Informatics in Medicine Unlocked 28 (2022) 100842

Contents lists available at ScienceDirect

Informatics in Medicine Unlocked

journal homepage: www.elsevier.com/locate/imu

UNAS-Net: A deep convolutional neural network for predicting Covid-19 severity

Abdusy Syarif [a], Novi Azman [a,*], Viktor Vekky Ronal Repi [a], Ernawati Sinaga [a], Muhamad Asvial [b]

[a] Biomedical Computing and Physics Engineering (BIOCOPE) Research Group, Universitas Nasional, Indonesia
[b] Department of Electrical Engineering, Universitas Indonesia, Indonesia

ARTICLE INFO

Keywords:
Deep learning
Covid-19
Image enhancement

ABSTRACT

We present a study on Covid-19 detection using deep learning algorithms that help predict and detect Covid-19. Chest X-ray images were used as the input dataset to prepare and train the proposed model. In this context, deep learning architecture (DLA) and optimisation strategies have been proposed and explored to support the automated detection of Covid-19. A model based on a convolutional neural network was proposed to extract features of images for the feature-learning phase. Data augmentation and fine-tuning with deep-feature-based methods were applied to improve the model. Image enhancement and saliency maps were used to enhance visualisation and estimate the disease severity level based on two parameters; degree of opacity and geographic extent. Contrast-limited adaptive histogram equalization and Otsu thresholding were employed with several parameters to investigate the effects on the visualisation results. An experimental investigation was performed between the proposed method and other pretrained DLAs. The proposed work obtained excellent classification accuracy and sensitivity of 97.36% and 95.24% respectively. In addition, the input parameters for image enhancement significantly affected the results. The overall performance metrics were perfect for DenseNet and adequately high for the proposed work which is comparable to other models. Data augmentation and fine-tuning successfully handed the networks to enhance the overall performance, especially in our case with limited datasets.

1. Introduction

Like any other country in this world, Indonesia could not escape from the corona-virus disease-19 (Covid-19); moreover, it has been witnessing an increasing number of infected people. As the fourth largest population in the world, Indonesia has a high risk of exposure. However, China, as the country with the largest population, employs artificial intelligence (AI)-assisted computed tomography (CT) imaging analysis to investigate Covid-19 cases and streamline diagnosis [8]. Since then, several research groups have launched AI initiatives to enhance the prediction of Covid-19 by exploiting clinical data and basic chest X-ray (CXR) images to create a predictive model. In this study, on the basis of Wang, Lin and Wong (2020), and to the best of our knowledge, Indonesia does not have any research study center that uses clinical data such as CXR images; by contrast, a prototype of a deep learning model with visualisation features has been built as a platform to predict Covid-19.

The rest of this study is organised as follows: Section 2 provides information of the related work. Section 3 explains the proposed scheme. Section 4 evaluates the performance method. Section 5 provides the conclusions and some perspectives on future works.

2. Related work

Since the outbreak of corona-virus in December 2019, several researchers from the fields of medicine, clinical, radiology, oncology, bioinformatics, and computer science have been working on this field.

Artificial Intelligence (AI) can be as accurate as humans, can save time of radiologists [24], perform a quick diagnosis and is cheaper than standard tests for Covid-19. Chest radiography (CXR) and computed tomography (CT) scans have been used. CT yields the highest diagnostic sensitivity; however, it is very complex to employ in an intensive care unit environment and is likely to be unavailable to patients in countries with less developed healthcare systems. By contrast, CXR exhibits

* Corresponding author.
E-mail address: novi.azman@civitas.unas.ac.id (N. Azman).

https://doi.org/10.1016/j.imu.2021.100842
Received 30 August 2021; Received in revised form 29 December 2021; Accepted 30 December 2021
Available online 4 January 2022

inferior diagnostic sensitivity but is commonly available, minimally invasive, fast, and a reasonable tool to diagnose and monitor Covid-19. Several studies, such as [10,13,14,21]; have described the use of deep learning [12] to diagnose Covid-19 using X-ray and/or CT scan images. However, they used a simple model only [15]. proposed a framework based on built-in smartphone sensors e.g microphones, cameras, temperature sensors, humidity sensors, inertial sensors, proximity sensors, colour sensors, and wireless sensors to detect Covid-19. However, these solutions are still far from being optimal as smartphone sensors exhibit a low level of accuracy and their technical properties may vary.

Few initiatives have been considered in this regard. Owing to the demand for faster interpretation of radiography images, AI methods based on deep learning have been proposed, and the results are effective in detecting Covid-19 via radiography imaging, accompanied by CT imaging, in terms of accuracy [4,13,14,17,23]. In addition [21],developed an open-source network, known as Covid-Net, to detect Covid-19 using X-ray images. It has been developed from ~ 13,000 images of patient cases, including Covid-19 positive cases. However, the authors have stated that Covid-Net is by no means a production-ready solution and needs an improvement.

[10] proposed a deep convolutional neural network (CNN), known as CoroNet model (which is based on the works of [2] - the Xception architecture pre-trained model), to automatically detect Covid-19 based on X-ray images. Their model is categorised into four classes: Covid-19, Normal, Pneumonia-bacterial and Pneumonia-viral. [16]; Acharya UR proposed a model based on DarkNet deep learning architecture DLA [6]) to detect and classify Covid-19 cases based on X-ray images. Their model obtained binary and 3-class classification accuracies of 98.08% and 87.02%, respectively, on a dataset containing 125 Covid-19, 500 normal and 500 Pneumonia CXR images. In recent times, several initiatives have used DLA to diagnose Covid-19 based on CT and/or X-ray images.

For the subset of diagnostic models based on medical imaging, we hypothesised that there are two main reasons for high risk of bias, *i.e.*, a lack of information to evaluate selection bias and a lack of apparent reporting of image annotation procedures and quality control measures. Moreover, we observed a high risk of bias in all the studies and the reported results were assured, and there were no visualisation features.

We brainstormed recent studies that used deep learning or machine learning models to detect Covid-19; however, there is an obstacle which is the reliability of training data that captures the issue complexity, but does not lead to undetected bias in the model. Therefore, there is a need to reinforce predictive modeling with a visualisation features to improve the interpretability or perception of information in images to aid human viewers, i.e doctors. Consequently, this research focuses on how to enrich our own dataset, especially training data, build a model, and provide computer visualisation using image enhancement algorithms and saliency maps.

3. Proposed work

In this section we discuss the proposed method and its implementation, including the architecture design methodology, network architecture, and process of dataset preparation.

3.1. UNAS-Net

Motivated and inspired by the absence of a platform, in this study, we propose a model known as Universitas National Network (UNAS-Net), which is designed by the Nasional University, Jakarta, Indonesia, using a deep learning method based on a CNN to detect Covid-19 based on the X-ray images of patients in Indonesia; these images are collected in a centralised dataset (Fig. 1).

Fig. 2 depicts the proposed method, including the main deep learning phases, such as preprocessing, data splitting, training, validation, and testing. These phases are explained in the following subsections.

