JOIV : Int. J. Inform. Visualization, 8(1) - March 2024 175-182

INTERNATIONAL JOURNAL ON INFORMATICS VISUALIZATION

journal homepage : www.joiv.org/index.php/joiv

SCOV-CNN: A Simple CNN Architecture for COVID-19 Identification Based on the CT Images

Toto Haryanto [a,*], Heru Suhartanto [b], Aniati Murni [b], Kusmardi [c,d], Marina Yusoff [e,f], Jasni Mohammad Zain [e,f]

[a] *Department of Computer Science, Institut Pertanian Bogor, Dramaga, Bogor, Indonesia*
[b] *Faculty of Computer Science, Universitas Indonesia, Beji, Depok, Indonesia*
[c] *Department of Anatomical Pathology, Universitas Indonesia/Cipto Mangunkusumo Hospital, Senen, Jakarta, Indonesia*
[d] *Human Cancer Research Cluster, Indonesia Medical Education and Research Institute, Universitas Indonesia, Jakarta, Indonesia*
[e] *Institute for Big Data Analytics and Artificial Intelligence (IBDAAI), Universiti Teknologi MARA (UiTM), Shah Alam, Selangor, Malaysia*
[f] *College of Computing, Universiti Teknologi MARA (UiTM), Shah Alam, Selangor, Malaysia*

Corresponding Author: [*]totoharyanto@apps.ipb.ac.id

***Abstract*— Since the coronavirus was first discovered in Wuhan, it has widely spread and was finally declared a global pandemic by the WHO. Image processing plays an essential role in examining the lungs of affected patients. Computed Tomography (CT) and X-ray images have been popularly used to examine the lungs of COVID-19 patients. This research aims to design a simple Convolution Neural Network (CNN) architecture called SCOV-CNN for the classification of the virus based on CT images and implementation on the web-based application. The data used in this work were CT images of 120 patients from hospitals in Brazil. SCOV-CNN was inspired by the LeNet architecture, but it has a deeper convolution and pooling layer structure. Combining seven and five kernel sizes for convolution and padding schemes can preserve the feature information from the images. Furthermore, it has three fully connected (FC) layers with a dropout of 0.3 on each. In addition, the model was evaluated using the sensitivity, specificity, precision, F1 score, and ROC curve values. The results showed that the architecture we proposed was comparable to some prominent deep learning techniques in terms of accuracy (0.96), precision (0.98), and F1 score (0.95). The best model was integrated into a website-based system to help and facilitate the users' activities. We use Python Flask Pam tools as a web server on the server side and JavaScript for the User Interface (UI) Design.**

***Keywords*— CNN; COVID-19; CT image; SCOV-CNN.**

Manuscript received 15 Apr. 2023; revised 8 Jul. 2023; accepted 19 Aug. 2023. Date of publication 31 Mar. 2024.

I. INTRODUCTION

The coronavirus was first reported in Wuhan and has continued to spread to spread until it was officially named COVID-19 by the WHO. Until June 30, 2020, this virus has infected more than ten million with a mortality rate of five hundred thousand [1], which has continued to increase. Coronavirus infects the lungs and causes the accumulation of large quantities of fluid in the lower part. This condition causes shortness of breath, which may lead to death. The appearance of the lungs of patients infected with the virus is seen through medical images such as X-ray or CT images. Although determining whether a patient is infected using polymerase chain reaction (PCR), medical images are needed to analyze the lung's condition further.

There are many famous deep learning architecture applications such as Residual Network (ResNet) [2], DenseNet [3]–[5], Visual Geometry Group Network (VGGNet) [6], GoogleNet [7], and AlexNet [8]. Deep learning is also widely used for several medical datasets such as Magnetic Resonance Imaging (MRI) in [9]–[12], Histopathology in[13]–[18], Computerized Tomography (CT), and Radiology [19]–[22]. In this work, a simple but effective design of CNN architecture was carried out to identify the CT images of COVID-19 patients. In computer vision and medical imaging, X-ray and CT images are used for diagnosis or prognosis[23]. Diagnosis is the actual recognition of the disease being suffered. In comparison, prognosis predicts the development of disease after it has been diagnosed. The potential to increase COVID-19 image data in both CT and X-ray, which will be processed in the parallel computing environment, was the background for this work. Meanwhile, the development of the vaccine is still in progress and has not been produced massively. Therefore, transmission still occurs in many countries, especially Indonesia.

One of the studies was carried out using medical X-ray images [24]. The support vector machine (SVM) approach as a classifier and multi-level thresholding for image feature extraction was used. Other studies using X-ray images were carried out in [25]. The fuzzy color was utilized to improve the image quality. Furthermore, two deep learning architectures, MobileNetV2 and SqueezeNet, were compared as feature extraction methods before classification with SVM. An automatic COVID-19 detection using X-ray images has also been proposed in [26]. In addition, a deep learning method with You Only Looks Once (YOLO) for object detection was applied. This method obtained 98% and 87% accuracy for binary and multi-class classification, respectively. The Bayesian method and SqueezeNet were integrated for COVID-19 detection using the X-ray images in [27]. This study claimed that their approach would reveal higher diagnosis accuracy.

