UDC 616.9: 004.42 **doi: 10.32620/reks.2025.1.12**

Muhammad IRHAMSYAH[1,2], Qurrata A'YUNI[2], Khairun SADDAMI[2], Nasaruddin NASARUDDIN[2], Khairul MUNADI[2], Fitri ARNIA[2]

[1] ***Doctoral Program, School of Engineering, Universitas Syiah Kuala, Banda Aceh Indonesia***
[2] ***Department of Electrical and Computer Engineering, Faculty of Engineering, Universitas Syiah Kuala, Banda Aceh Indonesia***

IMPACT OF USING VARIOUS X-RAY DATASET IN DETECTING TUBERCULOSIS BASED ON DEEP LEARNING

*The **subject matter** is that the characteristics of tuberculosis are difficult to study visually. Therefore, a computer-aided system based on deep learning can be applied to X-ray image recognition. Many studies have been conducted in this area but have yet to achieve a high accuracy rate. The **goal** of this study is to determine the effect of using various datasets in developing deep learning models. The **tasks** to be solved include exploring various deep learning architectures and deep fine-tuning hyperparameters, as well as using various dataset sources. The **method** used is the development of a deep learning model of convolutional neural network (CNN) using transfer learning to classify X-ray images into binary classes of normal and tuberculosis (TB). The CNN architectures used are the pretrained networks of ResNet and EfficientNet, along with their variants. The pre-trained network was trained on a dataset obtained from four sources: Shenzhen, Montgomery, RSNA CXR, and Belarus. The dataset is divided into three schemes: Scheme one consists of the Shenzhen dataset with low-quality X-ray images; Scheme two is the Montgomery, RSNA, and Belarus datasets that show good contrast in the indicated TB area; and Scheme three contains datasets from all sources to allow for more datasets to be learned. The augmentation, dropout, and L2 regularization methods were also applied to enhance learning performance. The following **results** were obtained: the models performed better with the high-quality X-ray images in Scheme Two but not with the large dataset in Scheme Three. Regarding network performance, the models resulting from ResNet-101 and EfficientNetB0 outperformed the others with good fit learning and capability in recognizing X-ray images with an accuracy rate of 99.2%. In **conclusion**, the best approach to enhance learning performance is to use high-quality input and apply regularizations.*

Keywords: *Tuberculosis; Covolutional Neural Network; ResNet; EfficientNet.*

1. Introduction

1.1. Research Motivation

Tuberculosis (TB) is a chronic infectious disease caused by Mycobacterium tuberculosis (Mtb). This disease has become a global health threat because of the high number of cases that were resistant to multiple drugs and extremely difficult to treat [1]. According to the World Health Organization (WHO), in 2018, nearly 10 million people were diagnosed with tuberculosis, of which 1.45 million died [2]. The limited number of doctors and equipment makes it difficult for many people to detect TB disease early; thus, patients only become aware of the disease when its symptoms are severe enough. Early detection is very useful so that infected individuals can start treatment immediately without waiting for their symptoms to worsen. This has motivated work on developing a computer-aided system that is user-friendly and can assist health workers in detecting TB disease quickly, supporting early treatment, and preventing the spread of this infectious disease.

Deep learning (DL), particularly convolutional neural networks (CNNs), has shown potential in extracting useful features for image classification tasks. The feature extraction process requires transfer learning, where a previously trained CNN model learns general features from a large-scale dataset like ImageNet and then transfers that knowledge to the required task [3]. Toraman et al. [4] stated that the images on X-rays must be examined and interpreted in detail by specialists. Yusoff et al. [5] reported that the classification of tuberculosis (TB) based on chest X-ray (CXR) was a time-consuming procedure that required expert interpretation. However, automated TB classification using CXR could become a significant clinical utility in addressing this issue due to its association with innovative technology.

T. Rahman et al. [6] used nine different CNN architectures to classify TB and non-TB cases as normal. ResNet-101 achieved 94.55% accuracy. Munadi et al. [7] implemented the ResNet18, ResNet50, and EfficientNetB4 architectures to train TB images. EfficientNetB4

ISSN 1814-4225 (print)
ISSN 2663-2012 (online)

achieved the highest scores, reaching 89.92% and 94.8% in classification accuracy and AUC (Area Under Curve) scores, respectively. Harahap et al. [8] constructed a CNN classifier on an imbalanced dataset consisting of 3500 normal and 718 tuberculosis cases. They obtained an accuracy rate of 98%. Nafisah et al. [9] applied various CNN models and compared their classification performances. They found that ResNet-50 achieved an accuracy of 79.5%, and EfficientNetB3 achieved an accuracy of 89.4%.

Chowdhury et al. [10] stated that EfficientNet is known for its high accuracy with fewer parameters and faster than the best existing CNN architectures. Oloko-Oba and S. Viriri [11] applied their proposed CNN model trained on Montgomery County (MC) Tuberculosis radiography and achieved a validation accuracy of 87.1%. Then, they applied five variants of EfficientNets [12], and they demonstrated that EfficientNet-B4 achieved the best accuracy of 92.%.

1.2. Objectives and State-of-the-art Approaches

Based on our literature study on existing similar works, various techniques such as augmentation [13] [14], image quality enhancement [15], and segmentation [16], were employed. Previous studies have used various combinations of datasets to develop deep learning-based models [17 - 19]. Pasa et al. combined the Montgomery datasets of Shenzhen and Belarus [17], whereas Inbaraj et al., only combined Montgomery with Shenzhen [18]. Natarajan et al. used four public X-ray datasets for TB detection; however, they excluded the Montgomery dataset. Therefore, to date, no study has explored the performance of deep learning networks trained using various X-ray datasets in the development of deep learning models. This work provides knowledge about the impact of using various X-ray datasets for developing early TB detection based on deep learning.

Therefore, we conducted research that provides state-of-the-art contributions as follows:

1. This study provides information about the characteristic features of chest X-rays that indicate tuberculosis. Deep analyses include the statistical analysis of the visual and statistical analysis of histograms and the mean average value of CXR from the datasets used in this research.

2. We develop a CNN model using the transfer learning of pre-trained ResNet and EfficientNet along with their variants. Deep training and fine-tuning of hyperparameter settings to recognize the CXR images resulted in a high-performance and novel model.

3. We performed a deep analysis of the simulation results using the training results of the learning curves and evaluation metrics of accuracy, sensitivity, and specificity using the confusion matrix of the good fit model.

4. We suggest some recommendations for future work to develop a robust CNN model to aid tuberculosis detection using CXR images.

The following sections describe our work. Section 2 explains the method, the dataset characteristics, preprocess, network architectures, hyperparameter settings, and regularization approach. Section 3 discusses network training results and testing performance in recognizing tuberculosis and normal CXR images. Section 4 provides recommendations for future work. Finally, Section 5 summarizes our research findings.

2. Materials and methods of research

2.1. Dataset

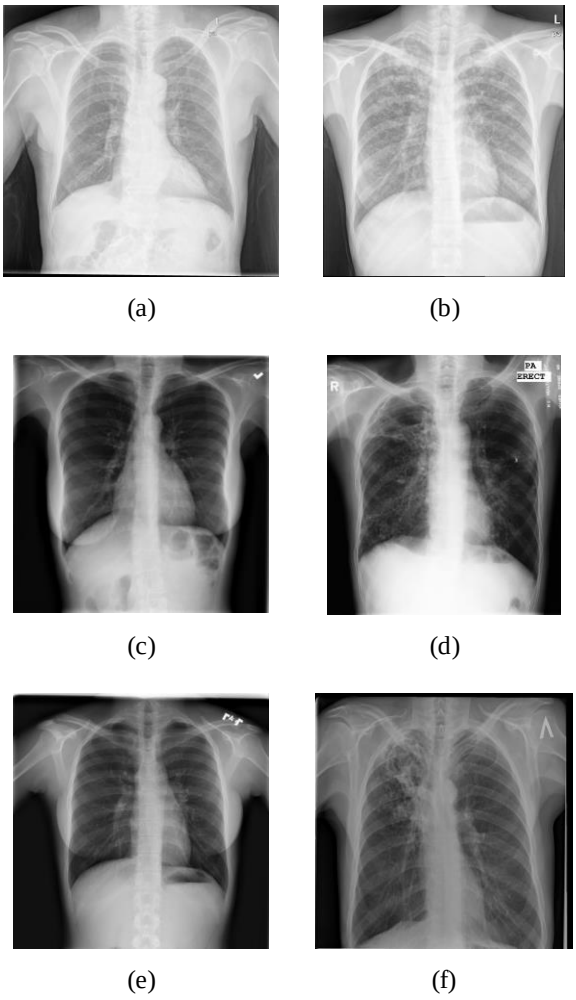
The characteristic features of tuberculosis are primarily observed in the lungs, specifically in chest X-rays. The typical radiological findings in chest X-rays (thorax) for tuberculosis are abnormalities in the apex (upper part) of the lungs. There are several types of radiographic images of pulmonary tuberculosis, such as infiltrates, cavities, and nodules. Infiltrates appear as small spots or dots scattered throughout the lungs. The cavities are rounded with visible walls. Nodules are small round lesions larger than infiltrates. If any of these three findings are present in the X-ray images may indicate tuberculosis. However, further examination is required to confirm the presence of Mycobacterium tuberculosis bacteria [20].

We utilized X-ray images obtained from Kaggle, consisting of four different sources: the Shenzhen dataset [21], the Montgomery dataset [22], the RSNA dataset [23], and the Belarus dataset [24]. The dataset comprised 1304 images. The Shenzhen and Montgomery datasets have two classes: normal and tuberculosis. The RSNA dataset contains only normal images, and the Belarus dataset exclusively contains tuberculosis images.

Figure 1 shows some X-ray images from each dataset source. In fact, visually, it was not easy to distinguish between TB and normal images in the Shenzhen dataset because they appeared very similar. However, on the Montgomery, RSNA, and Belarus datasets, there were clear differences between normal and TB images.

The pixel intensity values are shown in Figs. 2 and 3. In the Shenzhen data, both TB and normal images had pixel intensities above 100. This indicates that both datasets tend to have higher brightness levels compared to other normal datasets. Meanwhile, the Montgomery images showed more distinct differences. The images indicating tuberculosis showed high pixel intensity values above 100, indicating brighter areas in the image. On the other hand, normal Montgomery images tend to have

pixel intensity values approaching 0, indicating darker areas or closer to black in grayscale images. Thus, Montgomery's dataset provides better quality input for the learning approach than Shenzhen's.