3.2. Dataset and model

Machine learning requires a training dataset that is used to train the model. Before feeding the dataset, the first step was collect the specific data. We collected datasets form several sources (Table 1).

However, these datasets include CXR and CT scan images of different formats and sizes. As we are interested in CXR images, we need to preprocess the datasets. For instance, we selected 349 of 481 CXR images of Covid-19 based on the works of [3]. Moreover, we collected images from other repositories, i.e., [1,21]. In addition, we collected 850 CXR images from hospitals in Jakarta, i.e., Hospital of Christian University Indonesia, Duren Sawit Hospital and National Brain Center Hospital. All images were claimed as Covid-19 positive cases by doctors of three establishments. We admit that the dataset is small owing to the lack of benchmark datasets for Covid-19, especially in CXR images.

After preprocessing phase, we obtained a dataset that contains CXR of Covid-19 and non-Covid-19 patients, although the number of datasets is inadequate to validate the model. This condition of limited data to resolve a problem often occurs as gathering a large dataset can be prohibitively costly, especially data with X-ray images from hospitals in Indonesia. Consequently, the only choice is to work with a limited dataset and try to achieve as accurate predictions as possible. We started training a small network of images as training samples, without any regularisation to set up a preliminary study. This scheme produced a classification accuracy of overfitting which can be identified by looking at validation metrics, such as loss or accuracy. The validation metric

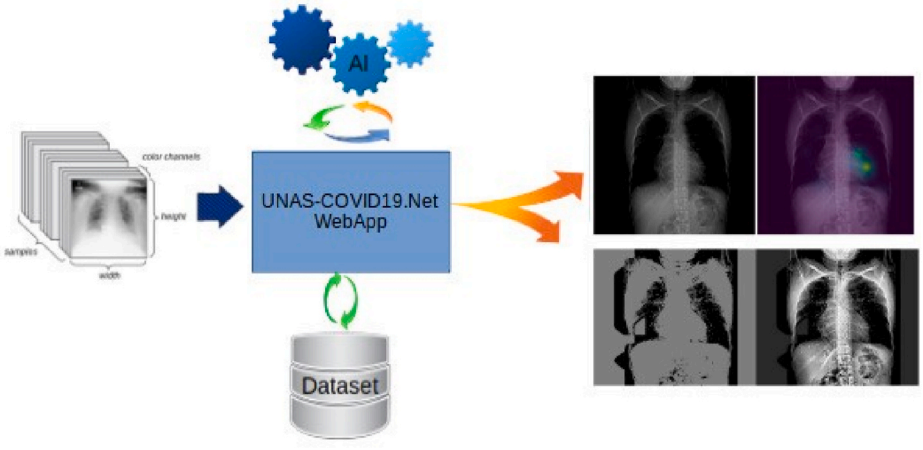

Fig. 1. UNAS-Net platform.

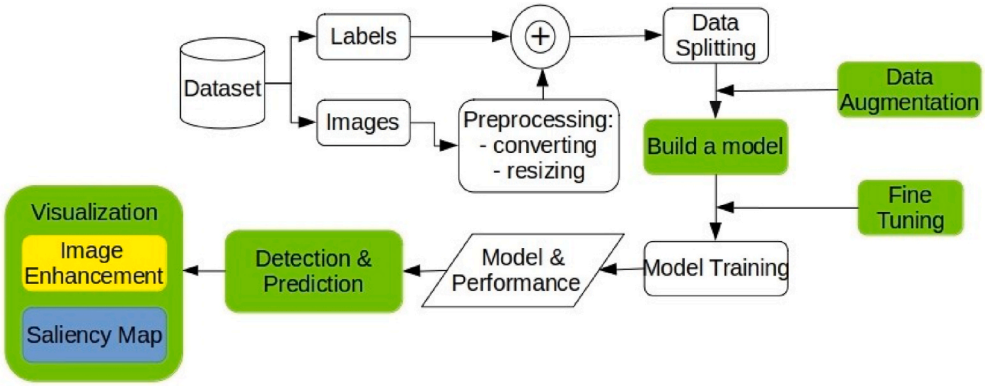

Fig. 2. Flowchart of the proposed transfer learning framework.

Table 1
Source of datasets.

Source	No. of Images	
	Covid-19	non-Covid-19
[3]	481	173
[21]	1387	330
[1]	912	116
UNAS-Net	850	0

stops reconstructing after a certain number of epochs and starts to decline afterwards. The training metric needs to be enhanced because the model attempts to determine the best fit for the training data. To mitigate this issue to improve the model and achieve better accuracy, we applied few methods, such as data augmentation and capacity reduction of the model to retain the training data. In our case, we used dropout layers that randomly eliminate particular features. In addition, as shown in Fig. 2, we applied fine-tuning to our pretrained network to obtain optimal accuracy. These strategies are designed to tackle the issue of performing image classification based on a small dataset, which included samples of images used in this study.

An images possesses three dimensions: width, height, and colour depth. Before being fed into the network, because in this case the data collected are images with different formats, the data should be first formatted into a proper format for the network. Thus, we need a data-preprocessing phase. The steps are as follows; first, load the image files and decode them into RGB grids of pixels; second, convert them into floating-points and rescale the pixel values into a [0,1] interval, because neural networks deal with small input values.

In this data preparation phase, such as image resizing and pixel scaling steps, they have been consistently applied to all the datasets that interact with the model. We performed this preprocessing strategy to remove bias and in variance the data. Models with high variance and low bias overfit the data, whereas models with low variance and high bias underfit the data.

We used a reshaping image while preprocessing digits before feeding it into the network. Reshaping images helps arrange rows and columns to match the target shape. The reshaped image has the total number of coefficients as the initial image. In this phase, the data are pre-processed by reshaping them into a shape according to the requirements of the network and scaling them, so that all the values are in the [0, 1] interval. In addition, the reshaping phase will be very useful for the next phases in deep learning methods, e.g., transfer learning and fine-tuning processes (which will be explained later).

In fact, few lung diseases refer to disorders affecting the lungs. However, as the focus was on Covid-19 detection, we classified lung diseases into; Covid-19 and non-Covid-19. However, the classification of images was performed and confirmed by experts. The dataset was divided into train and test (Table 2). Dataset splitting is necessary for the machine learning process to eliminate bias data. The dataset is classified into train and test, where 80% of the dataset or the training set and the remaining 20% is for the testing set.

In our case, we tackled a binary-classification problem. The model network should end with a single unit where this unit will encode the probability which the network recognises as one class (Covid-19) or the other (non-Covid-19). The general scheme of a CNN model has a feature extractor in the first phase and then a classification phase (Fig. 3) Our DLA was designed based on the CNN model (Fig. 4).

In general, the transposed convolutional and ReLU layers are the feature learning stage which is known as freeze layer, and the few last layers, the unfreeze layers, are the classifier, also known as the fine-tuning stage. The ReLu activation function transforms the value results of a neuron, by $y = (0, x)$, and clamps down any negative values to 0, whereas the positive values remain untouched. The result of this mathematical transformation was considered as the output of the ongoing layer and input to the next layer. In addition, all the convolutional layers used the same size of 224 × 224 pixel with 64 filters for each layer.