Meanwhile, using CT images to identify patients with COVID-19 has also been published in several journals. For example, it was used in [28], and the medical viewpoint was fully described. Furthermore, some studies that used CT images for analysis and classification, as proposed in [29], adopted texture feature extraction and SVM to classify the COVID-19 case. In addition, it was also used with the transfer learning method for classification, as published in [30]. Currently, the number of CT images obtained is minimal, and there are several similar images of COVID-19 and non-COVID-19 patients. Therefore, a good feature extraction was needed. The other datasets from 120 patients in Brazilian hospitals were collected as reported [31]. The Towards Explainable Deep Neural Network (xDNN) was applied for classification and compared to deep learning methods such as ResNet, GoogleNet, VGG-16, and AlexNet.

In our research, a simple CNN architecture was proposed, which performs feature extraction while producing a classification model for COVID-19 identification based on CT images. This proposed architecture was called SCOV-CNN, an abbreviation for Simple COVID CNN. In addition, a web-based application tool was developed to implement this mode. Its structure was inspired by the LeNet architecture, the pioneer of classic CNN [32]. LeNet only has two convolutions, pooling and fully connected layers. Therefore, extracting high-level features increased the number of convolutions, pooling, and fully connected layers to seven, seven, and three, respectively. Details of the SCOV-CNN structure is described in the method section.

This article was divided into some sections. Section I explained the introductory aspect, followed by section II, which explained the methodology with a brief narration of X-ray and CT images used to analyze and identify COVID-19 patients. Section III contains the results and comparison among the current studies on COVID-19. This work was discussed, and future direction was stated in section IV.

II. Materials and Method

A. Images for COVID-19 Identification

X-ray images are common modalities that are obtained to analyze any chest disease. COVID-19 prediction using this data has been carried out in several studies. They are relatively lower-cost images and have been widely used for various diseases such as osteoporosis [33], breast cancer, and cardiac disease [34]. Although obtaining an image is relatively low-cost, it is not as detailed as a CT when capturing information on certain parts, for example, smooth tissues. Therefore, a pre-processing technique is needed for further studies using X-ray images.

B. CT Images for COVID-19 Identification

CT image is a type of image used to analyze diseases of the lungs. Its quality is usually better than X-ray; therefore, it was used in this work to obtain COVID and non-COVID images. An example of a CT image of the patients is shown in Figure 1.

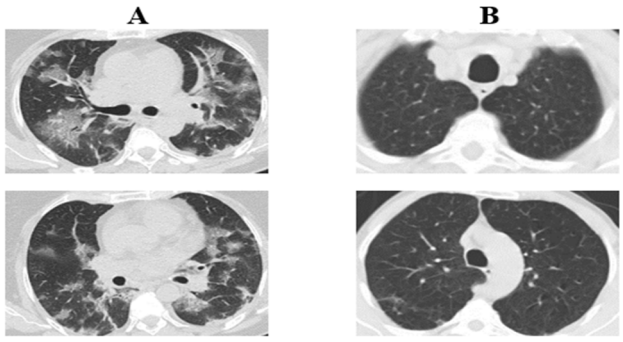

Fig. 1 Example COVID-19 CT images (A) and non-COVID-19 CT images (B)

CT data was obtained from a previous image dataset to develop a deep learning model from scratch research [30]. However, this number was too small. Therefore, other better representatives were sourced. The deep learning techniques will not function properly with limited data. Consequently, we collected another CT image dataset to develop a deep-learning model from scratch.

TABLE I
DETAIL CT IMAGES DATASET FROM PATIENTS [30]

No	Type of Data	COVID-19	Non COVID-19
1	Training	191	243
2	Validation	60	58
3	Testing	95	96
	Total	**346**	**397**

As the additional dataset, CT resources from some patients in Brazilian hospitals were also obtained [31]. There were 120 patients, 60 infected by COVID-19, and 60 were not. A total of 2482 CT scan images were obtained, consisting of 1252 infected or positive SARS-CoV-2 and 1230 from uninfected patients. Based on this dataset, separation into the training (85%), validation (10%), and testing (5%) were carried out. The detailed number of images of the dataset [30] and [31] is shown in Table 1 and Table 2.

TABLE II
DETAIL CT IMAGES DATASET FROM PATIENTS [31]

No	Type of Data	COVID-19	Non COVID-19
1	Training	1064	1044
2	Validation	125	123
3	Testing	63	63
Total		**1252**	**1230**

C. Design SCOV-CNN

Deep learning and designing CNN architecture are always challenging activities. For this research, an existing or a new architecture relevant to the problem was created. CNN can be a feature extractor [35] and a classifier. Therefore, SCOV-CNN was introduced as a simple yet robust architecture for feature extraction, including a CT image classifier for identifying COVID-19 patients. Visually, the final SCOV-CNN architecture is shown in Figure 2.