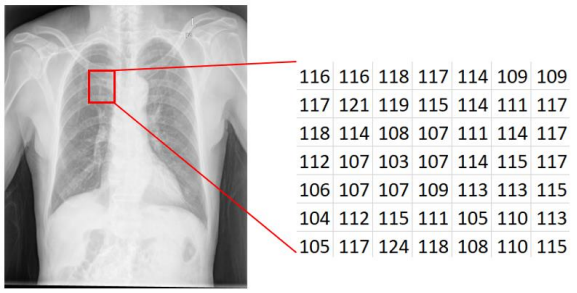
(a) (b)

(c) (d)

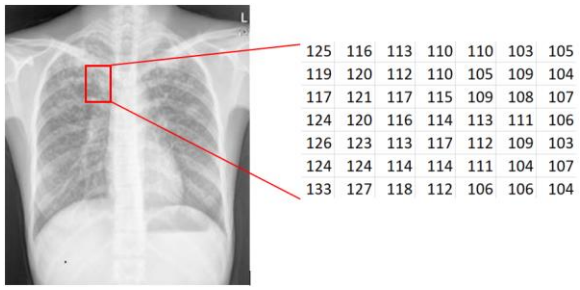
(e) (f)

Fig. 1. X-ray images of (a) normal Shenzhen, (b) tuberculosis Shenzhen, (c) normal Montgomery, (d) tuberculosis Montgomery, (e) normal RSNA, and (f) tuberculosis Belarus datasets

Furthermore, to gain more insight into the image characteristics, we applied a statistical approach to both datasets. As shown in Figs. 4, Fig. 5, Fig. 6, and Fig. 7, it can be observed that the pixel intensity distribution patterns in the histograms of tuberculosis images are similar to those of normal images in both the Montgomery and Shenzhen datasets. The intensity distribution in the histograms showed a similar range, with frequency concentration in the medium-to-high-intensity range and a significant peak in the high-intensity range (200–255), representing bone structures and other bright areas.

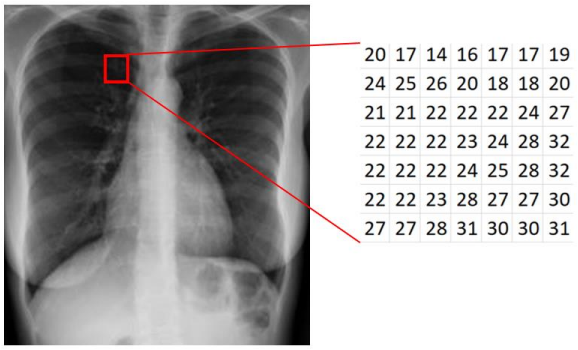


(a)

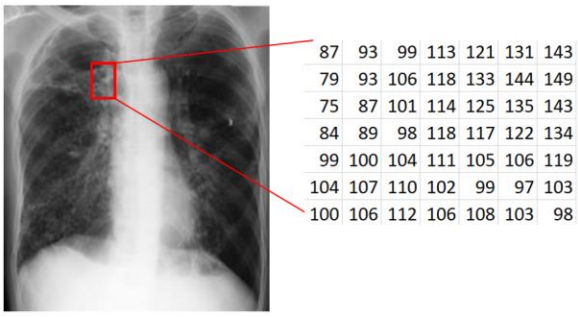


(b)

Fig. 2. Pixel value of X-ray image from Shenzhen dataset (a) normal and (b) tuberculosis



(a)



(b)

Fig. 3. Pixel value of X-ray image from Montgomery dataset (a) normal and (b) tuberculosis

ISSN 1814-4225 (print)
ISSN 2663-2012 (online)