The feature extraction phase can improve the accuracy of a model by extracting features from the image data. This phase decreases the dimensionality of image data by eliminating redundant data. Consequently, it increases the training speed and obtains newly generated features by combining and/or transforming the original feature set.

3.3. Deep transfer learning

The general problem of image classification is defined as follows:

- There are K possible image classes. A set {0, 1, …, $K-1$} defines the labels of the different classes (example: 0 = “non-Covid-19″ and 1 = “Covid-19".)
- We have a collection of N input images: {Xi} i {1, …, N }.
- The classes of N images are known in advance: each image (Xi) is labelled by $yi \in \{0, 1, \ldots, K-1\}$.
- The goal is to correctly classify a new image, whose class is not known. We want to find the right label.

In CNN, the nodes in the hidden layers do not consistently distribute their output with every node in the next layer. Deep transfer learning

Table 2
Classified number of images based on the Covid-19 dataset.

Dataset	Covid-19	non-Covid-19	Total
Train	466	542	1008
Test	210	262	472
Total	676	804	1480

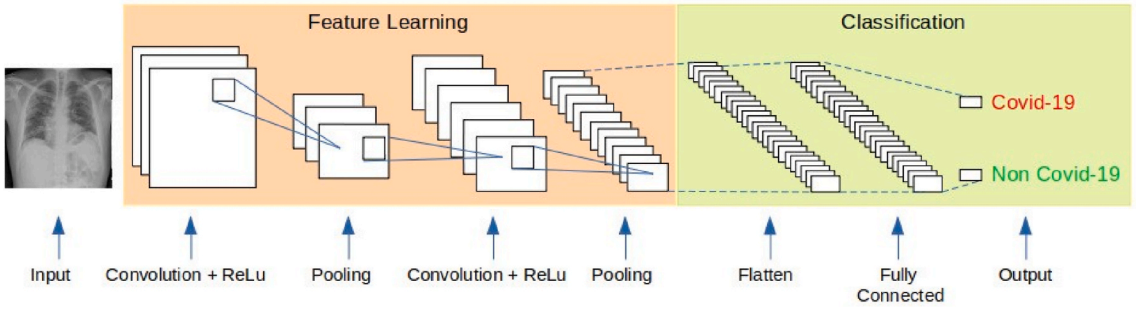

Fig. 3. Proposed DLA based on CNN.

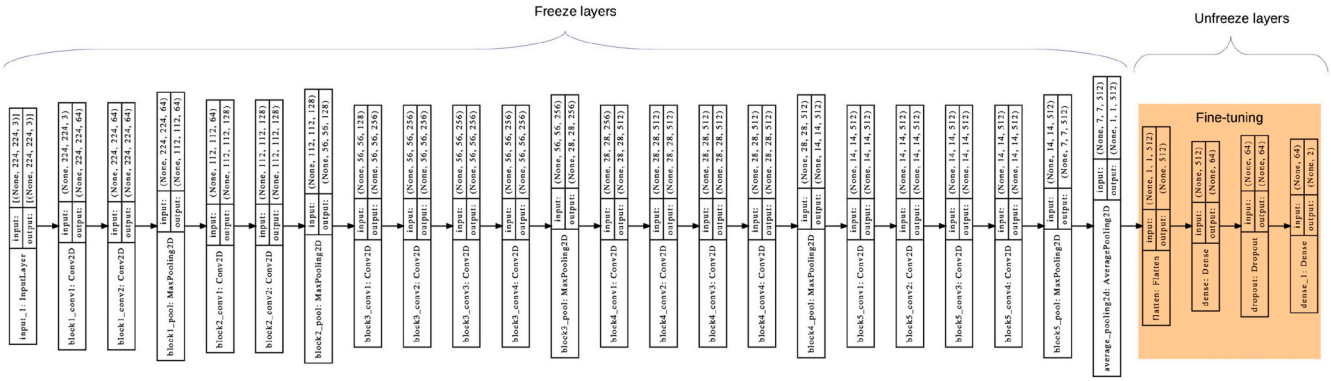

Fig. 4. Layers of the proposed model.

allows machines to recognise and extract features from images. This means that deep transfer-learning machines can discover features to look for in images by analysing sets of pictures.

Using deep learning, the transfer-learning method can solve the image classification problem. Few state-of-the-art results of image classification refer to the works of [2,5,7,18,19,25]; which are based on transfer learning solutions.

There are two different approaches in transfer-learning: model approach and pretrained model approach. However, in this phase, we choose to apply a model approach. For this purpose, we select a related predictive modeling problem that presents some relations between input data, output data and/or images learned during the mapping process from input to output data. The proposed model comprises layered compositions that learn original features at different layers, and these layers are eventually connected to the last layer. We use domain, task, and marginal probabilities to present the transfer-learning framework. The framework is defined as follows: domain D includes 2 tuple defining feature space χ and marginal probability distribution $P(X)$, where X denotes sample data, $X = \{x1, \ldots, xn\}$, $xi \in \chi$, where xi denotes a specific vector. Hence, the domain can be represented as $D = \{, P(X)\}$. The transfer learning layers reconstruct images from a new domain task into a n-dimensional vector based on its hidden characters, thereby allowing to extract features from a new domain task and applying the knowledge from a source-domain task. Next, we develop a source model for the previous phase to guarantee that feature learning has been carried out. Later, we reuse the model, which means that the source model can be used as the basis for a model on the next task of interest. Further phase is the tune model, as the model needs to be adjusted based on the input and output data.

3.4. Data augmentation and fine-tuning

A large dataset is important for a deep learning model, especially for its performance. Training a deep learning model based on a large dataset can result in an optimal model, and the data augmentation method might create image variations which help improve the ability of the fit model to generalise the model that has learned new images which have never been seen before. In our scheme, we apply image data augmentation to the training dataset instead of the validation or test dataset. In addition, data augmentation increases the size of the dataset images, which would be larger than the original. We can augment the image data by flipping images horizontally or vertically. We used some common data augmentation techniques. The parameter settings for data augmentation were as follows: rotation range of 15% which means that a range within which pictures are randomly rotated, and zoom range of 10% (maximum) for randomly zooming internal pictures, enabling horizontal flip randomly to flip half of the images horizontally and fill in the missing pixels with the nearest filled value, which refers to the technique used for filling in newly created pixels; the pixel can appear after a rotation or a width/height shift.

Fine-tuning is widely used for model reuse and is complementary to the preceding process, which is feature extraction. It helps unfreeze head layers of *basemodel* for future extraction purpose and mutually re-train it with a low learning rate to make the network model more suitable for the problem. The role of fine-tuning will be as follows: First, feed the neural network with the training data, *train_images* and *train_labels*. Then, the network will discover the correlation between images and labels. Finally, call the network to generate predictions for *test_images*, and confirm whether these predictions match the labels from *test_labels*.