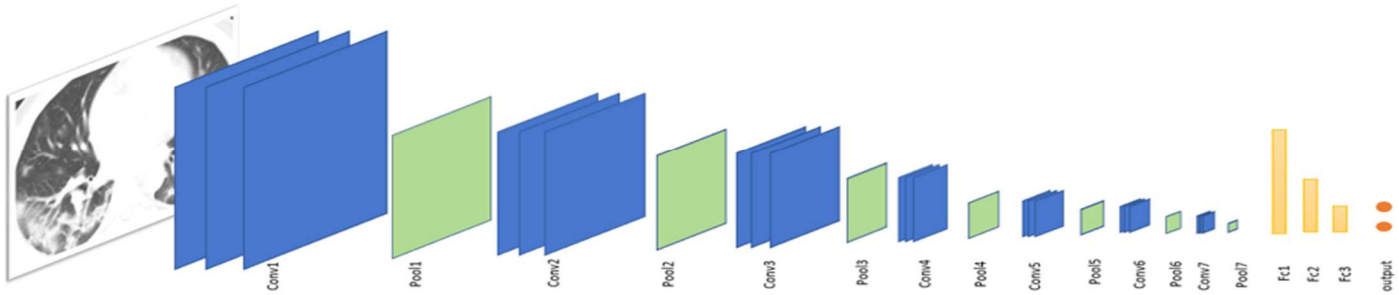

Fig. 2 Proposed SCOV-CNN architectures

The layers used were adopted from the LeNet architecture [32] and some preliminary studies with fewer layer structures. The differences between SCOV-CNN architecture and several developed architectures are presented in Table 3. The first layer was the input layer with an image size of three RGB channels and dimensions of 3 x 224 x 224. The SCOV-CNN architecture has seven convolution layers, followed by pooling for each. The combination of the convolution and pooling layers are shown in Figure 2. A neural network is a linear combination of neurons and the weights of related neurons. The result of this linear combination was transformed using the activation function. There was a ReLu activation function at each convolution layer, which was useful in avoiding vanishing gradients. The ReLu equation formula refers to equation (1).

$$Relu = \begin{cases} 0\,; & \text{for x} < 0 \\ x\,; & \text{for } x \geq 0 \end{cases} \quad (1)$$

Where x is the result of a linear combination, which will be zero when the result from the linear combination is 0. In addition, the ReLu value becomes x when the linear combination is positive. At convolution layers 1, 3, 5 and 7, the feature map was obtained by multiplying the input image with a 7x7 kernel. Meanwhile, for 2, 4 and 6, the kernel size used to form the feature map was 5x5. The output at the convolution layer used the same for padding. Therefore, the output dimensions did not change. The reason for implementing this padding mode was to preserve the input dimensions to minimize the loss of feature information. The image dimension reduction mechanism occurred in the pooling process with a stride of 2. This process causes the resulting output size to exceed the feature map image input.

TABLE III
SCOV-CNN AMONG DIFFERENT DEEP LEARNING ARCHITECTURES

LeNet-5 [32]	**AlexNet** [8]	**VGGNet** [6]	**ResNet** [2]	**SCOV-CNN [our]**
input	Input	input	input	input
Conv5-6	Conv11-96	Conv3-64	Conv7-64	Conv7-128
Max pooling	Max pooling	Conv3-64	Conv3-64	Max pooling
Conv5-6	Conv11-256	Max pooling	Conv3-64	Conv5-256
Max pooling	Max pooling	Conv3-128	Conv3-64	Max pooling
FC-120	Conv3-384	Conv3-128	Conv3-64	Conv7-512
FC-84	Conv3-384	Max pooling	Conv3-64	Max pooling
Gaussian	Max pooling	Conv3-256	Conv3-64	Conv5-512
	FC-4096	Conv3-256	Conv3-128	Max pooling
	FC-4096	Conv3-256	Conv3-128	Conv7-256
	SoftMax	Max pooling	Conv3-128	Max pooling
		Conv3-512	Conv3-128	Conv5-256
		Conv3-512	Conv3-128	Max pooling
		Conv3-512	Conv3-128	Conv7-128
		Max pooling	Conv3-128	Max pooling
		Conv3-512	Conv3-128	FC-1024
		Conv3-512	Conv3-128	FC-512
		Conv3-512	Conv3-128	FC-256
		Max pooling	Conv3-128	SoftMax
		FC-4096	Conv3-128	
		FC-4096	Conv3-256	
		FC-1000	Conv3-256	
		SoftMax	Conv3-256	
			Conv3-256	
			Conv3-256	
			Conv3-256	
			Conv3-256	
			Conv3-256	
			Conv3-256	
			Conv3-256	

LeNet-5 [32]	AlexNet [8]	VGGNet [6]	ResNet [2]	SCOV-CNN [our]
			Conv3-256	
			Conv3-512	
			Conv3-512	
			Conv3-512	
			Conv3-512	
			Conv3-512	
			Conv3-512	
			Avg. Pooling	
			FC-1000	
			SoftMax	

Based on Figure 2, starting from the first to the seventh pooling layer, the extraction process was produced by SCOV-CNN. The next is the three fully connected (FC) layers. Each had a dropout of 0.3, which plays a role in preventing the resulting model from overfitting. In the output section, the SoftMax function (eq. 2) was used due to the classification.

$$\text{Softmax}(\vec{z}) = \frac{\exp(z_i)}{\sum_{j=1}^{K} \exp(z_i)} \quad (2)$$

SoftMax converted the numeric output value of the last linear layer into a probability value. Furthermore, it raised the natural number e with a certain numeric value (x_i) and divided by the total power of the natural number *e* to the respective numerical values generated. With this function, the predicted class values will be proportionally compared with all the predicted values. The function with the highest probability value represents the predicted class. The architecture was trained with a 2000 epoch, 32 mini batch-size per epoch, and 0.00001 learning rate. The SCOV-CNN detail parameters are shown in Table 4.

