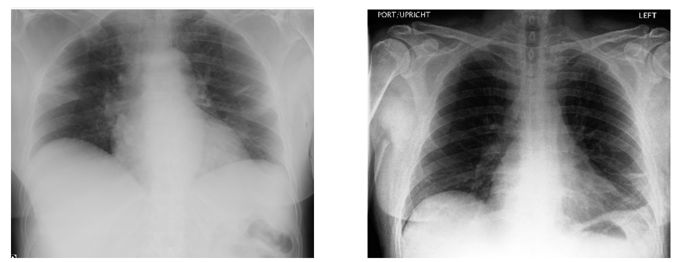
(a) covid (b) noncovid

Fig. 2. Samples from Binary dataset.

A. *Xception*

Xception network was engineered by Google [21] with the goal of enhancing the performance of Inception-v3. The network takes advantage of Depthwise Separable Convolution and adapts it to Inception model. Xception performs a 1×1 convolution, followed by channel-wise spatial convolution. Additionally, there is no intermediate non-linearity in Xception due to the absence of ReLu activation functions. The Depthwise Separable Convolution consists of three flows: Entry, Middle, and Exit. Each flow is treated as an Inception model. Additionally, all the flows contain skip connections, which were originally proposed in ResNet-50, as will be discussed in the next subsection. For the purpose of this study, Middle Flow is of main importance. It consists of a sequence of ReLu activation function and Separable Convolution blocks, as seen in Figure 3. Separable Convolution is more powerful than spatial convolution because it can achieve the same effects within a fewer number of operations, which gives the network the ability to run faster.

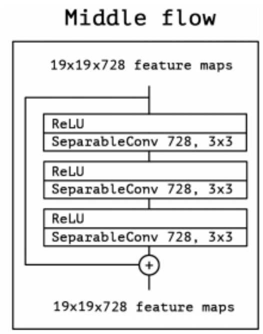

Fig. 3. Middle flow layer of Xception.

B. ResNet-50

ResNet [22] is a well-known network that won ImageNet challenge in 2015. Afterwards, it was used as a backbone in many other CNNs for various image processing tasks. ResNet-50 has 50 layers, and it is a smaller version of ResNet-152. The main strength of ResNet-50 lies in skip connections, where this concept was first introduced. A skip connection means adding the original input to the output of a convolutional block. This reduces vanishing gradient problem by allowing easier flow of information from the top layers to the bottom layers, which in turn eases the process of training deep networks. The general architecture of ResNet-50 consists of five stages, as shown in Figure 4. The most important components of the network's architecture are the convolutional blocks and the identity blocks, as they contain skip connections. The identity block has no convolutional layer and the purpose of using this block in ResNet is to match the input and output dimensions. On the other hand, the convolutional block has convolutional layer at its shortcut path and, in that case, the input is resized to a different dimension.

C. ReXception

The proposed network is a hybrid version of Xception and ResNet-50. First of all, since max pooling layers cause losses in spatial information, and often lead to false maximums, they are removed from ResNet-50. Afterwards, Stage 3 is removed and replaced with the Middle Flow from Xception. The Middle Flow consists only of a series of ReLu function and separable convolution without any max or average pooling, hence, it is chosen rather than Entry flow or Exit flow. Additionally, the shape and size of the feature maps from the Middle Flow match that of ResNet-50, therefore, it fits into the architecture seamlessly. Multiple Middle Flows can be stacked back-to-back depending on the performance of the network on a given dataset. The number of Middle Flow stages can be considered as a hyperparameter. For the purpose of this study, only one Middle Flow is used. This hybrid network combines two types of skip connections used in both ResNet-50 and Xception. A Dropout(0.5) layer can be added between the final Flatten and Fully Connected layers in the event of overfitting, which may occur if the dataset has a limited number of samples. Figure 5 shows the overall architecture of ReXception, and Table II summarizes the training parameters of the network.

TABLE II
TRAINING PARAMETERS.