The proposed model uses the SoftMax activation function in the fine-tuning phase which is used to determine the probability distribution of a set of numbers within an input vector. The output of function is a vector in which its set of values indicates the probability of an instance of a class. With regard to optimiser, Adam [11] with momentum value of 0.9 is used in our algorithm. This optimiser updates weight parameters at the training and fine-tuning phases. We implemented a dropout layer to select the best training steps. In the training phase, our scheme generates a batch of images, called the training batch. The batch size is the number

of samples processed before the model is updated. The number of epochs is the number of complete passes through the training dataset. The size of a batch should be equal to or greater than one and less than or equal to the number of samples in the training dataset. The Model was trained with a learning rate of 0.0001. However, the learning rate is a hyper-parameter that manages the extent to which the model is modified in response to the predicted error each time the model weights are renewed.

3.5. Detection and prediction

A key issue to detect Covid-19 based on X-ray images is that the number of objects in the foreground can differ across images. The proposed model presents a solution the classification issues of the prescribed object detection problem. Subsequently, using CNN layers to extract feature maps, the region proposal network yields several windows which are likely to accommodate an object. Then, the model reclaims feature maps inside each window, resizes them into fixed sizes, or the region of interest (RoI) pooling, and predicts the class probability.

Every pixel belongs to a particular class (either background or RoI) that is described by the same colour. Our scheme uses simple semantic segmentation with a threshold value where the pixel values will be contrasting for objects and background if there is a sharp contrast between them. Pixel values lower than or above the threshold value can be classified as an object or background.

As a proof-of-concept, a prototype using a web application was developed using the Flask platform based on Python, hoping that it can enrich the Covid-19 dataset and would be helpful for researchers, doctors, or paramedics in Indonesia. The purpose of this web application is to determine whether our proposed model can classify, detect and predict an X-ray image that has never been seen before. In other words, it was implemented to demonstrate the detection and prediction functions with regard to Covid-19 using a deep learning model (Fig. 5).

3.6. Visualisation

3.6.1. X-ray image enhancement

The main objective of this procedure is to enhance the visualisation of X-ray images by using image enhancement methods, i.e., contrast limited adaptive histogram equalization (CLAHE) and Otsu thresholding. The purpose of X-ray image enhancement, in our case, is to improve the visual interpretability of an X-ray image by intensifying the apparent distinction between the features in the image.

The CLAHE technique is a histogram-based method used to enhance the contrast of an image. This technique calculates the histogram for the region around each pixel in the image, develops the local contrast and improves the edges in each region.

Since adaptive histogram equalization (AHE) may overamplify the noise of an image, CLAHE avoids this by limiting the amplification which can be defined as follows:

Fig. 5. Prototype of the UNAS-Net application.

$$\beta = \frac{A}{G}\left(1 + \frac{\alpha}{100}(S_{max} - 1)\right) \quad (1)$$

where A is the size of area, G refers to the grayscale value, and α is the clip factor that serves as the addition of the limit of a histogram (from 0 to 100).

Even though CLAHE may take a high load, the enhanced image method is considerably better than the unprocessed one, because it shows the hidden features of an image by enhancing its edges and amplifying its visibility. In our case, we apply few input parameters to enhance an image: window size (WS) is the size of the rectangular area around the pixel to be refined, clip limit (CL) is the maximum number of pixels in the histogram, and clipping iterations (CI) that refer to the number of clipping repetitions.

Otsu thresholding is a method to reform an image of grey to black and white depending on the threshold value ratio to improve the X-ray image quality. In our study, we apply Otsu thresholding by converting an image into a grey image and then binarising it.

3.6.2. Saliency maps

To guarantee that the proposed model points to reasonable aspects of the images [9,20], we performed a saliency map. It is estimated by computing the gradient of the output prediction with respect to the input image, i.e., supposing that a pixel is changed, how much will it change the prediction. In addition, to smooth out results of the saliency map, the image is blurred using a Gaussian kernel. In this phase, a score system based on two parameters is used to exhibit the disease severity level, i.e., geographic extent and degree of opacity, as reported by [22]. A score of 0–4 is designated as pneumonia severity level for each lung determined based on the extent of involvement of ground-glass opacities as follows: 0 for no involvement, 1 for $<$ 25% involvement, 2 for 25%–50% involvement, 3 for 50%–75% involvement, and 4 for $>$ 75% involvement. The scores for each lung were enumerated to demonstrate the final severity score (ranging from 0 to 8, for right and left lungs collectively). The degree of opacity was scored as follows: 0, no opacity; 1, ground glass opacity; 2, consolidation; and 3, white-out. Hence, the opacity score for both the left and right lungs ranged from 0 to 6.

4. Performance evaluation and comparison

In this section, we investigated and compared the proposed model, known as UNAS-Net, with few deep transfer-learning models such as Xception [2], Inception [19], DenseNet [7]), ResNet50 [5] and VGG16 [18]. Moreover, we analysed CLAHE and saliency maps.

The proposed model was designed using Keras and Tensorflow in a Python environment. All simulations were conducted on a computer server equipped with an Intel socket 36 core hyperthread, 32 GB of RAM, and graphics processing units. We measured and compared the classification performance of the models using the following metrics:

- *Accuracy*. This is a metric to evaluate classification models, calculated as the ratio of number of correct predictions on the total number of predictions.

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \quad (2)$$

where *TP* is True Positive, *TN* is True Negative, *FP* means False Positive, and *FN* refers to False Negative.

- *Sensitivity* (*Recall*). This metric measures the rate of negatives that are correctly detected.

$$Sensitivity = \frac{TP}{TP + FN} \quad (3)$$

- *Specificity*. This indicator is represented as the proportion of actual negatives, or otherwise stated as true negative rate.

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

- *F-score*. This indicator is a measure of the accuracy of a binary model to compare diversity and similarity of performance.

$$Precesion = \frac{TP}{TP + FP} \quad (5)$$

$$Fscore = \frac{(\beta^2 + 1) \times Precision \times Sensitivity}{\beta^2 \times Precision + Sensitivity} \quad (6)$$

- *Area under curve (AUC)*. This metric measures the performance of the classifier.

$$AUC = \frac{Sensitivity + Specificity}{2} \quad (7)$$

- *Confusion matrix*. It summarises of prediction results based on a classification problem, and shows ways in which the classification model is confused while making predictions.

The performance results of all the trained models are listed in Table 3. In terms of accuracy, DenseNet attained 100%, followed by VGG16 and Inception with 98.18%, the proposed scheme achieved 97.36%, and ResNet50 achieved a lower accuracy of 85.45%. This accuracy level indicates model accuracy in classifying X-ray images as Covid-19 or non-Covid-19. This implies that in this training, the proposed model, UNAS-Net, exhibits a classification accuracy of 97.36%.

We compared the sensitivity (recall) of the trained models and noticed that the improvement in sensitivity caused by the use of data augmentation and fine-tuning methods is consistent with specificity. The correct sensitivity and specificity were obtained using DenseNet. This is because the DenseNet architecture manages the residual structure to its maximum by constructing every layer that is densely associated with its subsequent layers.