TABLE IV
DETAIL PARAMETER OF SCOV-CNN

No	Layers	Dimension	Parameters
1	Input	3x224x224	0
2	conv1 (n=7, m=7, l=3, k=128)	128x224x224	18.944
3	Pooling1	128x112x112	0
4	conv2 (n=5, m=5, l=128, k=256)	256x112x112	819.456
5	Pooling2	256x56x56	0
6	conv3 (n=7, m=7, l=256, k=512)	512x56x56	6.423.040
7	Pooling3	256x56x56	0
8	conv4 (n=5,m=5,l=512,k=512)	512x28x28	6.554.112
9	Pooling4	256x56x56	0
10	conv5(n=7,m=7,l=512,k=256)	256x14x14	6.422.784
11	Pooling5	256x56x56	0
12	conv6(n=5,m=5,l=256,k=256)	256x7x7	1.638.656
13	Pooling6	256x56x56	0
14	conv7(n=7,m=7,l=256,k=128)	128x3x3	1.605.760
15	Flatten	128	0
16	Dense (l=128, k=1024)	1024	132.096
17	Dense (l=1024, k=512)	512	524.800
18	Dense (l=512,k=256)	258	131.328
19	Output (l=56,k=2)	2	514
	Total	-	**24.271.490**

D. Experiment Setup

The experiment ran on Ubuntu 16.04 Operating System supported by Intel Core i7 16 cores @3GHz clock speed with 128 GB of RAM. The training was supported by a triple Graphic Processing Unit (GPU) NVIDIA GTX-1080 server hardware resource with 8 GB memory per GPU. CUDA 9.0 and 'cudnn' were used to support deep learning. This was carried out in parallel on the three GPUs using the multi-GPU model library, Keras 2.0 and TensorFlow 1.14.0.

E. Evaluation

The proposed Model developed through the training process was tested using data testing to ascertain its performance. In the medical field, for diagnosis, the sensitivity, specificity, and ROC graphs are commonly used metrics. The sensitivity and specificity value is formulated through equations (3) and (4), respectively. Meanwhile, for ROC, a graph between the True (TPR) and False Positive Rate (FPR) (5) was plotted.

$$\text{sensitivity} = \text{recall} = \text{TPR} = \frac{TP}{TP+FN} \quad (3)$$

$$\text{specificity} = \frac{TN}{TN+FP} \quad (4)$$

$$\text{FPR} = 1 - \text{specificity} = 1 - \frac{\text{TN}}{\text{TN+FP}} \quad (5)$$

The recall, precision and F1 score, were calculated and compared with those from the previous studies. The formula used for recall was similar to sensitivity. Meanwhile, precision and F1 score formulas are presented in equation (6) and (7).

$$\text{precision} = 1 - \frac{\text{TP}}{\text{TP+FP}} \quad (6)$$

$$\text{F1 score} = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precsion} + \text{recall}} \quad (7)$$

III. RESULTS AND DISCUSSION

A. Data Acquisition and Pre-processing

Theoretically, variations in data will have an impact on the model's ability to recognize new data under various conditions. Therefore, the augmentation process was implemented on both datasets [30] and [31]. For the pre-processing step, some of the augmentations performed include clockwise rotation = 0.5, vertical flip, and horizontal flip.

B. SCOV-CNN Training Progress

The designed architecture produced the output dimensions for each layer and several parameters, as presented in Table 3. In total, the SCOV-CNN architecture revealed 24.271.490 parameters. During the training process, with 2000 epochs and 0.00001 learning rate, the model performance was viewed and presented using the loss values for each epoch. The loss value in this work was calculated using categorical loss entropy with equation (8) $\text{Loss} = -\sum_{i=1}^{N} y_i \times \log(\hat{y}_i)$ (8)

where N is the output size as the number of scalar value in the model output, yi is the i-th scalar value in the model output and represents the target value. The validation loss value for the datasets from [30]and [31] are shown in Figure 3(A). Furthermore, the validation accuracy during training was

recorded, as illustrated in Figure 3 (B). Therefore, the loss of value and accuracy becomes the consideration for the next analysis and future studies. Based on the loss parameters and accuracy of the two datasets used, it was understood that the number and dataset used has an impact on the performance of the training process. Therefore, for further analysis of SCOV-CNN performance, more focus will be on the dataset [31].

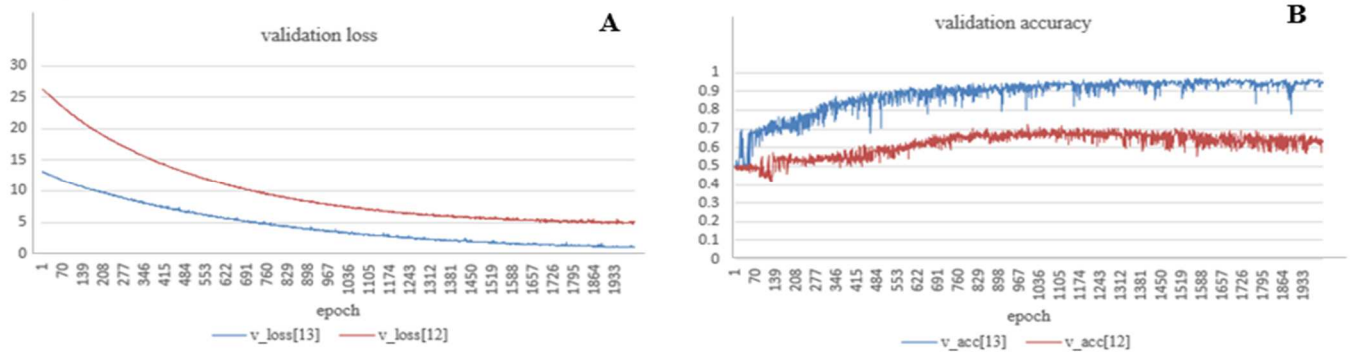

Fig. 3 Validation Loss and accuracy values for both data set [30] and [31].