Parameter	Value
Epochs	100
Learning rate	0.001
Optimization function	Adam
Batch size	32
Shuffle	true
Loss	Categorical cross entropy
Num. of Middle Flows	1

IV. RESULTS AND ANALYSIS

ReXception network is tested on the datasets described in Section II. For both datasets, 80% of the data is used for training and the remaining is used for testing. Of this 80%, 20% is used for validation. The same training parameters are set for both datasets, which were chosen empirically and are listed in Table II. The model was developed and trained using Python Keras (2.3.1) library with Tensorflow backend using NVIDIA Quadro P6000-24GB GPU and Intel(R) 12 core Processor CPU with 380GB RAM.

A. Multi-class Dataset

For the Multi-class dataset, the network achieves a testing accuracy of 99.0% and a loss of 0.092 after 100 epochs. Figure 6 shows that the accuracy of the network becomes relatively more stable after epoch 80, and the accuracy fluctuates approximately between 97.0% and 99.0% towards the end of the training. The results are summarized in Table VI along with the loss, recall, precision, sensitivity at specificity, and training time measured in seconds per step. The confusion matrix for the Multi-class dataset is shown in Table IV. It can be observed from there that the number of False Positive (FP) cases is only 1, while the number of False Negative (FN) cases is 0. Considering the size of the dataset, this is a satisfactory result and a positive indication of the performance of ReXception.

TABLE IV
CONFUSION MATRIX OF THE MULTI-CLASS DATASET.

		Predicted		
		Covid	Pneumonia	Normal
Actual	Covid	194	0	0
	Pneumonia	0	199	5
	Normal	1	10	191

B. Binary Dataset

For the Binary dataset, the network achieves a testing accuracy of 99% and a loss 0.015 after 100 epochs. The results are summarized in Table VI along with the loss, recall, precision, sensitivity at specificity, and training time measured in seconds per step. The training and validation accuracy progression throughout 100 epochs is illustrated in Figure 7, which shows that the accuracy keeps fluctuating in a similar

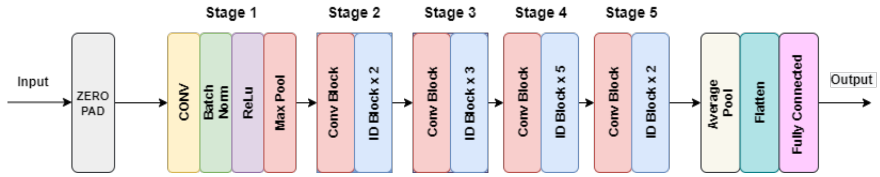

Fig. 4. General architecture of ResNet-50.

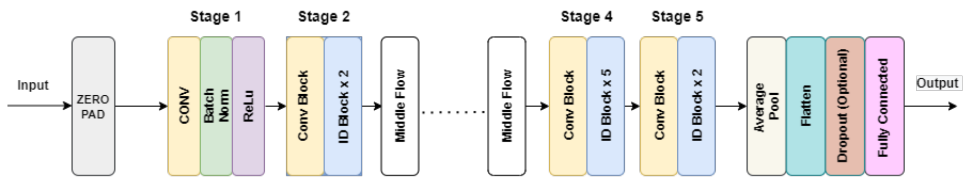

Fig. 5. General architecture of ReXception.

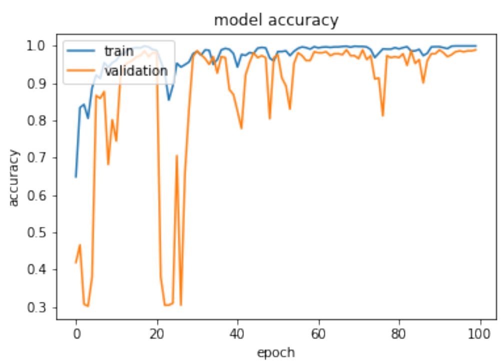

Fig. 6. Accuracy of ReXception performance on the Multi-class dataset over 100 epochs.

manner shown with the Multi-class dataset. These fluctuations diminish gradually as the training epochs approach 100. The confusion matrix for 100 epochs seen in Table V shows that the numbers of FP and FN cases are 3 and 0, respectively. It is worth mentioning that FN cases being 0 for both binary and Multi-class datasets is a reassuring indicator that mitigates potential harm.