UNAS-Net achieved 95.24% sensitivity and 100% specificity, which is equal to Inception and VGG16, in such a way that the proposed model is excellent at predicting, implying that Covid-19 or non-Covid-19 can be accurately detected. However, ResNet50 exhibits the lowest sensitivity, which implies that Covid-19 cannot be detected by the model even when non-Covid-19 can be detected accurately. Consequently, it can lead to a higher false negative rate, where patients with Covid-19 (positive) are predicted as non-Covid-19 (negative).

The proposed model achieved an F-score of 98% which corresponded to Inception and VGG16. Thus, we can conclude that they exhibit good precision and sensitivity. Nonetheless, ResNet50 attained an F-score only 78%, indicating that it exhibits lower precision and lower sensitivity in this study.

In terms of the validation of training loss and accuracy, as shown in Fig. 6, the proposed model does not underfit much as ResNet50. At the basic level, the loss curve represents how good or bad a given model is classified. As we can see, the proposed model has a smaller loss, which is

Table 3
Comparison Results of models training on the dataset.

Model	Accuracy	Sensitivity	Specificity	F-Score	AUC
UNAS-Net	0.9736	0.9524	1.00	0.98	0.9808
Inception	0.9818	0.9524	1.00	0.98	0.9805
Xception	0.9636	0.9524	0.9706	0.95	0.9987
DenseNet	1.00	1.00	1.00	1.00	1.00
ResNet50	0.8545	0.6667	0.9706	0.78	0.9263
VGG16	0.9818	0.9524	1.00	0.98	0.9997

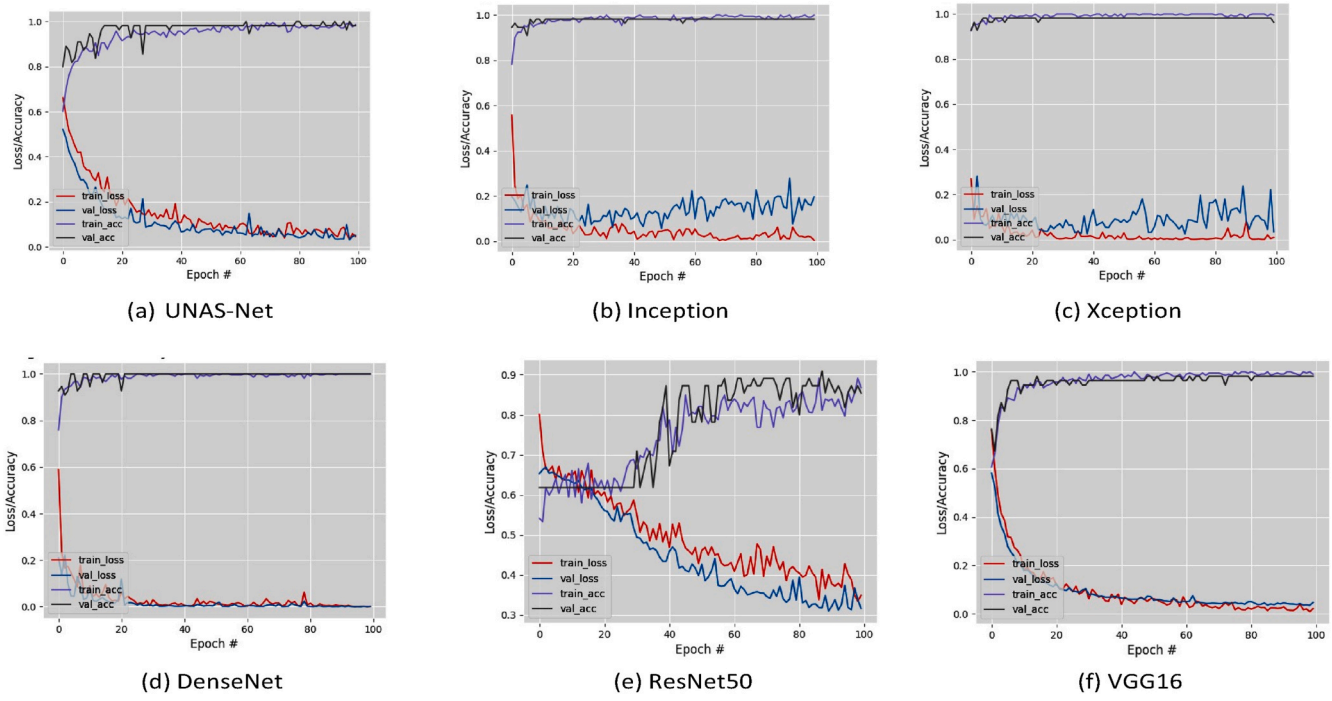

Fig. 6. Training loss and accuracy comparison.

almost similar to that of VGG16. This means that they have a better classifier. Moreover, from the epoch of 0 to approximately 60, our validation loss is lower than the training loss. A quick observation is that this phenomenon is due to regularisation applied during the training phase, but not during the validation or testing phase. In addition, the training loss was measured at the time of any epoch, while the validation loss was measured after each epoch. It is feasible to enhance the model by increasing the number of epochs and batch size. However, ResNet50 suffers in terms of performance from the initiation of training (Fig. 6(e)). For the area under the curve (AUC), the results obtained with regard to various layers of the models are shown in Fig. 7 DenseNet and VGG16 have exhibited AUC values near 1, similar to our scheme. This implies that they possess an effective measurement process of distinguishing between the positive and negative classes. Inception and Xception have exhibited moderate results. In contrast, ResNet50 is likely to be a poor model because it has an AUC near to 0, which implies that ResNet50 includes an ineffective measurement method of classification, or in other words, it has no class separation efficiency.

Table 4 compares the confusion matrix among the proposed model (UNAS-Net), Inception, Xception, DenseNet, ResNet50 and VGG16. We