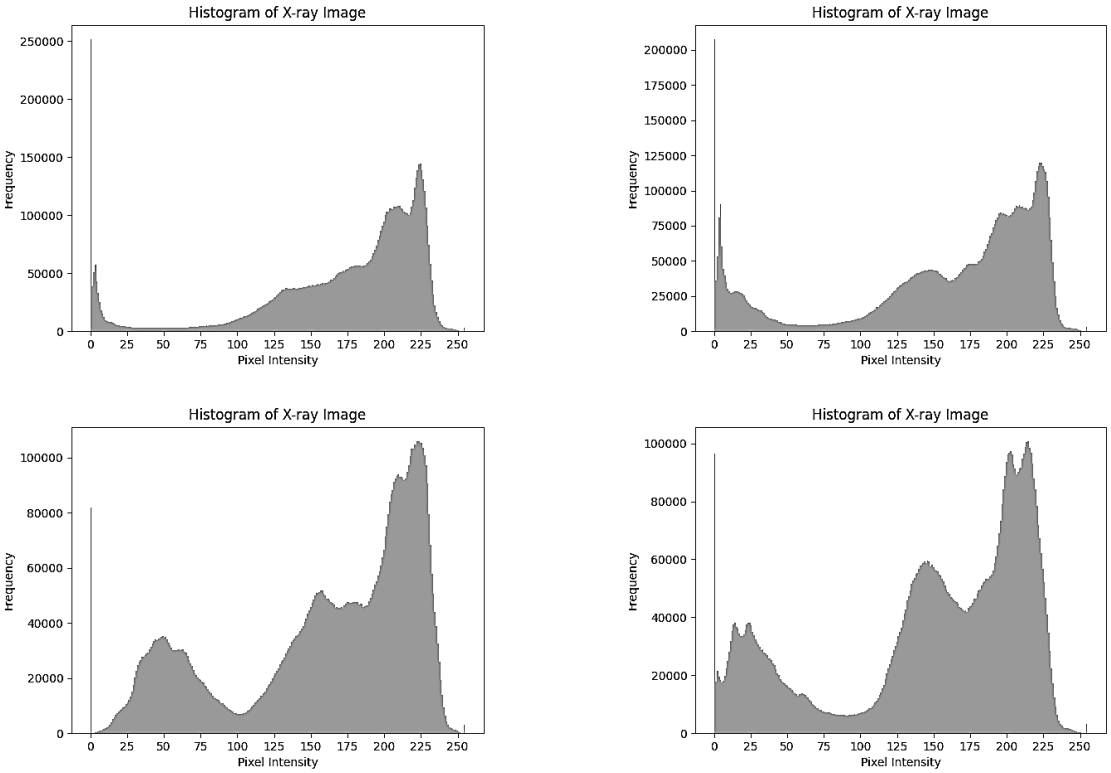


Fig. 4. Histogram of the Shenzhen normal image dataset

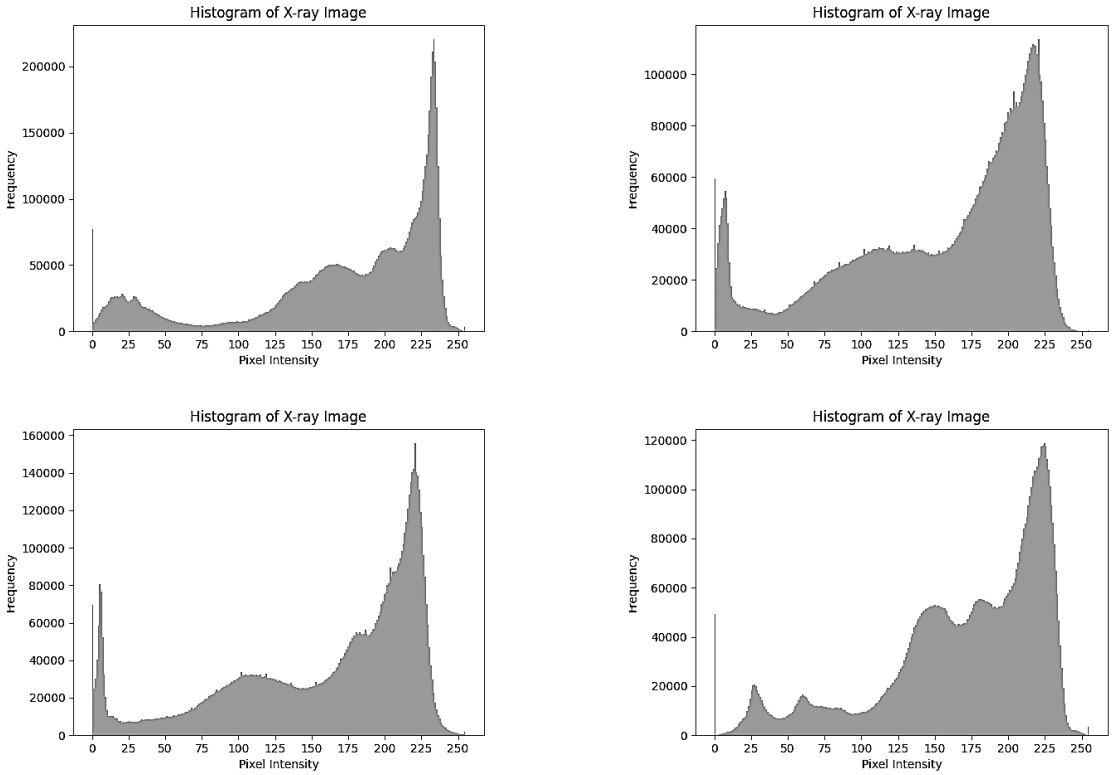


Fig. 5. Histogram of the Shenzhen tuberculosis image dataset

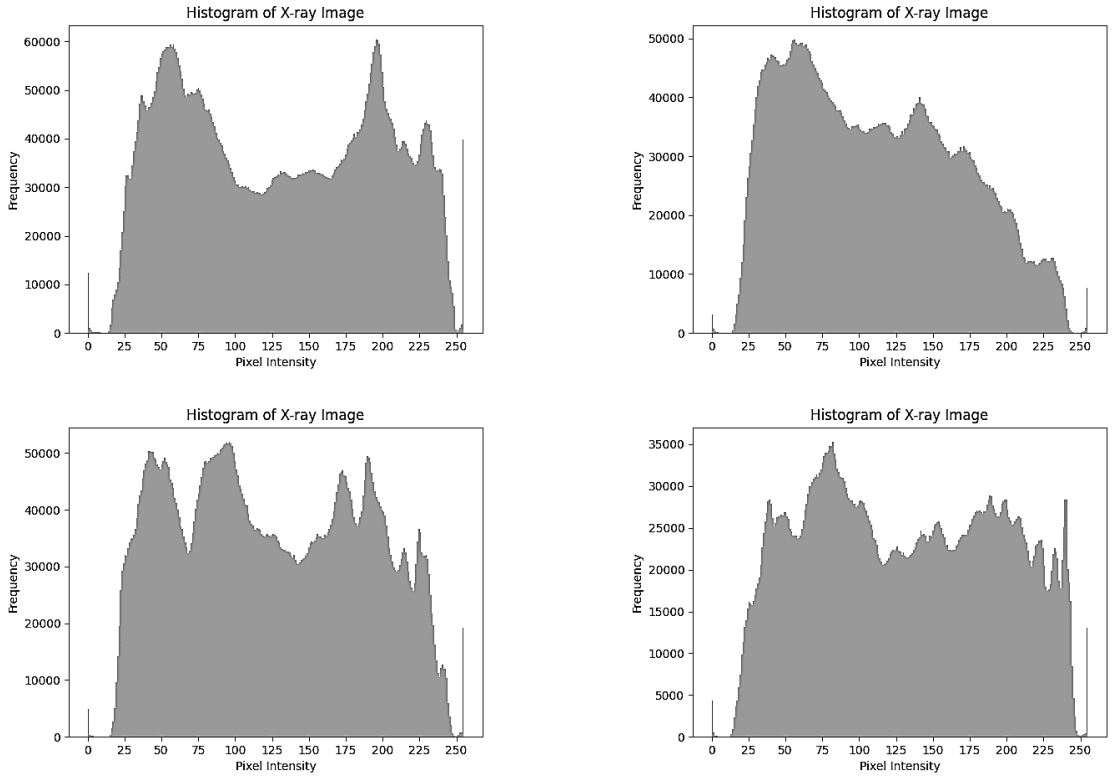


Fig. 6. Histogram of the Montgomery normal image dataset

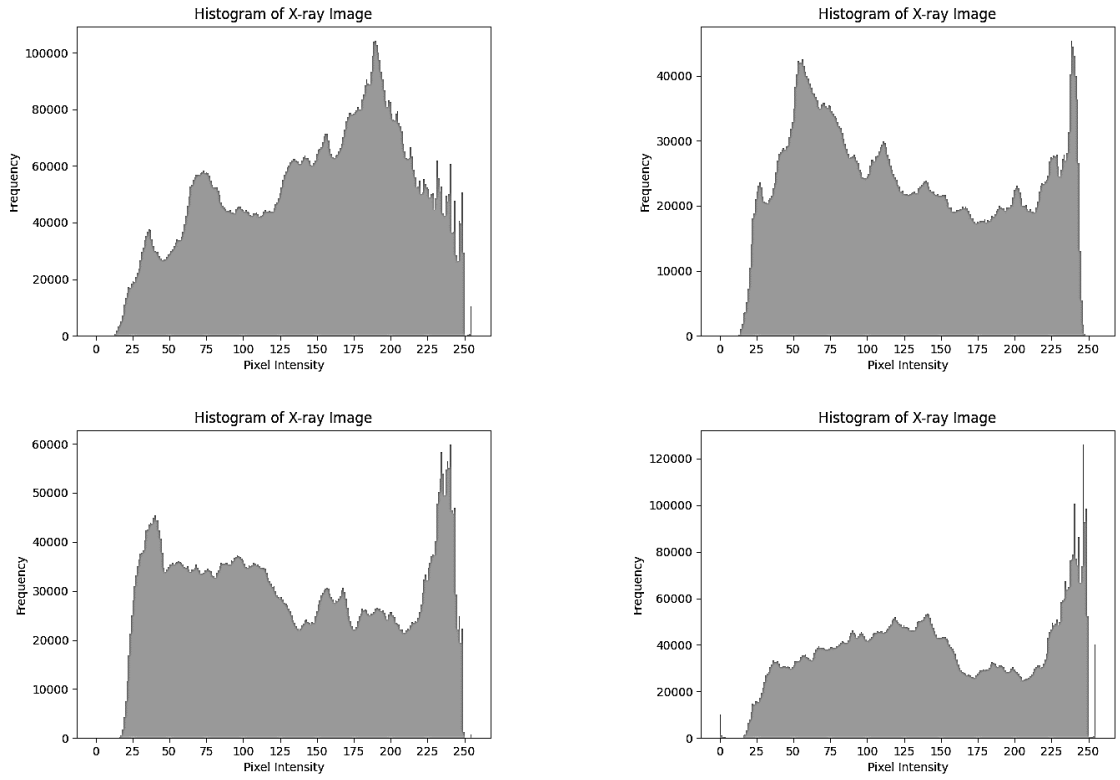
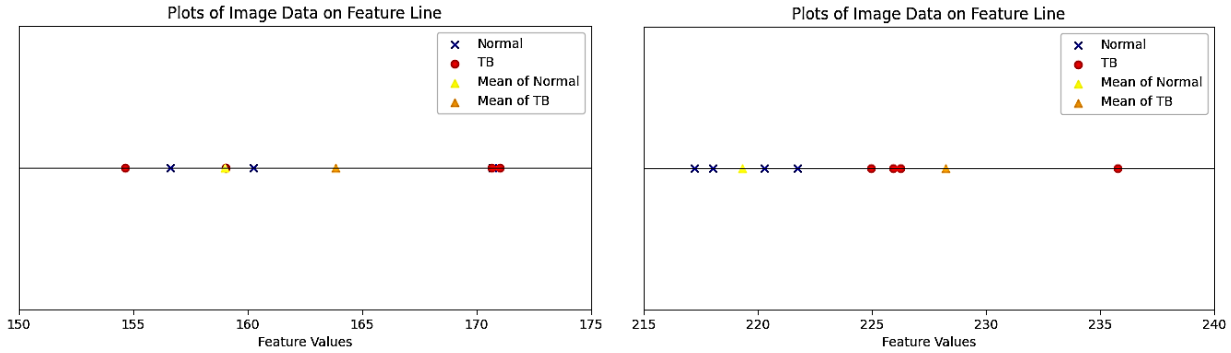


Fig. 7. Histogram of the Montgomery tuberculosis image dataset

ISSN 1814-4225 (print)
ISSN 2663-2012 (online)



(a) (b)

Fig. 8. Mean values of normal and tuberculosis groups of (a) Shenzhen and (b) Montgomery dataset

In addition, we also plotted the mean values of the CXR images from both datasets, Shenzhen and Montgomery, as shown in Fig. 9. The mean values of Shenzhen normal and tuberculosis CXR images were similar. However, the normal and tuberculosis CXR images from Montgomery exposed significantly different mean values.

There was no noticeable difference between the histograms of normal and tuberculosis conditions, indicating that the visual characteristics of tuberculosis in X-rays are not always explicitly reflected in the pixel intensity distribution. Thus, the deep learning approach is a good way to distinguish the complex characteristics of the CXR images.

In this study, we used three dataset schemes to determine the effect of each dataset on the learning approach. Scheme One uses only the Shenzhen dataset, Scheme Two incorporates the Montgomery, RSNA, and Belarus datasets, and Scheme Three uses all images from all datasets.

2.2. Image Preprocessing

As illustrated in Figs. 1, 2, and 3, each image is of various sizes and qualities. The images from Belarus datasets sized 2248×2248, Montgomery normal and Montgomery TB 4020×4892, RSNA 1024×1024, while Shenzhen normal and Shenzhen TB 3000×2939. Thus, we resized the dataset prior to feeding it into the network. All networks required the same size of images of 224×224 pixels. The distribution of the datasets was 60% for training, 20% for validation, and 20% for testing data. The number of datasets for the normal and tuberculosis classes are detailed in Table 1.

2.3. Residual Network (ResNet)

The convolutional neural network (CNN) is a neural network model that is used to find relationships and patterns between data items according to their relative positions. CNNs are mostly used to learn image data. Implementing CNNs has been widely used in many areas [25 - 27]. The function of the convolutional operation is to break down the spatial features of data that are initially complex into smaller subsets of data by trying to learn more features from the data that has been studied. The basic CNN architecture contains multiple convolution and pooling layers, with a fully-connected layer at the end, as shown in Fig. 9. Each of these stages has a different process; convolution is the first layer that takes the input image, pooling reduces the dimensions of the input image, and the fully connected layer performs the classification [28].

Table 1

Dataset Distribution

Database	Data type	Normal	Tuberculosis
Shenzhen	Training	196	196
	Validation	65	65
	Test	65	65
Montgomery, RSNA, and Belarus	Training	196	196
	Validation	65	65
	Test	65	65
Shenzhen, Montgomery, RSNA, and Belarus	Training	392	392
	Validation	130	130
	Test	130	130
Total		1304	1304

One good CNN architecture that demonstrated the best performance in the ImageNet Large-Scale Visual Recognition Challenge competition [29] is ResNet. ResNet is a deep neural network with residual learning [30, 31] that has a network structure designed to address the vanishing gradient problem in deep network training.

The architecture begins with an initial convolutional layer (Conv1) that uses a 7×7 kernel with a stride of 2, followed by a max pooling operation. Thereafter, there are several stages of residual blocks, the number

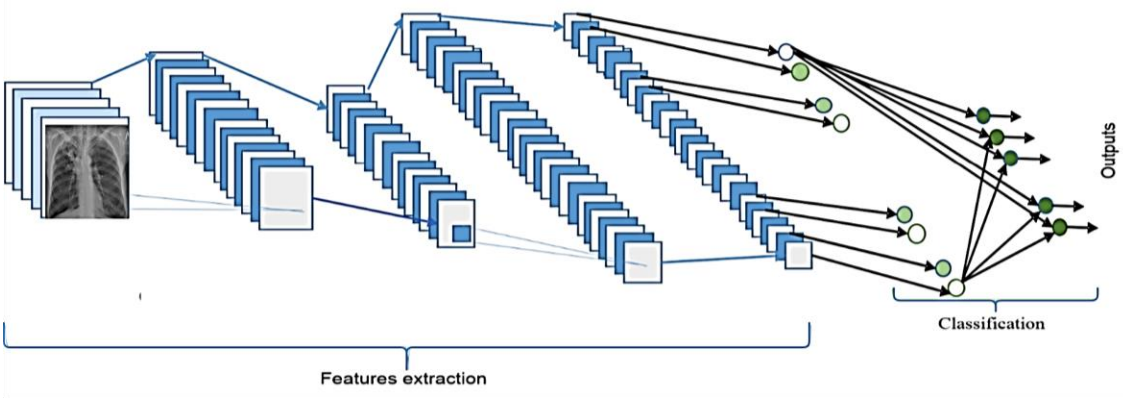


Fig. 9. The structure of the CNN algorithm [32]

of which depends on the depth of the model, such as ResNet-18, ResNet-34, or ResNet-50. Each stage contains residual blocks consisting of 2 or 3 convolutional layers equipped with batch normalization and ReLU activation functions. The key innovation of ResNet is the skip connection, which bypasses one or more layers in each residual block, allowing gradients to flow directly to the earlier layers. This structure includes a fully connected layer for classification, which is preceded by a global average pooling operation [30].