C. Model Performance and Analysis

Models generated from the SCOV-CNN architecture were saved in a file format with the extension .hdf5. Using this model, 63 test data were identified. The results were mapped into a confusion matrix and the sensitivity, specificity and ROC curve were further calculated. The test result configuration matrix is shown in Table V dataset [31] and performance analysis by ROC shown in Figure 4.

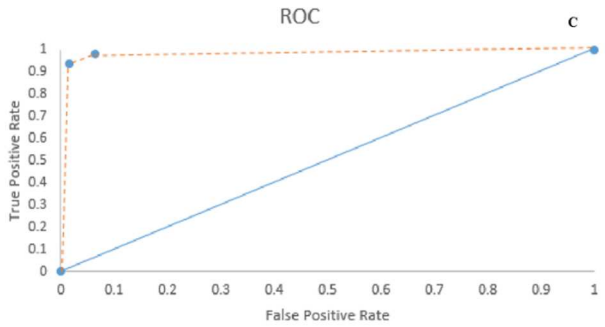

Fig. 4 Receiver operating characteristic (ROC) curve for CT imagas classification

TABLE V
CONFUSION MATRIX

	Predicted	
Actual	COVID	Non-COVID
COVID	59	4
Non-COVID	1	61

The data obtained from this research was used to determine the position of this work against others. It was compared with previous studies, i.e., [31] and [36], which used similar data. The comparison is presented in Table 6.

TABLE VI
COMPARISON OF PERFORMANCE WITH PREVIOUS RESEARCH

Method	Metric			
	Accuracy	Recall	Precision	F1-score
DL-voting based [36]	98,99%	99,20%	98,80%	99,00%
xDNN [31]	97,38%	99,16%	95,53%	97,31%
ResNet [31]	94,96%	93,00%	97,15%	95,03%
GoogleNet [31]	91,73%	90,20%	93,50%	91,82%
VGG-16 [31]	94,96%	94,02%	95,43%	94,97%
AlexNet [31]	93,75%	94,98%	92,28%	93,61%
Decision Tree [31]	79,44%	76,81%	83,13%	79,84%
Adabost [31]	95,16%	93,81%	96,71%	95,14%
SCOV-CNN [our]	96,00%	98,00%	94,00%	95,93%

Although the accuracy and F1 score obtained were relatively good, it was better when the model was tested with patient data from other countries. This was carried out to avoid overestimating the performance because in computer vision, differences in tools, sensors, and the environment affect the diversity of images obtained. Further analysis focuses on the misclassification which occurred on the SCOV-CNN model. Based on the confusion matrix in Table 5 dataset [31], four CT COVID images were not identified as COVID. Conversely, only one CT image of non-COVID was identified as a COVID sample, as shown in Figure 5 for the detailed results.

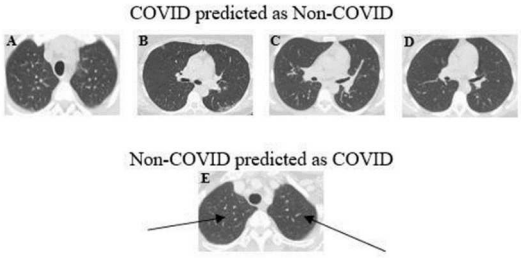

Fig. 5 Example of misclassification of CT images: (A, B, C, and D) are COVID images that predicted as non-COVID images. (E) is a non-COVID image that is predicted as a COVID image.

Further analysis focuses on the misclassification that occurred in the SCOV-CNN model. Based on the confusion matrix in Table 5 dataset [31], four CT COVID images were not identified as COVID. Conversely, only one CT image of non-COVID was identified as a COVID sample, as shown in Figure 5 for the detailed results.

Figure 5 shows four images of COVID that have been misclassified and, therefore, were predicted to be non-COVID.

When viewed at a glance by non-experts, the four images do not show clear COVID features in patients' lungs. Conversely, the misclassification of one detected image was thought to be due to fluid spots in the lungs, as indicated by the arrow. However, this certainly requires further analysis from a radiologist or pulmonologist.

D. Model Implementation

The good models were best saved in the. hdf5 file format. Furthermore, they were integrated into a website-based system to help and facilitate the users' activities. In other for the model to be read and integrated, the Python Flask Pam tool was used as a web server on the server-side. Meanwhile, for the client-side, the application was carried out using a browser. The graphical User Interface (GUI) implementation is presented in Figure 6.