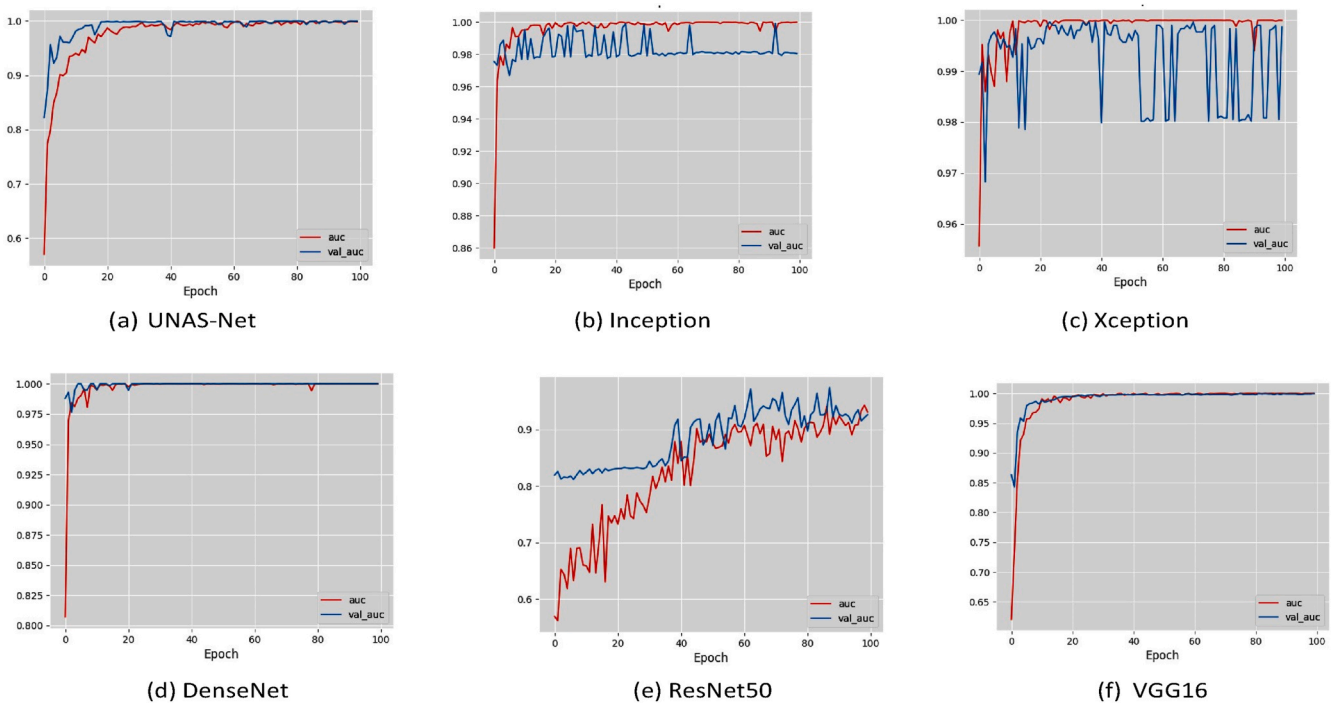

Fig. 7. Auc comparison.

Table 4
Confusion matrix.

	UNAS-Net		Inception		Xception		DenseNet		ResNet50		VGG16	
	True	False	True	False	True	False	True	False	True	False	True	False
Positive	20	1	20	1	20	1	21	0	14	7	20	1
Negative	33	1	34	0	33	1	34	0	33	1	34	0

found that DenseNet achieved 100% classification accuracy. UNAS-Net outperformed ResNet50 and achieved the same results as those of Inception, Xception, and VGG16. The proposed model exhibits a true positive (TP) of 20, implying that 20 positive Covid-19 images are accurately classified and only one image is incorrectly classified as it belonged to the positive class. These results are comparable with those of Inception, Xception and VGG16. In addition, the proposed model has a true negative (TN) of 33, implying that 33 negative Covid-19 images were accurately classified. Moreover, as per ResNet50, 14 positive Covid-19 images are accurate, and 7 images are falsely classified. However, ResNet50 have correctly detected 33 of 34 negative Covid-19 images.

In terms of visualisation and image enhancement, we conducted several investigations on the qualitative analysis of the predicted scores using some parameters. As CLAHE enumerates a histogram to all pixels, its complexity becomes high and requires a certain amount of processing time to accomplish its work. We considered a wide range of values for input parameters to see and compare effects on an image. In this study, we used several different parameters: e.g., WS of 8–15, CL of 4–150, and CI of 1–5. However, the input parameters significantly affect the results (Fig. 8). Based on our experiments, we found that the optimal input parameter combination was WS = 100, CL = 150, and CI = 1 (Fig. 8(a), (b), and 8(c)). Moreover, the CLAHE image is the threshold value calculated using Otsu to load its CL value. Fig. 9 depicts enhanced X-ray images of the Covid-19 positive case with CLAHE threshold and Otsu threshold.

Fig. 10 shows the saliency maps of the images that have never been observed by the model during the training phase. The images were correctly predicted by the model. For most of the results, the proposed model correctly points to the opaque areas of the lungs. The predictions are demonstrated with a saliency map produced by calculating the gradient of the output prediction and then blurred using a 5 x 5 Gaussian kernel. For instance, Fig. 10(a) shows a heatmap of an infant CXR of Covid-19 positive case with a geographic extent score of 1.475 and opacity of 0.9121. Moreover, as shown in Fig. 10(b), for an adult CXR, the proposed model predicts severity by generating a geographic extent score of 1.973 and opacity of 2.442. This implies that the proposed model can predict the ground glass by focusing on the lungs.

The proposed work, UNAS-Net, focuses on predicting the ground glass and severity of the lungs based on a range of clinical indicators. The