Table 2

ResNet architecture [30]

Layer name	Output size	50-layer	101-layer
Conv1	112×112	7×7, 64, stride 2	
Conv2_x	56×56	3×3 max pool, stride 2	
		$\begin{bmatrix}1\times1, & 64\\ 3\times3, & 64\\ 1\times1, & 256\end{bmatrix}\times3$	$\begin{bmatrix}1\times1, & 64\\ 3\times3, & 64\\ 1\times1, & 256\end{bmatrix}\times3$
Conv3_x	28×28	$\begin{bmatrix}1\times1, & 128\\ 3\times3, & 128\\ 1\times1, & 512\end{bmatrix}\times4$	$\begin{bmatrix}1\times1, & 128\\ 3\times3, & 128\\ 1\times1, & 512\end{bmatrix}\times4$
Conv4_x	14×14	$\begin{bmatrix}1\times1, & 256\\ 3\times3, & 256\\ 1\times1, & 1024\end{bmatrix}\times6$	$\begin{bmatrix}1\times1, & 256\\ 3\times3, & 256\\ 1\times1, & 1024\end{bmatrix}\times23$
Conv5_x	7×7	$\begin{bmatrix}1\times1, & 512\\ 3\times3, & 512\\ 1\times1, & 2048\end{bmatrix}\times3$	$\begin{bmatrix}1\times1, & 512\\ 3\times3, & 512\\ 1\times1, & 2048\end{bmatrix}\times3$
	1×1	Average pool, 1000-d fc, softmax	
FLOPs		3.8×10^9	7.6×10^9

Table 2 describes the architecture of ResNet-50 and ResNet-101. ResNet 50-layer uses 3-layer bottleneck blocks with the following structure: conv2.x (64 filters, 3 blocks), conv3.x (128 filters, 4 blocks), conv4.x (256 filters, 6 blocks), and conv5.x (512 filters, 3 blocks), resulting in a complexity of 3.8 billion FLOPs. In addition, the ResNet 101 layer has more bottleneck blocks to conv4.x, increasing it to 23 blocks, with a total complexity of 7.6 billion FLOPs, making it deeper yet still efficient. Both models demonstrated improved performance compared to shallower models [30].

The ResNet-50 architecture comprises 50 layers. This architecture combines residual blocks, which consist of multiple convolution layers, batch normalization, and activation functions, with shortcut connections [33]. ResNet-101 works with the same idea as ResNet-50 but has 101 layers; thus, it can go deeper than ResNet50 and extract more advanced features [34].

2.4. EfficientNet

The EfficientNet model is based on a multiple scaling method that expands the basic convolutional network model size to efficiently target the model size, thereby achieving the highest model accuracy gain. The combined scaling method allows networks to scale uniformly across width, depth, and resolution [35]. This architecture had fewer parameters, approximately 8.4 times less, and ran faster, approximately 6.1 times faster [36]. The EfficientNet model comprises a deeper network than the base model, which understands complex and richer features and generalizations. In addition, EfficientNet comprises a wider network that can extract optimal features and patterns useful for classification tasks. The basic component of EfficientNet is known as EfficientNetB0, and its latest variant is EfficientNetB7 [37].

EfficientNet is a neural network architecture utilizes compound scaling techniques to improve accuracy and computational efficiency. The network begins with an initial layer (stem) consisting of a 3×3 convolution with a stride of 2 and 32 filters. The main structure consists of a series of MBConv (Mobile Inverted Bottleneck Convolution) blocks that leverage separable convolutions and squeeze-and-excitation modules to enhance efficiency and channel feature modeling. Each stage has a specific block count, kernel size, expansion ratio, and filter number configuration. The Swish activation function

ISSN 1814-4225 (print)
ISSN 2663-2012 (online)

replaces ReLU to improve performance. At the end of the network, there is a head layer consisting of a 1×1 convolution, followed by global average pooling and a fully connected layer. EfficientNet is designed to balance depth, width, and input resolution optimally, achieving high performance with efficient resource utilization [38].

Table 3

EfficientNet-B0 baseline network [37]

Stage	Operator	Resolution	#Channels	#Layers
1	Conv3×3	224×224	32	1
2	MBConv1, k3×3	112×112	16	1
3	MBConv6, k3×3	112×112	24	2
4	MBConv6, k5× **5**	56×56	40	2
5	MBConv6, k3×3	28×28	80	3
6	MBConv1, k5× **5**	14×14	112	3
7	MBConv1, k5× **5**	14×14	192	4
8	MBConv1, k3×3	7×7	320	1
9	Conv1×1 & Pooling & FC	7×7	1280	1

Table 3 describes the architecture of EfficientNetB0. EfficientNet is designed using nine structured stages to balance efficiency and accuracy. In the initial stages, simple operations with high resolution and a low number of channels are performed. As the stages progress, the image resolution decreases, while the number of channels and the complexity of operations, such as the use of MBConv with 3x3 or 5x5 kernels, increase. In the final stage, a Conv1x1 operation with pooling and fully connected layers is performed to produce the final output. This approach allows the network to process features efficiently while maintaining strong representational capabilities [37].

2.5. Transfer Learning

Transfer learning is a machine learning technique in which a previously trained model learns common features on large-scale data sets, such as ImageNet, which is then transferred to the required task [3]. Transfer learning has several advantages, the most important of which is time efficiency, because under certain data conditions, it can take days or even weeks to train a neural network from scratch on complex tasks [39].

2.6. Hyperparameters

In this study, 10 deep learning architectures were applied to learn chest X-ray images. These architectures include ResNet-50, ResNet-101, EfficientNetB0, EfficientNetB1, EfficientNetB2, EfficientNetB3, EfficientNetB4, EfficientNetB5, EfficientNetB6, and EfficientNetB7.

In developing the model, all networks were initialized with pre-trained weights obtained from large ImageNet datasets. The final layers of the model are then adjusted by modifying the hyperparameters and recording the best values (Table 4). These hyperparameter values were applied to all networks trained to classify chest X-ray images into two classes: normal and tuberculosis.

Table 4

The learning parameters

Hyper-parameter	**Values**
Epoch	200
Batch Size	4
Learning Rate	0.001
Optimizer	SGD
Momentum	0.9
Input Size	224×224×3

2.7. Regularization

One indication of not good fit learning is low training loss but high validation loss. This indicates that the model is over-adjusting to the training data without capturing generalizable patterns. The strategies to address this issue include the use of data augmentation to provide more learning input, dropout rate to reduce dependency on certain features, L2 regularization to control model weights, and early termination to stop training when validation loss no longer improves. With the implementation of these regularizations [40], the training loss and validation loss graphs became more consistent without significant differences, and the model exhibited better generalizability.

3. Results and Discussion

The simulation results were recorded for each dataset usage scenario in each network learning. During training, we captured the best learning curves to analyze the network learning performance. Among the best learning curves, we saved and tested the model. The testing results are presented in a confusion matrix displaying the evaluation metrics values of accuracy, sensitivity, and specificity. We then compared the performance of the model in terms of its learning ability and accuracy rate in recognizing images.

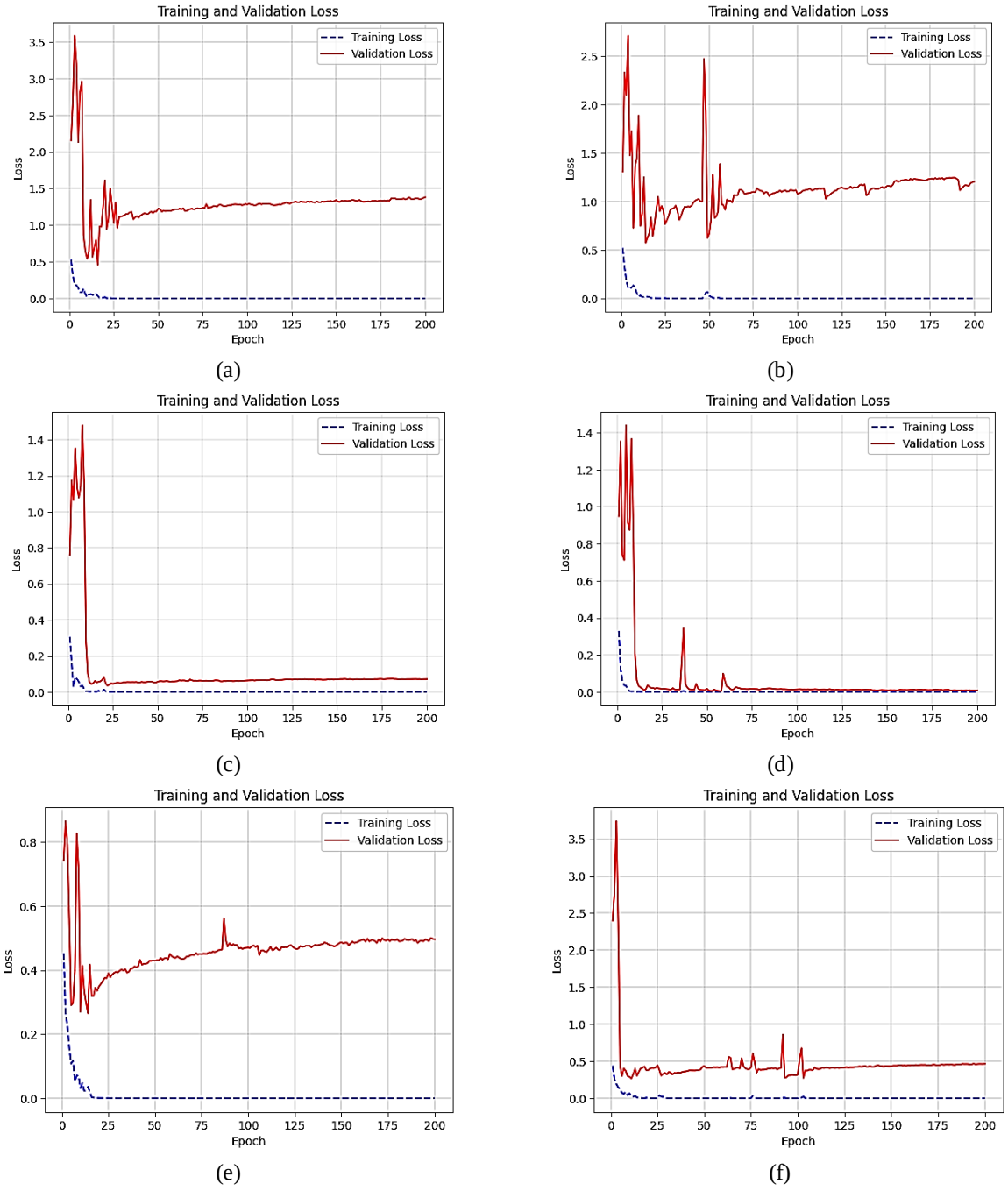


Fig. 10. Training result using (a) Scheme One on ResNet-50 and (b) ResNet-101; (c) Scheme Two on ResNet-50 and (d) ResNet-101; (e) Scheme Three on ResNet-50 and (f) ResNet-101

3.1. Scheme One

In In Scheme one, the Shenzhen dataset was used to train the ResNet-50, ResNet-101, and EfficientNet, along with the variants, by fine-tuning the hyperparameters. Fig. 10 (a) and (b) show the learning performances of ResNet-50 and ResNet-101 when trained on Shenzhen datasets.