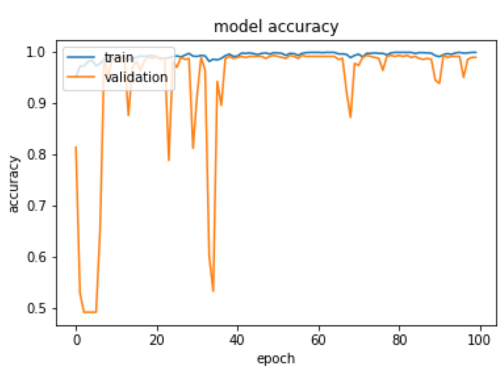

Fig. 7. Accuracy of ReXception performance on the Binary dataset over 100 epochs.

TABLE V
CONFUSION MATRIX OF BINARY DATASET (100 EPOCHS).

		Predicted	
		Covid	Non-Covid
Actual	Covid	411	0
	Non-Covid	3	386

C. Comparison

The results of both Multi-class and Binary datasets show that the network has compelling performance. In comparison to the original Xception and ResNet-50, ReXception shows superior accuracy, loss, recall, precision, and sensitivity at specificity, as seen in Table VI. Furthermore, the total number of scalars composing the weights for ResNet-50, Xception, and ReXception are 23591810, 20867627, and 21922947, respectively. Hence, ReXception network size represents a middle ground between ResNet-50 and Xception. Furthermore, the average time taken to execute each step for Rexcption is only 4ms, whereas for ResNet-50 and Xception, it is 2s and 918ms, respectively. Therefore, ReXception is more computationally efficient than its constituent networks while performing better than both in terms of several evaluation metrics.

V. CONCLUSION

In this research, ReXception was developed as a hybrid product of ResNet-50 and Xception for automatically detecting COVID-19 infection from CXR images. The proposed model is trained and tested using a Multi-class dataset and a Binary dataset. The testing results show promising outcomes in terms of the overall accuracy, loss, recall, precision, and sensitivity at specificity. Additionally, no FN cases were observed from both datasets. Comparisons against the original ResNet-50 and Xception in terms of evaluation metrics, training time, and network size prove the superiority and efficiency of ReXception model. In the future, the proposed methodology can be tested on the new COVID-19 strain datasets.

TABLE VI
RESULTS SUMMARY AND COMPARISON FOR MILT-CLASS AND BINARY DATASETS.

Dataset	Evaluation Metrics	ResNet-50	ReXception	Xception
Multi-class	Accuracy(%)	87.8	**99.0**	98.3
	Loss	0.537	**0.092**	0.117
	Recall(%)	87.8	**99.0**	98.3
	Precision(%)	87.8	**99.0**	98.7
	Sensitivity at Specificity (%)	95.1	**99.0**	**99.0**
	Avg. Sec\Step	2s	**4ms**	918ms
Binary	Accuracy(%)	95.3	**99.0**	98.1
	Loss	0.136	**0.133**	0.338
	Recall(%)	95.3	**99.0**	98.1
	Precision(%)	95.3	**99.0**	98.1
	Sensitivity at Specificity (%)	99.0	**99.2**	98.8
	Avg. Sec\Step	2s	**4ms**	918ms

REFERENCES

[1] A. Bagula, H. Maluleke, O. Ajayi, et al., "Predictive models for mitigating covid-19 outbreak," in *Symposium on Computers and Communications (ISCC)*. IEEE, 2020, pp. 1–7.