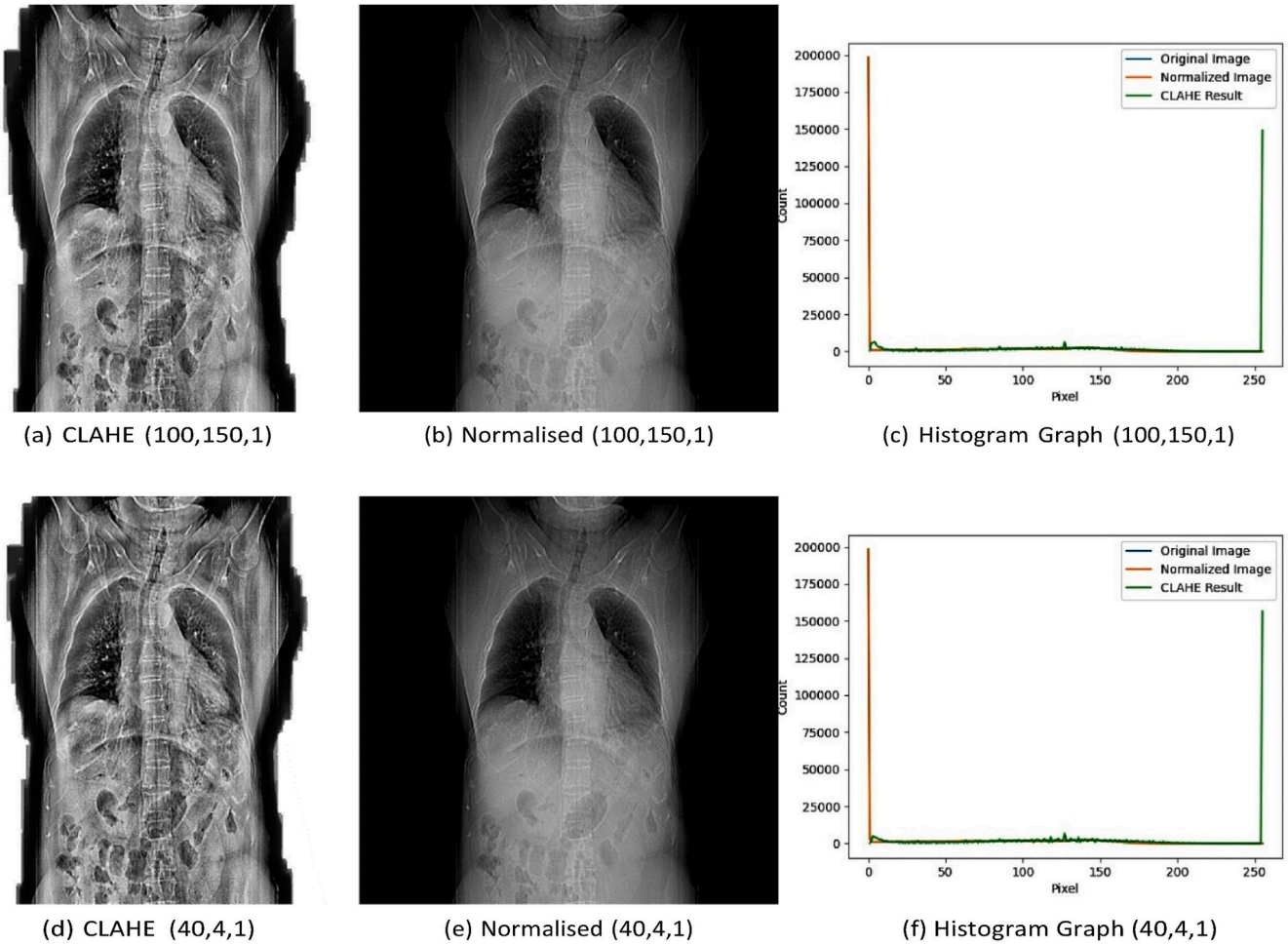

Fig. 8. CLAHE comparison between parameters WS: 100, CL: 150, CI: 1 and WS: 40, CL: 4, CI: 1.

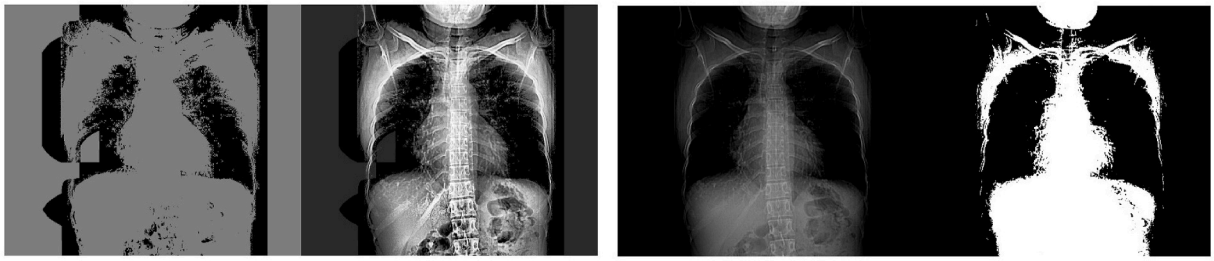

Fig. 9. Enhanced image with (a) CLAHE threshold and (b) with Otsu threshold.

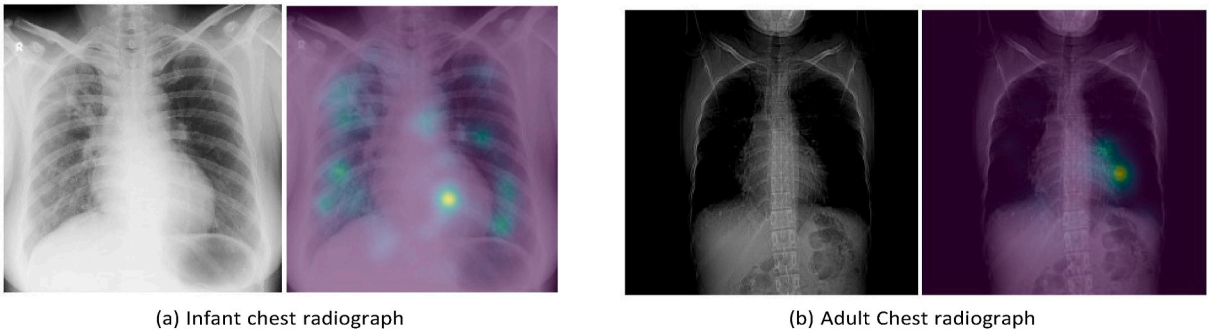

Fig. 10. Heatmap comparison. (A) Infant with Geographic Extent Value: 1.475, Opacity: 0.9121, and (B) Adult Chest X-ray with Geographic Extent Value: 1.973, Opacity: 2.442.

proposed model can be integrated and enhanced to other models and potentially assist in making decisions for those who need it.

The challenge is when the number of labelled images is small, and we need to create a model that helps label images and achieve acceptable performance. In our case, building a predictive model for Covid-19 based on X-ray images with the lack of a public dataset, particularly Indonesian X-ray images, made it difficult to manage large-scale evaluations. The small number of dataset counters proper cohort selection, which is a limitation to this study. Nonetheless, we use a deep transfer-learning model, which was trained based on a large dataset with similar functions and was equipped with an unbiased Covid-19 feature extractor. Thus, it reduces overfitting of the proposed model.

The performance evaluation can be improved with the condition that we are able to achieve a new cohort of labelled images to determine the generalisation of the proposed model.

5. Conclusions

A platform based on deep learning to predict Covid-19 severity in Indonesia was developed. Based on preliminary results, we observed several interesting observations during the performance evaluation. The overall performance metrics for DenseNet were perfect, and for UNAS-Net was adequately high, which is comparable to other models. The DenseNet model outperforms other models with regard to the classification task. Data augmentation and fine-tuning successfully handed the network to enhance the overall performance, especially for our case with limited datasets.

The results of the study on the CLAHE and Otsu algorithms shows that the CLAHE algorithm exhibits the best result for our system in terms of image visualisation. The only drawback of this algorithm is the slow running time. In addition, the proposed model focuses on classifying and predicting lung severity.

Further work that can be considered is the improvement of the proposed model by enhancing quality and coherent images more, thereby tuning the hyper-parameters of the model, which can improve the accuracy.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

Research paper document and the results of the research project funded by the Ministry of Research and Technology/National Research and Innovation Agency of Republic Indonesia for Acceleration of Handling COVID-19 Funding Program with contract number: 71/FI/P-KCOVID19.2B3/IX/2020.