The graphs show a significant difference between the training and validation loss curves, along with a continuous increase in the validation loss as the number of epochs increases, indicating the occurrence of overfitting. Additionally, the graphs also displayed chaotic fluctuations in the validation loss, indicating instability and inconsistency in the model's predictions. In this scheme, the models have difficulty learning the dataset because chest X-ray images from Shenzhen expose the presence of bones with high white color intensity in the lung area, which is considered tuberculosis, while it is actually normal. Similar trends are shown in Fig. 11 when EfficientNet variants were trained on the Shenzhen dataset.

ISSN 1814-4225 (print)
ISSN 2663-2012 (online)

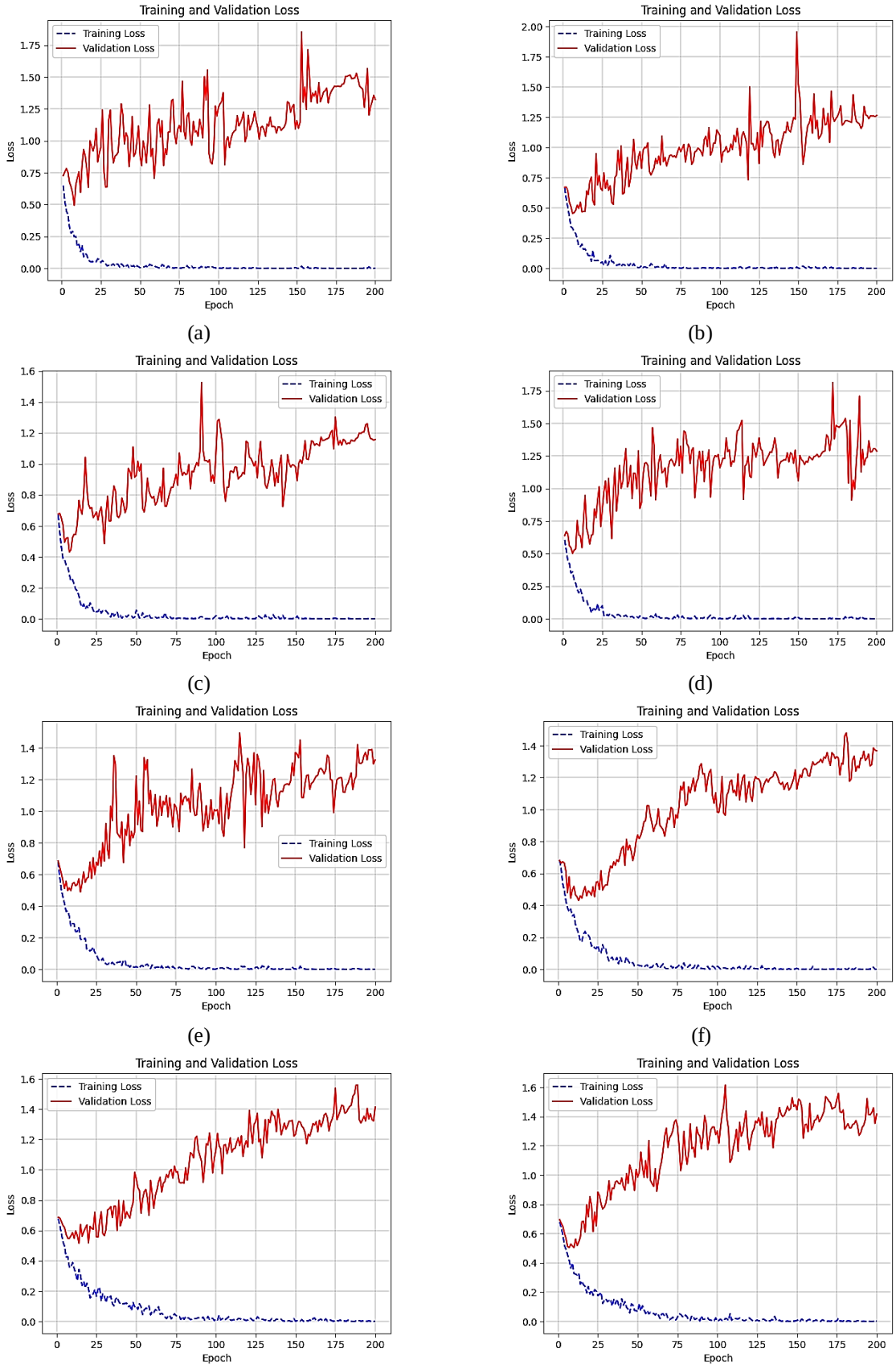


Fig. 11. Training result using Scheme One on (a) EfficientNetB0, (b) EfficientNetB1, (c) EfficientNetB2, (d) EfficientNetB3, (e) EfficientNetB4, (f) EfficientNetB5, (g) EfficientNetB6, and (h) EfficientNetB7

3.2. Scheme Two

In Scheme Two, the images were from datasets of the Montgomery, RSNA, and Belarus. Fig. 10 (c) and (d) and Fig. 12 displayed the training results using X-ray images from Scheme 2, which show better quality compared to Scheme One. In Fig. 10 (c), there are initial indications of overfitting as the validation loss curve increases over time. In Fig. 10 (d), the training loss and validation loss curves exhibit stability, with close and converging trends, indicating a good fit. The graphs shown in Fig. 12 exhibit significant fluctuations; however, the gap between training loss and validation loss remained low, which indicates that the models were a good fit.

When fluctuations occurred while maintaining a low gap, the models were considered capable of recognizing common patterns in the validation data and were not overly influenced by small variations in the data. Thus, ResNet-101 demonstrated the best learning on dataset Scheme 2.

3.3. Scheme Three

In this scheme, we combined the datasets from Schemes One and Two to increase the number of datasets and allow for more learning. X-ray images from the Shenzhen, Montgomery, RSNA, and Belarus datasets were used to train the ResNet and EfficientNet. The learning performances are shown in Fig. 10 (e) and (f) and Fig. 13.

In Fig. 10 (e) and (f), the upward trend of validation loss with increasing number of epochs indicates overfitting. This phenomenon, where the model becomes too specialized in the training data and loses its ability to adapt to new data, can have significant implications for the model's real-world performance. Overfitting is a common problem in complex models or when training data are limited.

In Fig. 13, the small fluctuations in the training loss curves indicate that the model was "memorizing" the training data. However, the large or erratic fluctuations in the validation loss curves indicated that the model could not generalize well to unseen data. The learning curves confirmed significant overfitting. The model performed very well on the training data but experienced a drastic decline in performance when faced with unseen data or validation datasets. Thus, the model could not recognize more general patterns and was too specific to the training data.

3.4. Applying Regularization

The overfitting problem observed in the initial training of all architectures (ResNet-50, ResNet-101, and EfficientNet B0 to B7) is characterized by very low training loss or increased validation loss. This indicates that the model is over-adjusting to the training data without capturing generalizable patterns. We applied augmentation to increase the model's ability to capture pattern diversity and enhance the generalizability of the new data. The data augmentation includes rotation, flip, and zoom.

The data augmentation was dynamic and applied only through code during the model training process. Augmentation techniques such as rotation, shifting, shearing, zooming, and horizontal flipping, are applied directly to the images when processed in batches by the data generator. Therefore, even though the augmentation generates different variations of the images during training, the number of images in the dataset folder remains the same as the original data. No augmented images were permanently saved in the dataset storage folder because the augmentation was used only to enrich the data virtually during model training.

Then, we used the dropout rate to reduce dependency on certain features, L2 regularization to control model weights, and early termination to stop training when validation loss no longer improved. Each regularization is individually applied to measure its effectiveness. However, the learning results demonstrate that the individual implementation of each technique cannot fully overcome the overfitting problem. This was indicated by the model performance, which still showed a large deviation between the accuracy rates of the training and validation data.

Next, we applied all regularizations in each network learning. For data augmentation, the transformation parameters used include rotation up to 10 degrees, width and height shifts up to 0.1 of the image dimensions, shear transformation of 0.1, zoom up to 0.1, and random horizontal flipping. These transformations help create diversity in the training data, whereas the validation and test data were only rescaled without additional transformations.

In the model architecture, L2 regularization was applied with a penalty parameter of 0.01 for each dense layer. Two dense layers were added, each with 64 and 32 neurons, respectively, and ReLU activation. After each dense layer, dropout was applied at a ratio of 0.5, randomly deactivating half of the neurons during training. This was combined with batch normalization to stabilize the output distribution between layers. The output layer employs a single neuron with sigmoid activation to support binary classification.

Early stopping was implemented by monitoring the loss value in the validation data. If the loss value did not improve over five consecutive epochs, training was terminated, and the best weights were automatically restored. The proposed method ensures that training stops at the right time, thereby preventing overfitting due to prolonged training.