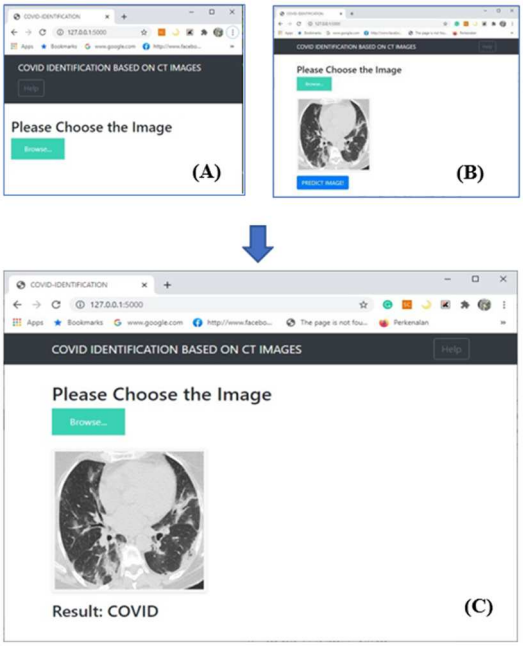

Fig. 6 GUI for SCOV-CNN implementation: (A) homepage, (B) choose the image, (C) result of model prediction

CT images are one of the modalities used for the identification of COVID. Two datasets were used in this work, namely patient data from [30] and [31]. After preliminary research and considering the loss and accuracy values, the dataset in [31] was used. It consisted of 60 CT image data infected with COVID, comprising of 32 men and 20 women. Meanwhile, that of the uninfected patients was obtained from 30 men and women.

The SCOV-CNN architecture designed in this work was different from that of many architectures developed today, which is seen in Table 3. The inspiration for this design was LeNet [32], which was too simple to carry out feature extraction of CT images. Finally, the proposed architecture is presented in Figure 2. After the training process, the model obtained was tested on 63 new data using dataset, which was seen in Table 5. According to the confusion matrix in Table 5 dataset [31], the performance level using the Receiver Operating Characteristic (ROC) curve was visualized. Figure 4 showed the performance of SCOV-CNN using ROC curve.

The results were further compared with other architectural designs by comparing several parameters such as accuracy, recall, precision and F1 score. Based on the results obtained in Table 5, SCOV-CNN architecture was quite competitive. It obtained 96.00%, 98.00%, and 95.93% in terms of accuracy, precision, and F1 score, respectively, among other supervised learning. However, it was slightly better than prominent deep learning architectures such as ResNet, VGG-16, GoogleNet and AlexNet.

Compared to previous architectures such as VGG-16 with 138.4 million parameters and VGG-16 with 147.7 million total parameters [6], SCOV-CNN is more efficient with around 24 million. Automatically, the model training process will be faster. With the same dataset, research with CT images produces an accuracy of 94.96% [31]. So, SCOV-CNN outperformed with 96,00% accuracy. AlexNet [8], with several parameters of around 60 million, can only produce 93.75% [31]accuracy with the same dataset. In this study, the proposed SCOV-CNN is still better than ResNet [37], which has parameters varying between 25 - 60 million with an accuracy obtained of 94.96% [31]. Meanwhile, for XDNN and DL-voting-based, SCOV-CNN still needs more accuracy.

An architecture that is too complex is too fit to the dataset during training so that it has the potential to cause overfitting when tested with data testing. SCOV-CNN proposes a simple convolution layer considering that the availability of medical image datasets is also limited. SCOV-CNN also uses a combination of layers with a size of 5x5 and 7x7 refers to previous work [38] and a padding scheme [39] so that the formation of a convolution map can retain information from the source image.

In terms of the feature extraction process, the convolution approach is capable of extracting basic features such as image edges on the initial convolution layer up to high level features. This capability has made CNN currently the state of the art in the field of image recognition compared to several hand-crafted feature extraction methods such as morphology, color or texture based. However, the main issue in the medical field is the availability of limited datasets so that for future research, strategic transfer learning is worth considering as a deep learning approach with little data.

In addition, cross evaluation was carried out to avoid overestimation of this model. After being tested against other data, the accuracy dropped significantly by 54.45%. This shows that the size and diversity of the datasets have an important role and are needed to obtain a robust model for the various datasets.

Figure 6 is a Graphic User Interface (GUI), which applied the best model stored to predict CT images data. Figure 6(A) is the homepage of the application. In this section, users were asked to upload one CT image in .jpg, .jpeg or .png formats. The selected image was displayed after using a CT image as shown in Figure 6(B). At the bottom, there was a button" PREDICT IMAGE" and the prediction result for" COVID" appeared after the user presses the button, as shown in Figure 6 (C).

IV. CONCLUSIONS

The SCOV-CNN architecture has been successfully designed and is powerful enough to extract and identify COVID based on CT images. The resulting model was compared to prominent deep-learning architecture. This

architecture has the potential to be developed and applied to a web-based GUI application, making it easier for users to apply this model. In addition, it was concluded that the CT image sampling method, tools, and environment play an essential role in producing good data quality; therefore, the model will be more robust.

CONFLICT OF INTEREST

The authors declared no potential conflict of interest relevant to this research.