References

[1] Chest X-ray Penumonia Dataset. https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia. [Accessed 14 June 2020].
[2] Chollet F. Xception: deep learning with depthwise separable convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR); 2017.
[3] Cohen JP, Morrison P, Dao L. Covid-19 image data collection. arXiv C 2020:11597. URL, https://github.com/ieee8023/covid-chestxray-dataset.
[4] Gozes O, Frid-Adar M, Greenspan H, Browning PD, Zhang H, Ji W, Bernheim A, Siegel E. Rapid AI development cycle for the coronavirus (COVID-19) pandemic: initial results for automated detection and patient monitoring using deep learning CT image analysis. arXiv 2020;2003:05037.
[5] He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. CVPR 2016:770–8.
[6] Hendry, Chen RC. Automatic License Plate Recognition via sliding-window darknet-YOLO deep learning. Image Vis Comput 2019;87:47–56. https://doi.org/10.1016/j.imavis.2019.04.007.
[7] Huang G, Liu Z, Der Maaten LV, Weinberger KQ. Densely connected convolutional networks. In: Proceedings of the 2017 IEEE conference on computer vision and pattern recognition (CVPR); 2017. p. 2261–9.
[8] Jin Y, Cai L, Cheng Z, et al. A rapid advice guideline for the diagnosis and treatment of 2019 novel coronavirus (2019-nCoV) infected pneumonia (standard version). Milit. Med. Res. 2020. https://doi.org/10.1186/s40779-020-0233-6.
[9] John RZ, Marcus AB, Manway L, Anthony BC, Joseph JT, Eric Karl O. Variable generalization performance of a deep learning model to detect pneumonia in chest radiographs: a cross-sectional study. PLoS Med 2018;15. https://doi.org/10.1371/journal.pmed.1002683. URL.

[10] Khan AI, Shah JL, Bhat MM. CoroNet: a deep neural network for detection and diagnosis of COVID-19 from chest x-ray images. Comput Methods Progr Biomed 2020;196:105581. https://doi.org/10.1016/j.cmpb.2020.105581.
[11] Kingma DP, Ba J. Adam: a method for stochastic optimization. arXiv 2014. 1412.6980.
[12] LeCun Y, Bengio Y, Hinton G. Deep learning. Nature 2015;521:436–44.
[13] Li L, et al. Artificial intelligence distinguishes covid-19 from community acquired pneumonia on chest ct. Radiology 2020:200905. https://doi.org/10.1148/radiol.2020200905.
[14] Li X, Li C, Zhu D. Covid-mobilexpert: on-device covid-19 screening using snapshots of chest x-ray. arXiv 2020;2004:03042.
[15] Maghdid H, Ghafoor KZ, Sadiq AS, Curran K, Rabie K. A novel AI-enabled framework to diagnose coronavirus COVID-19 using smartphone embedded sensors: design study. arXiv 2020;2003:07434.
[16] Ozturk T, Talo M, Yildirim EA, Baloglu UB, Yildirim O, Acharya UR. Automated detection of COVID-19 cases using deep neural networks with X-ray images. Comput Biol Med 2020 Apr 28:103792.
[17] Shi F, et al. Large-scale screening of covid-19 from community acquired pneumonia using infection size-aware classification. arXiv 2020;2003:09860.
[18] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. arXiv 2014;2014. 1409.1556.
[19] Szegedy C, Vanhoucke V, Ioffe S, Shlens J, Wojna ZB. Rethinking the inception architecture for computer vision. In: Proceedings of the 2016 IEEE conference on computer vision and pattern recognition (CVPR); 2016. p. 2818–26.
[20] Viviano JD, Simpson B, Dutil F, Bengio Y, Cohen JP. Saliency is a possible red herring when diagnosing poor generalization. 2019.
[21] Wang L, Lin ZQ, Wong A. COVID-net: a tailored deep convolutional neural network design for detection of COVID-19 cases from chest X-ray images. preprint arXiv 2020;2003:09871 [eess.IV].
[22] Wong HYF, et al. Frequency and distribution of chest radiographic findings in COVID-19 positive patients. Radiology 2019;3. https://doi.org/10.1148/radiol.2020201160.
[23] Xu X, et al. Deep learning system to screen coronavirus disease 2019 pneumonia. arXiv 2020;2002:09334.
[24] Yazid MHA, Satria MH, Talib S, Azman N. Artificial neural network parameter tuning framework for heart disease classification. In: 2018 5th international conference on electrical engineering, computer science and informatics (EECSI); 2018. p. 674–9.
[25] Yosinski J, Clune J, Bengio Y, Lipson H. How transferable are features in deep neural networks? Adv Neural. Inf. Process. Syst. 2014:3320–8.

Abdusy Syarif is an associate professor at Universitas Nasional, Jakarta, Indonesia. He received his Bachelor degree in 1999, and his Master degree from Universitas Indonesia in 2005. He obtained his PhD in 2015 from University of Haute Alsace, France. During 2016–2019, he was a Post-doc researcher at University of Haute-Alsace and LINEACT-CESI, France. He is the author/co-author of 1 intellectual property right, more than 30 international publications in refereed journals and conferences, some of them achieved Best Paper Awards. His research interests involve Machine Learning, Artificial Intelligence, Multimedia System, Embedded System, Internet of Things, Wireless Sensor Network, Routing Protocol, and QoS. He has been serving as Technical Program Committee and Reviewer for some International Conferences and Journals since 2015.

Novi Azman finished his undergraduate with majoring in Electrical Engineering at Universitas Nasional in 2000. Then continued his master degree at Universitas Indonesia in 2005. He is currently completing his doctoral degree at Universitas Teknikal Malaysia Malaka, Malaysia in the field of Biomedical Engineering. His experience involve Telecommunication and Biomedical **Engineering**, especially in Telemonitoring or Internet of Things in Healthcare. Presently, he is a lecturer in Electrical Engineering and Dean of Faculty of Engineering and Science at Universitas Nasional, Indonesia.

Viktor Vekky Ronal Repi achieved his undergraduate with majoring in Engineering Physics at Universi-tas Nasional in 2001, and his master degree at Universitas Indonesia with specialized in Optoelectronic in 2005. In 2014, he finished his doctoral degree in Material Science at Universitas Indonesia. His re-search interest involves Absorber Material for electro-magnetic waves. He is currently a lecturer in Engineer-ing Physics and Vice Dean of the Faculty of Engineering and Sci-ence at Universitas Nasional, Indonesia.

Ernawati Sinaga is a Full Professor in School of Biology Universitas Nasional, Jakarta, Indonesia, and now serves as Vice Rector for Research, Community Services and Collaboration, and as Director of Center for Medicinal Plants Research in Universitas Nasional. She received her PhD from Department of Chemistry University of Indonesia with research emphasis on molecular biology of drug delivery conducted as joint research with Department of Pharmaceutical Chemistry University of Kansas, Lawrence, Kansas, USA. She received her Master degree in Biomedical Sciences and her Bachelor degree from School of Pharmacy from University of Indonesia, Jakarta, Indonesia.

Muhamad Asvial received Ir. (Insinyur) degree in Electrical Engineering from Electrical Engineering Department, University of Indonesia in 1993, MSc degree from Keio University, Japan in 1998 and PhD degree from University of Surrey, UK in 2003. He is currently a researcher and lecturer at Electrical Engineering Department Universitas Indonesia. He was also a guest lecturer at University Duisburg Essen, Germany in 2005–2006. His research interests include mobile communication (terrestrial and satellite communication), HAPs Network, Genetic Algorithm Applications, Broadband and Ultra Wide Band Communication System. He published more than 93 papers in several international journals and conferences. He is a member of the IEEE, the IEE and the AIAA.