ISSN 1814-4225 (print)
ISSN 2663-2012 (online)

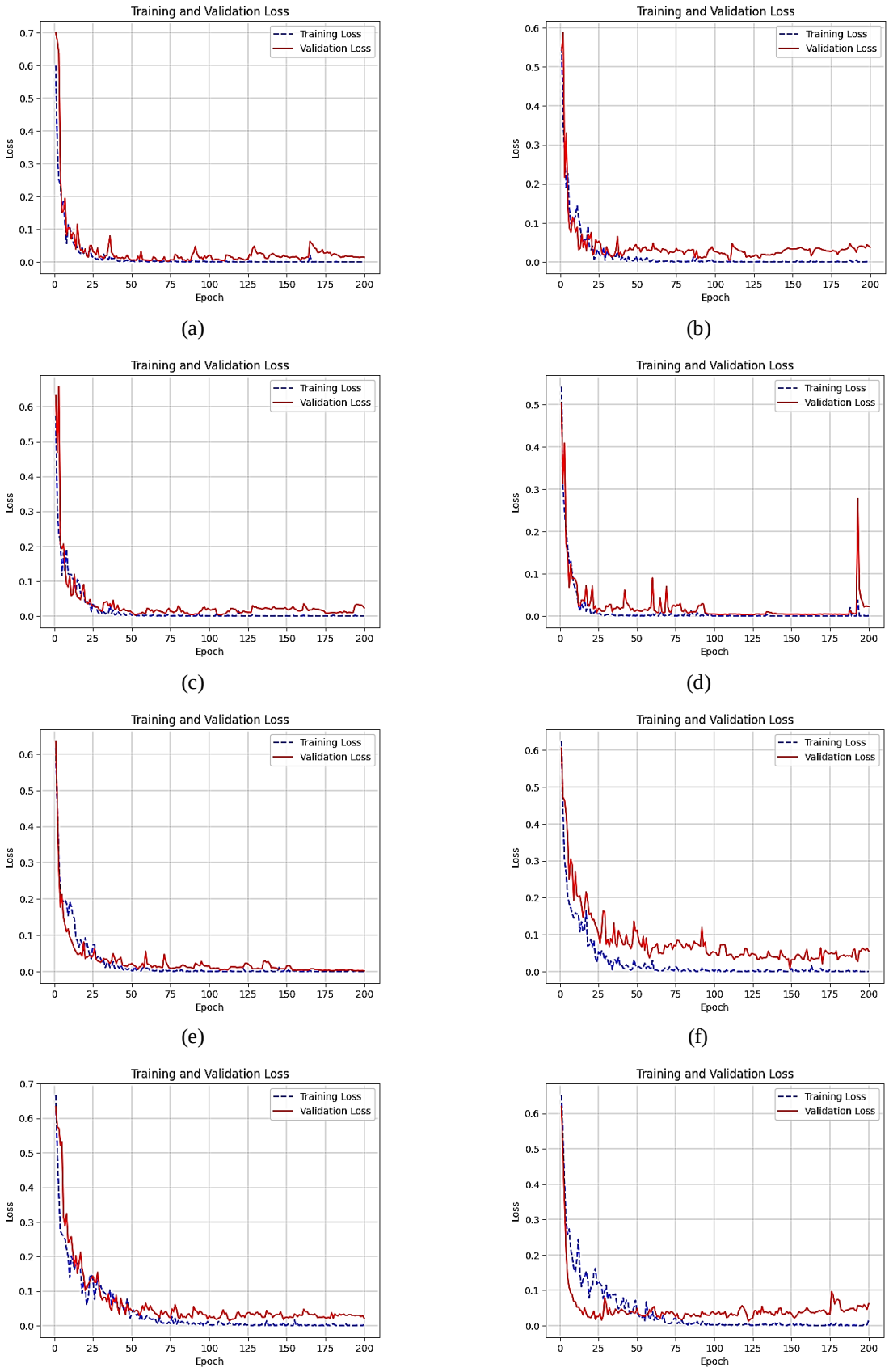


Fig. 12. Training result using Scheme Two on (a) EfficientNetB0, (b) EfficientNetB1, (c) EfficientNetB2, (d) EfficientNetB3, (e) EfficientNetB4, (f) EfficientNetB5, (g) EfficientNetB6, and (h) EfficientNetB7

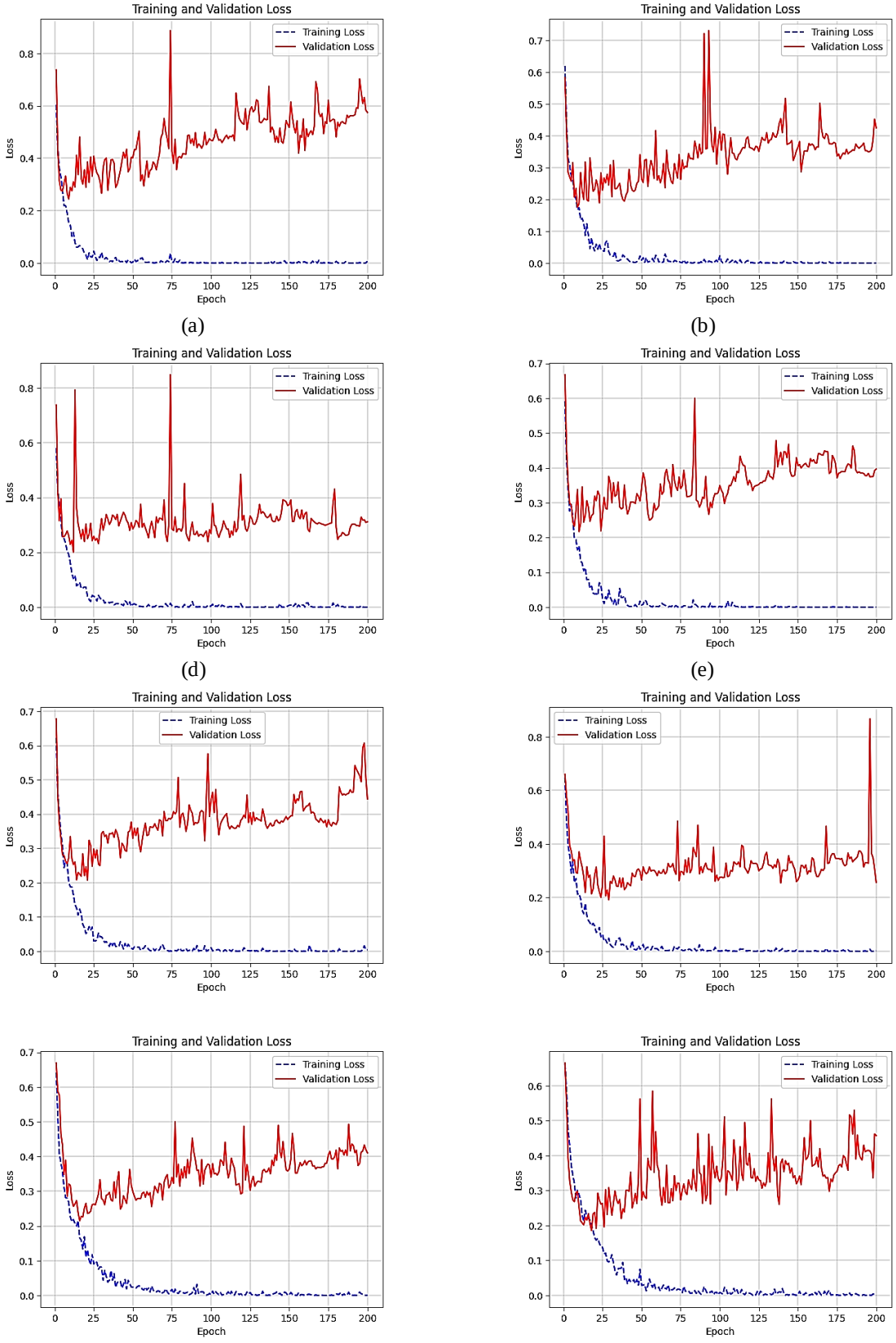


Fig. 13. Training result using Scheme Three on (a) EfficientNetB0, (b) EfficientNetB1, (c) EfficientNetB2, (d) EfficientNetB3, (e) EfficientNetB4, (f) EfficientNetB5, and (g) EfficientNetB6, and (h) EfficientNetB7

ISSN 1814-4225 (print)
ISSN 2663-2012 (online)

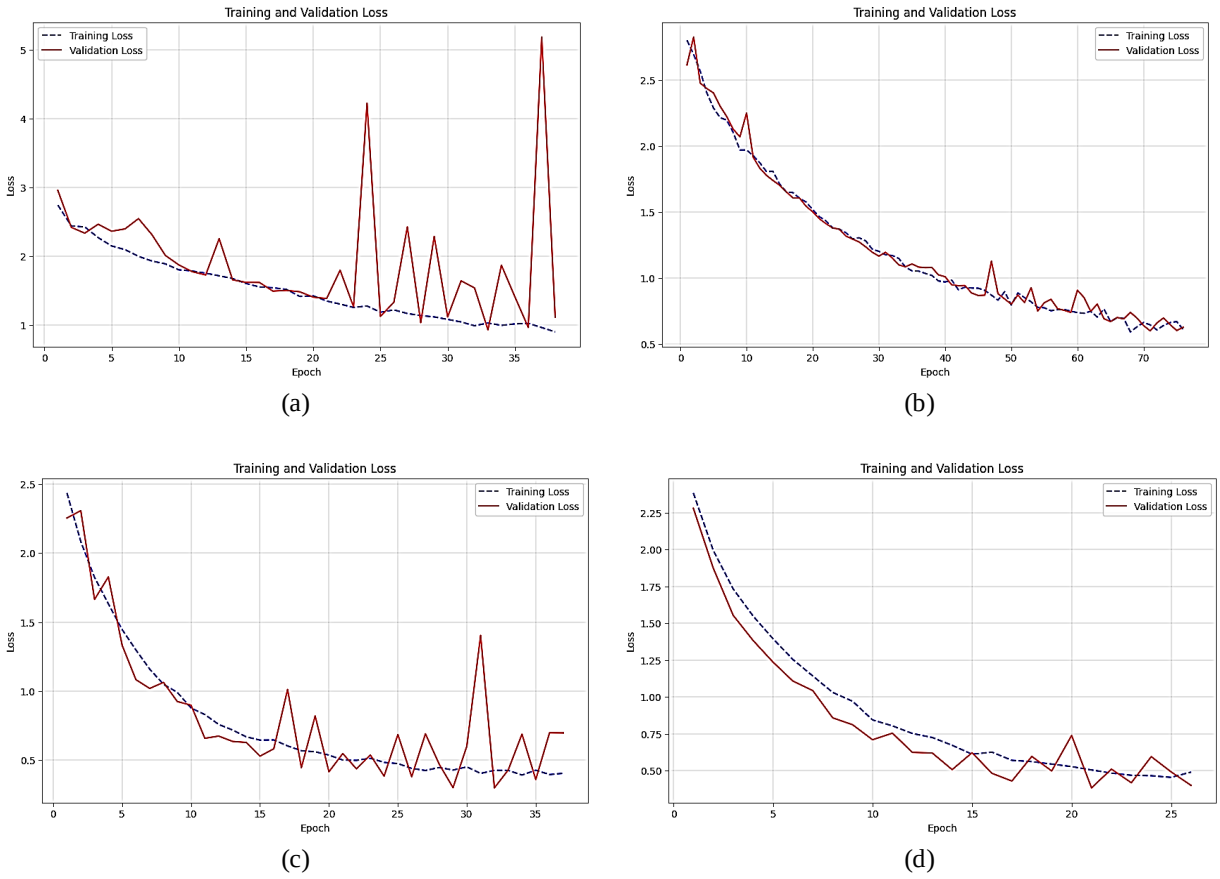


Fig. 14. Training result using regularizations from Scheme One on (a) ResNet-50 and (b) ResNet-101; Scheme Three on (c) ResNet-50 and (d) ResNet-101