ACKNOWLEDGMENT

The authors thank the Ministry of Research Technology, Republic of Indonesia, for funding the Penelitian Terapan Unggulan Perguruan Tinggi (PTUPT) No. 8/AMD/E1/KP grant.PTNBH/2020 with contract number 8/AMD/E1/KP.PTNBH/2020 and 332/PKS/R/UI/2020 date Mei, 11 2020. This research was also partially funded by Program Bilateral Strategic Alliance (UI-UiTM BISA) Research Collaboration during 2021-2022 with contract number NKB675/UN2.RTS/HKP.05.00/2021.

REFERENCES

[1] World Health Organization, "Coronavirus Disease (COVID-19)," 2020. [Online]. Available: https://www.who.int/docs/default-source/coronaviruse/20200630-covid-19-sitrep-162.pdf?sfvrsn=e00a5466_2

[2] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Jun. 2016, doi: 10.1109/cvpr.2016.90.

[3] N. Hasan, Y. Bao, A. Shawon, and Y. Huang, "DenseNet Convolutional Neural Networks Application for Predicting COVID-19 Using CT Image," SN Computer Science, vol. 2, no. 5, Jul. 2021, doi: 10.1007/s42979-021-00782-7.

[4] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely Connected Convolutional Networks," 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Jul. 2017, doi: 10.1109/cvpr.2017.243.

[5] X. Li, X. Shen, Y. Zhou, X. Wang, and T.-Q. Li, "Classification of breast cancer histopathological images using interleaved DenseNet with SENet (IDSNet)," PLOS ONE, vol. 15, no. 5, p. e0232127, May 2020, doi: 10.1371/journal.pone.0232127.

[6] E. Guerra, J. de Lara, A. Malizia, and P. Díaz, "Supporting user-oriented analysis for multi-view domain-specific visual languages," Information and Software Technology, vol. 51, no. 4, pp. 769–784, Apr. 2009, doi: 10.1016/j.infsof.2008.09.005.

[7] C. Szegedy et al., "Going deeper with convolutions," 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Jun. 2015, doi: 10.1109/cvpr.2015.7298594.

[8] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," Communications of the ACM, vol. 60, no. 6, pp. 84–90, May 2017, doi: 10.1145/3065386.

[9] T. Rashid et al., "Deep learning based detection of enlarged perivascular spaces on brain MRI," Neuroimage: Reports, vol. 3, no. 1, p. 100162, Mar. 2023, doi: 10.1016/j.ynirp.2023.100162.

[10] S. Gassenmaier et al., "Deep Learning Applications in Magnetic Resonance Imaging: Has the Future Become Present?," Diagnostics, vol. 11, no. 12, p. 2181, Nov. 2021, doi: 10.3390/diagnostics11122181.

[11] Q. Lyu et al., "A transformer-based deep-learning approach for classifying brain metastases into primary organ sites using clinical whole-brain MRI images," Patterns, vol. 3, no. 11, p. 100613, Nov. 2022, doi: 10.1016/j.patter.2022.100613.

[12] X. Cao et al., "Add-on individualizing prediction of nasopharyngeal carcinoma using deep-learning based on MRI: A multicentre, validation study," iScience, vol. 25, no. 9, p. 104841, Sep. 2022, doi:10.1016/j.isci.2022.104841.

[13] J. Conway et al., "Integration of deep learning-based histopathology and transcriptomics reveals key genes associated with fibrogenesis in patients with advanced NASH," Cell Reports Medicine, vol. 4, no. 4, p. 101016, Apr. 2023, doi: 10.1016/j.xcrm.2023.101016.

[14] A. Pal et al., "Deep multiple-instance learning for abnormal cell detection in cervical histopathology images," Computers in Biology and Medicine, vol. 138, p. 104890, Nov. 2021, doi:10.1016/j.compbiomed.2021.104890.

[15] P.-W. Huang et al., "Deep-learning based breast cancer detection for cross-staining histopathology images," Heliyon, vol. 9, no. 2, p. e13171, Feb. 2023, doi: 10.1016/j.heliyon.2023.e13171.

[16] S. Hosseinzadeh Kassani, P. Hosseinzadeh Kassani, M. J. Wesolowski, K. A. Schneider, and R. Deters, "Deep transfer learning based model for colorectal cancer histopathology segmentation: A comparative study of deep pre-trained models," International Journal of Medical Informatics, vol. 159, p. 104669, Mar. 2022, doi:10.1016/j.ijmedinf.2021.104669.

[17] A. Mundhada, S. Sundaram, R. Swaminathan, L. D' Cruze, S. Govindarajan, and N. Makaram, "Differentiation of urothelial carcinoma in histopathology images using deep learning and visualization," Journal of Pathology Informatics, vol. 14, p. 100155, 2023, doi: 10.1016/j.jpi.2022.100155.

[18] T. Haryanto, H. Suhartanto, A. M. Arymurthy, and K. Kusmardi, "Conditional sliding windows: An approach for handling data limitation in colorectal histopathology image classification," Informatics in Medicine Unlocked, vol. 23, p. 100565, 2021, doi:10.1016/j.imu.2021.100565.

[19] O. Attallah, "RADIC:A tool for diagnosing COVID-19 from chest CT and X-ray scans using deep learning and quad-radiomics," Chemometrics and Intelligent Laboratory Systems, vol. 233, p. 104750, Feb. 2023, doi: 10.1016/j.chemolab.2022.104750.