[2] WHO, "Who coronavirus disease (covid-19) dashboard," 2021 (accessed January 12 , 2021).

[3] M. E. H. Chowdhury, T. Rahman, A. Khandakar, et al., "Can ai help in screening viral and covid-19 pneumonia?," *IEEE Access*, vol. 8, pp. 132665–132676, 2020.

[4] J. Robinson, "How reliable are lateral flow covid-19 tests?," May 2021.

[5] D. Javor, H. Kaplan, A. Kaplan, et al., "Deep learning analysis provides accurate covid-19 diagnosis on chest computed tomography," *European journal of radiology*, vol. 133, 2020.

[6] V. Perumal, V. Narayanan, and S. J. S. Rajasekar, "Detection of covid-19 using cxr and ct images using transfer learning and haralick features," *Applied Intelligence*, vol. 51, pp. 341–358, 2021.

[7] F. Shi, J. Wang, J. Shi, et al., "Review of artificial intelligence techniques in imaging data acquisition, segmentation and diagnosis for covid-19," *IEEE Reviews in Biomedical Engineering*, pp. 1–1, 2020.

[8] A. Panthakkan, S. M. Anzar, S. Al Mansoori, et al., "Accurate prediction of covid-19 (+) using ai deep vgg16 model," in *2020 3rd International Conference on Signal Processing and Information Security (ICSPIS)*, 2020, pp. 1–4.

[9] A. Panthakkan, S. M. Anzar, S. Al Mansoori, et al., "A novel deepnet model for the efficient detection of covid-19 for symptomatic patients," *Biomedical Signal Processing and Control*, vol. 68, pp. 102812, 2021.

[10] N. Aburaed, A. Panthakkan, M. Al-Saad, et al., "The impact of super resolution on detecting covid-19 from ct scans using vgg-16 based learning," *Journal of Physics: Conference Series*, vol. 1828, pp. 012009, 2021.

[11] M. Elpeltagy and H. Sallam, "Automatic prediction of covid19 from chest images using modified resnet50," *Multimedia Tools Applications*, vol. 80, pp. 26451–26463, 2021.

[12] S. Rajpal, N. Lakhyani, A. K. Singh, et al., "Using handpicked features in conjunction with resnet-50 for improved detection of covid-19 from chest x-ray images," *Chaos, Solitons & Fractals*, vol. 145, pp. 110749, 2021.

[13] R. Jain, M. Gupta, S. Taneja, et al., "Deep learning based detection and analysis of covid-19 on chest x-ray images," *Applied Intelligence*, vol. 51, pp. 1690–1700, 2020.

[14] C. Szegedy, W. Liu, Y. Jia, et al., "Going deeper with convolutions," 2014.

[15] S. Xie, R. Girshick, P. Dollàr, et al., "Aggregated residual transformations for deep neural networks," 2017.

[16] A. Gupta, A. S. Gupta, and R. Katarya, "Instacovnet-19: A deep learning classification model for the detection of covid-19 patients using chest x-ray," *Applied Soft Computing*, vol. 99, pp. 106859, 2021.

[17] M. Rahimzadeh and A. Attar, "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*, vol. 19, pp. 100360, 2020.

[18] J. P. Cohen, P. Morrison, L. Dao, et al., "Covid-19 image data collection: Prospective predictions are the future," *arXiv 2003.11597*, 2020.

[19] X. Wang, Y. Peng, L. Lu, et al., "Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases," *CoRR*, vol. abs/1705.02315, 2017.

[20] S. Minaee, R. Kafieh, M. Sonka, et al., "Deep-covid: Predicting covid-19 from chest x-ray images using deep transfer learning," *Medical Image Analysis*, vol. 65, pp. 101794, 2020.

[21] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," *CoRR*, vol. abs/1610.02357, 2016.

[22] K. He, X. Zhang, S. Ren, et al., "Deep residual learning for image recognition," *CoRR*, vol. abs/1512.03385, 2015.