As a result, when all the regularizations were combined, the model performance showed a significant improvement. The results became more stable with an alignment between the accuracy of the training and validation data, as shown in Fig. 14, Fig. 15, and Fig. 16. Figure 14 shows that ResNet50 learning was difficult to achieve stability in either Scheme One or Scheme Three, but ResNet101 learning tended to improve along the training process. The learning curves demonstrate that applying regularizations effectively improves model stability for deep-layer networks.

In addition, applying regularizations to EfficientNet was effective for nearly all EfficientNet variants. As shown in Fig. 15 and Fig. 16, training the network with regularizations on Scheme One improved the learning performance of EfficientNetB1, EfficientNetB3, and EfficientNetB7, and on Scheme Three, EfficientNetB0, EfficientNetB1, EfficientNetB2, and EfficientNetB6. However, more work is required to obtain a good fit model using various hyperparameter settings and regularizations.

3.5. Classification Performance

The models with the best learning performance were then tested using unseen test data to ensure their ability to perform well on new data. Scheme Two resulted in the best learning, and from Fig. 10 and Fig. 12, we find that ResNet-101, EfficientNetB0, EfficientNetB1, EfficientNetB2, EfficientNetB4, and EfficientNetB6 show a good learning curve or good fit results; thus, it should increase confidence in the evaluation of the model's performance. The model was evaluated using a confusion matrix that describes the performance of the model when classifying the dataset by providing detailed information about the accuracy, sensitivity, and specificity rates.

Fig. 17 displayed the confusion matrix of the testing results using the ResNet-101, EfficientNetB0, EfficientNetB1, EfficientNetB2, EfficientNetB4, and EfficientNetB6 architectures in Scheme 2. Based on the confusion matrix, the accuracy, sensitivity, specificity, and values were calculated to assess the performance of each architecture (Table 5). The results confirmed that the ResNet-101 and EfficientNetB0 architectures outperformed the other architectures in testing images.

Based on the testing results, using a good dataset (Scheme 2) resulted in a model capable of recognizing negative images effectively. This is indicated by the false positive value of 0 for each model. Table 5 confirms this finding. According to Table 5, the specificity value was

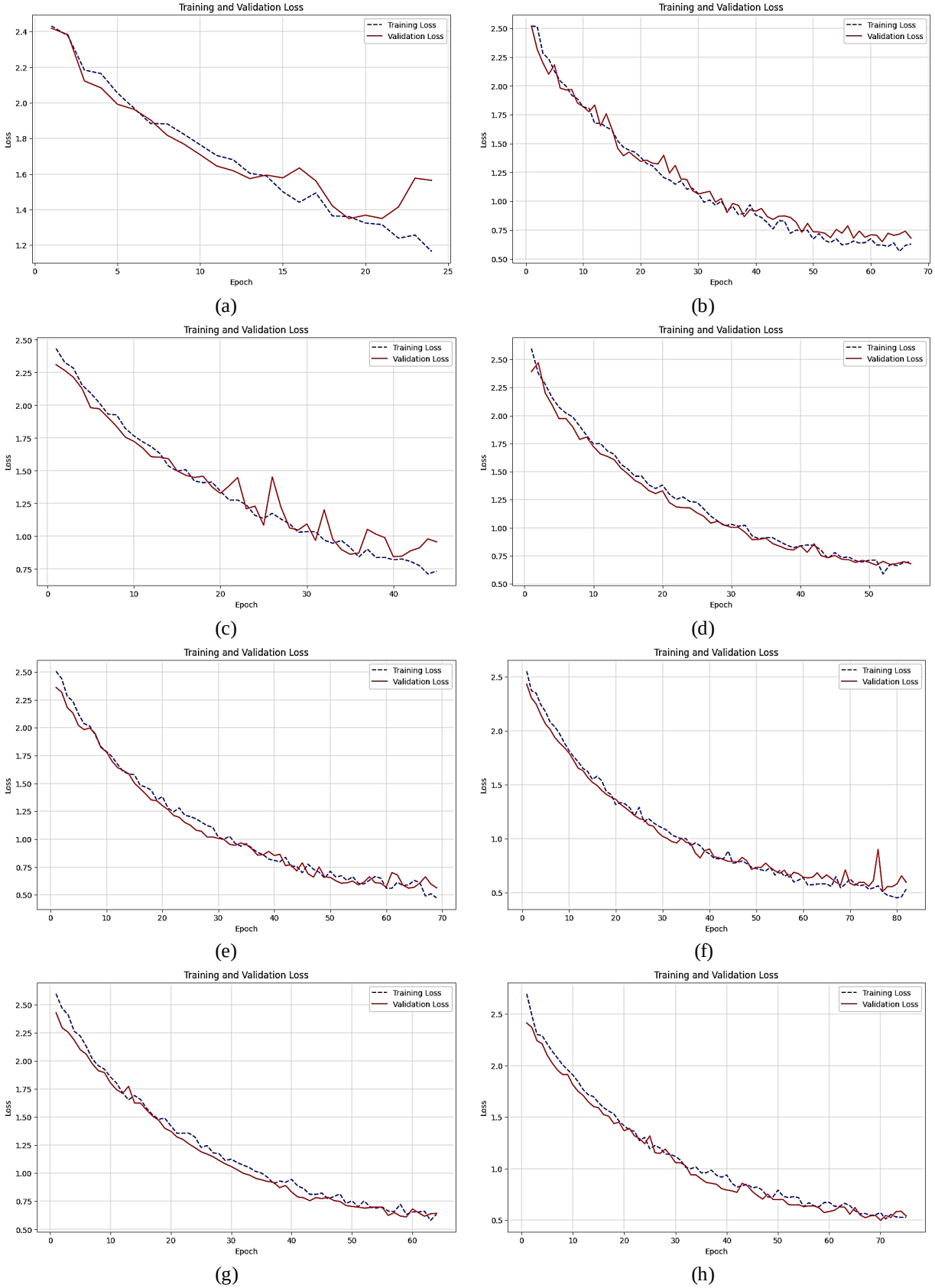


Fig. 15. Training result using regularizations from Scheme One on (a) EfficientNetB0, (b) EfficientNetB1, (c) EfficientNetB2, (d) EfficientNetB3, (e) EfficientNetB4, (f) EfficientNetB5, and (g) EfficientNetB6, and (h) EfficientNetB7

ISSN 1814-4225 (print)
ISSN 2663-2012 (online)

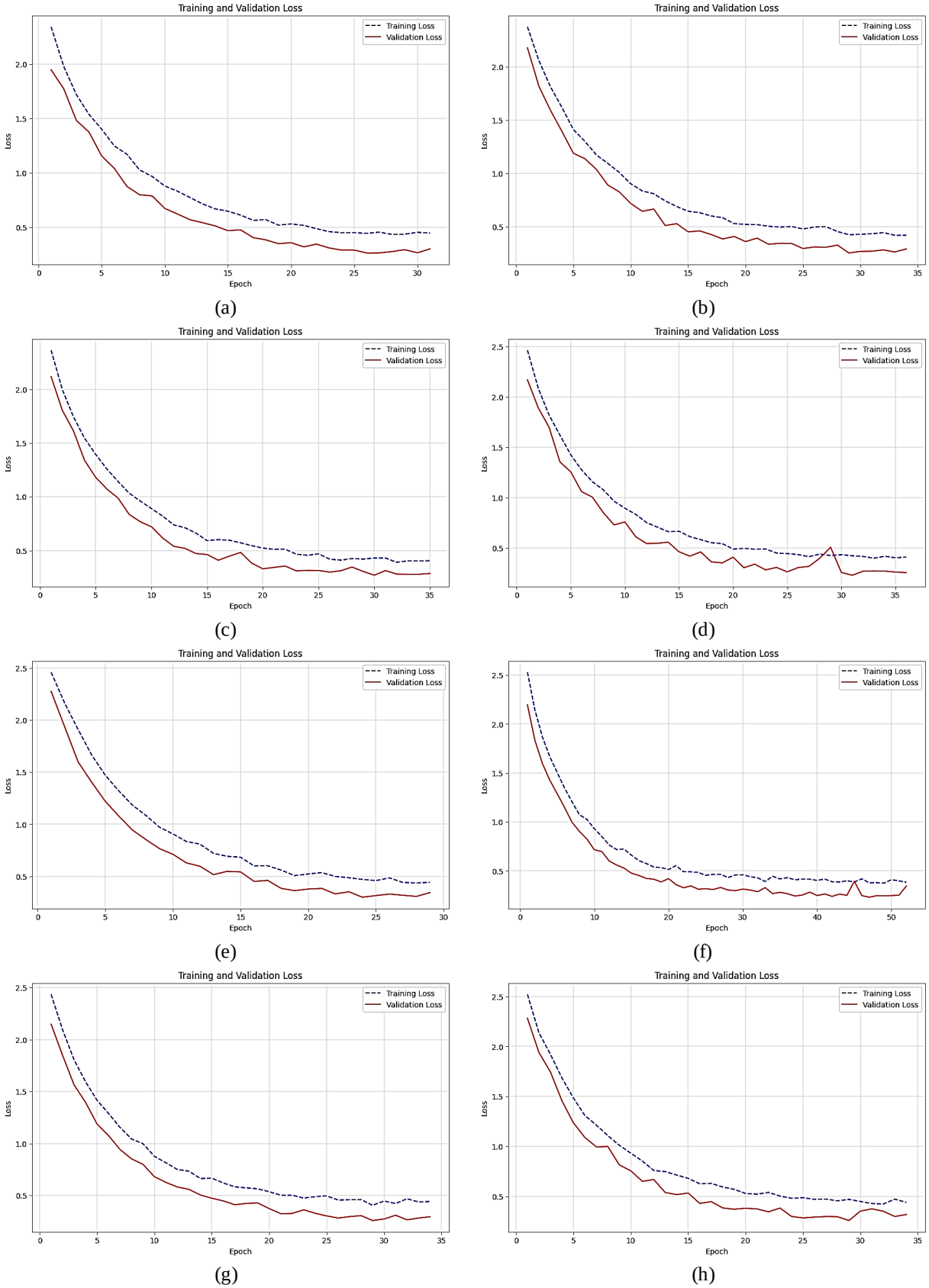


Fig. 16. Training result using regularizations from Scheme Three on (a) EfficientNetB0, (b) EfficientNetB1, (c) EfficientNetB2, (d) EfficientNetB3, (e) EfficientNetB4, (f) EfficientNetB5, and (g) EfficientNetB6, and (h) EfficientNetB7