[20] H. Ulutas, M. E. Sahin, and M. O. Karakus, "Application of a novel deep learning technique using CT images for COVID-19 diagnosis on embedded systems," Alexandria Engineering Journal, vol. 74, pp. 345–358, Jul. 2023, doi: 10.1016/j.aej.2023.05.036.

[21] G. Celik, "Detection of Covid-19 and other pneumonia cases from CT and X-ray chest images using deep learning based on feature reuse residual block and depthwise dilated convolutions neural network," Applied Soft Computing, vol. 133, p. 109906, Jan. 2023, doi:10.1016/j.asoc.2022.109906.

[22] G. Wu and J. Duan, "BLCov: A novel collaborative–competitive broad learning system for COVID-19 detection from radiology images," Engineering Applications of Artificial Intelligence, vol. 115, p. 105323, Oct. 2022, doi: 10.1016/j.engappai.2022.105323.

[23] Y. Liao, H. Liu, and I. Spasić, "Deep learning approaches to automatic radiology report generation: A systematic review," Informatics in Medicine Unlocked, vol. 39, p. 101273, 2023, doi:10.1016/j.imu.2023.101273.

[24] L. N. Mahdy, K. A. Ezzat, H. H. Elmousalami, H. A. Ella, and A. E. Hassanien, "Automatic X-ray COVID-19 Lung Image Classification System based on Multi-Level Thresholding and Support Vector Machine," Apr. 2020, doi: 10.1101/2020.03.30.20047787.

[25] M. Toğaçar, B. Ergen, and Z. Cömert, "COVID-19 detection using deep learning models to exploit Social Mimic Optimization and structured chest X-ray images using fuzzy color and stacking approaches," Computers in Biology and Medicine, vol. 121, p. 103805, Jun. 2020, doi: 10.1016/j.compbiomed.2020.103805.

[26] T. Ozturk, M. Talo, E. A. Yildirim, U. B. Baloglu, O. Yildirim, and U. Rajendra Acharya, "Automated detection of COVID-19 cases using deep neural networks with X-ray images," Computers in Biology and Medicine, vol. 121, p. 103792, Jun. 2020, doi:10.1016/j.compbiomed.2020.103792.

[27] F. Ucar and D. Korkmaz, "COVIDiagnosis-Net: Deep Bayes-SqueezeNet based diagnosis of the coronavirus disease 2019 (COVID-19) from X-ray images," Medical Hypotheses, vol. 140, p. 109761, Jul. 2020, doi: 10.1016/j.mehy.2020.109761.

[28] H. Dai et al., "High-resolution Chest CT Features and Clinical Characteristics of Patients Infected with COVID-19 in Jiangsu, China," International Journal of Infectious Diseases, vol. 95, pp. 106–112, Jun. 2020, doi: 10.1016/j.ijid.2020.04.003.

[29] M. Barstugan, U. Ozkaya, and S. Ozturk, "Coronavirus (COVID-19) Classification using CT Images by Machine Learning Methods," no. 5, pp. 1–10, 2020.

[30] J. Zhao, Y. Zhang, X. He, and P. Xie, "COVID-CT-Dataset: A CT Scan Dataset about COVID-19," pp. 1–5, 2020.

[31] P. Angelov and E. Soares, "Explainable-By-Design Approach For Covid-19 Classification Via CT-Scan," Apr. 2020, doi:10.1101/2020.04.24.20078584.

[32] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," Proceedings of the IEEE, vol. 86, no. 11, pp. 2278–2324, 1998, doi: 10.1109/5.726791.
[33] P. Pisani, "Screening and early diagnosis of osteoporosis through X-ray and ultrasound based techniques," World Journal of Radiology, vol. 5, no. 11, p. 398, 2013, doi: 10.4329/wjr.v5.i11.398.
[34] M. A. Speidel, B. P. Wilfley, J. M. Star-Lack, J. A. Heanue, and M. S. Van Lysel, "Scanning-beam digital x-ray (SBDX) technology for interventional and diagnostic cardiac angiography," Medical Physics, vol. 33, no. 8, pp. 2714–2727, Jul. 2006, doi: 10.1118/1.2208736.
[35] L. Hertel, E. Barth, T. Kaster, and T. Martinetz, "Deep convolutional neural networks as generic feature extractors," 2015 International Joint Conference on Neural Networks (IJCNN), Jul. 2015, doi:10.1109/ijcnn.2015.7280683.
[36] P. Silva et al., "COVID-19 detection in CT images with deep learning: A voting-based scheme and cross-datasets analysis," Informatics in Medicine Unlocked, vol. 20, p. 100427, 2020, doi:10.1016/j.imu.2020.100427.
[37] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Jun. 2016, doi: 10.1109/cvpr.2016.90.
[38] T. Haryanto, H. Suhartanto, A. M. Arymurthy, and K. Kusmardi, "Conditional sliding windows: An approach for handling data limitation in colorectal histopathology image classification," Informatics in Medicine Unlocked, vol. 23, p. 100565, 2021, doi:10.1016/j.imu.2021.100565.
[39] T. Haryanto, I. S. Sitangggang, M. Agmalaro, and R. Rulaningtyas, "The Utilization of Padding Scheme on Convolutional Neural Network for Cervical Cell Images Classification2020.," in *International Conference on Computer Engineering, Network and Intelligent Multimedia (CENIM 2020)*, 2020, pp. 34–38.