100%. Furthermore, from the confusion matrix that resulted in a non-zero false negative value, we can conclude that there are positive images that are mistakenly identified as negative. This possibility arises because positive images in the testing dataset are considered negative.

These results also affirm the importance of dataset selection in model development. A non-representative dataset can result in poorly performing models with limited generalizability. Scheme Two, which combines multiple good and representative datasets, also yields a good model. However, when the dataset in Scheme Two is merged with a poor or non-representative dataset (Scheme Three), the model's performance deteriorates or decreases.

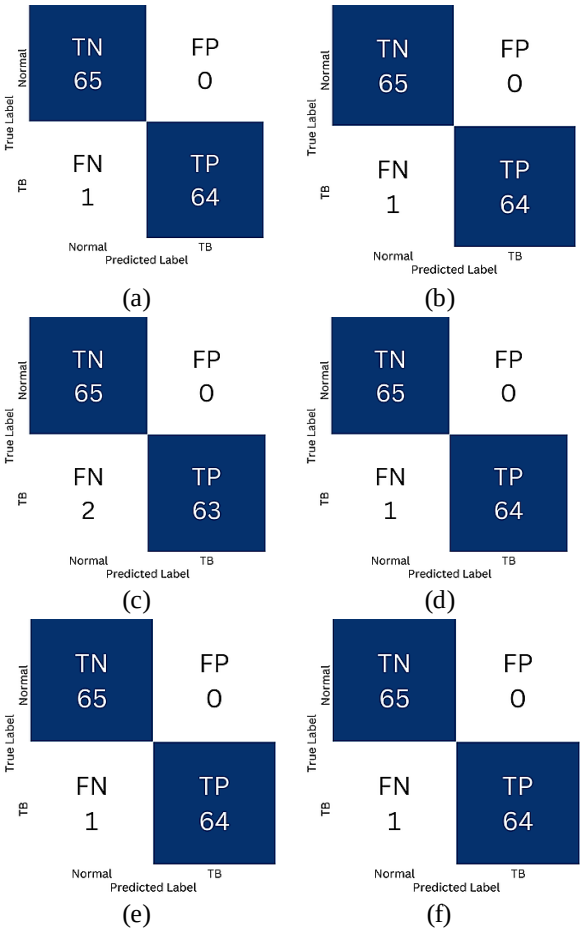


Fig. 17. Confusion matrix of the testing results from Scheme Two without regularizations: (a) ResNet-101, (b) EfficientNetB0, (c) EfficientNetB1, (d) EfficientNetB2, (e) EfficientNetB4, and (f) EfficientNetB6

Furthermore, we also tested the models resulting from Schemes One and Three, which were improved using regularizations. From Fig. 14, Fig. 15, and Fig. 16, applying regularizations to ResNet 101, EfficientNetB1, EfficientNetB3, and EfficientNetB7 effectively increased the learning stability of Scheme One. In addition, applying regularizations to Scheme Three effectively increased the learning performance on networks EfficientNetB0, EfficientNetB1, EfficientNetB2, and EfficientNetB6.

Table 5

Evaluation matrix of the testing result for Scheme Two

Architecture	Evaluation Matrix		
	Accuracy	Sensitivity	Specificity
ResNet-101	**99.2%**	**98.5%**	**100%**
EfficientNetB0	**99.2%**	**98.5%**	**100%**
EfficientNetB1	98.5%	96.9%	100%
EfficientNetB2	99.2%	98.4%	100%
EfficientNetB4	99.2%	98.4%	100%
EfficientNetB6	99.2%	98.4%	100%

Tables 6 and 7 present the evaluation results of the models using the accuracy, sensitivity, and specificity metrics after regularization in the two schemes, respectively. Table 6 presents the results from Scheme One, and Table 7 presents the testing results of the best model in Scheme Three.

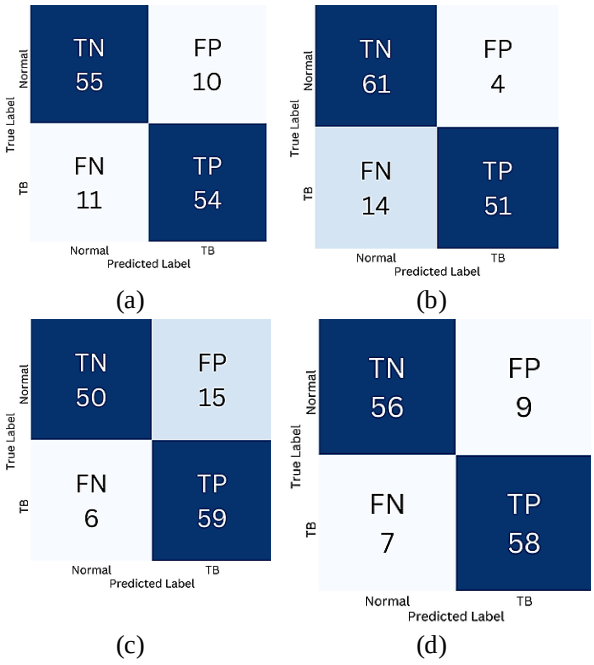


Fig. 18. Confusion matrix of the testing results from Scheme One with regularizations: (a) ResNet-101, (b) EfficientNetB1, (c) EfficientNetB3, (d) EfficientNetB7

Under Scheme One, the best model was obtained using EfficientNetB1, EfficientNetB3, EfficientNetB7, and ResNet-101. Based on the evaluation results presented in Table 6, EfficientNetB7 demonstrated the best overall performance with an accuracy of 87.6%,

ISSN 1814-4225 (print)
ISSN 2663-2012 (online)

sensitivity of 89.2%, and specificity of 86.1%. In addition, EfficientNetB1 achieved the highest specificity (93.8%) despite having a relatively low sensitivity (78.4%). The ResNet-101 and EfficientNetB3 architectures demonstrated balanced outcomes across several metrics, although with variations in accuracy, sensitivity, and specificity.

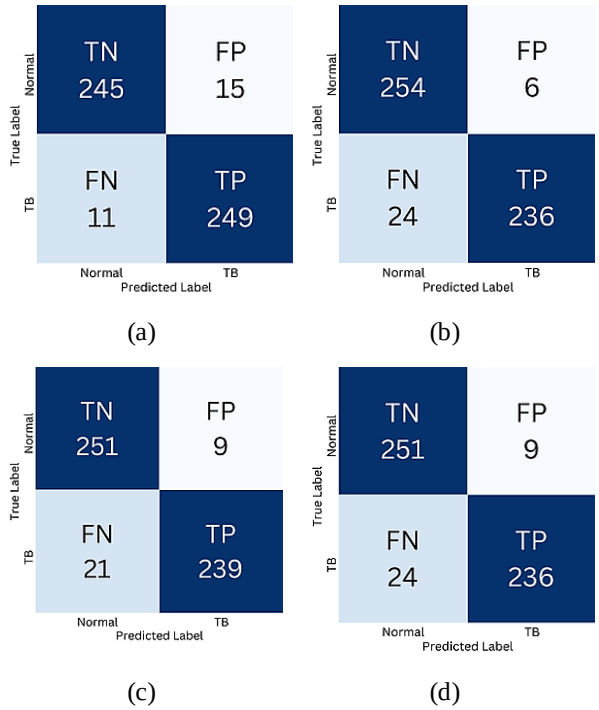


Fig. 19. Confusion matrix of the testing results from Scheme Three with regularizations: (a)EfficientNetB0, (b) EfficientNetB1, (d) EfficientNetB2, and (c) EfficientNetB6

Table 6

Evaluation matrix of testing results from Scheme One after using regularizations

Architecture	Evaluation Matrix		
	Accuracy	Sensitivity	Specificity
EfficientNetB1	86.1%	78.4%	93.8%
EfficientNetB3	83.8%	90.7%	76.9%
EfficientNetB7	87.6%	89.2%	86.1%
ResNet-101	83.8%	83%	84.6%

In addition, Table 7 presents the evaluation results obtained in Scheme Three, which demonstrate a significant performance improvement compared to Scheme 1. The tested models were EfficientNetB0, EfficientNetB1, EfficientNetB2, and EfficientNetB6. EfficientNetB0 demonstrated the highest performance with an accuracy of 95%, sensitivity of 95.7%, and specificity of 94.2%. The EfficientNetB1 and EfficientNetB2 models also demonstrated excellent results, with accuracy exceeding 94% and high specificity. Although EfficientNetB6 demonstrated the lowest accuracy (93.6%), its sensitivity and specificity remained high at 90.7% and 96.5%, respectively.

Table 7

Evaluation matrix of testing results from Scheme Three after using regularizations

Architecture	Evaluation Matrix		
	Accuracy	Sensitivity	Specificity
EfficientNetB0	95%	95.7%	94.2%
EfficientNetB1	94.2%	90.7%	97.6%
EfficientNetB2	94.2%	91.9%	96.5%
EfficientNetB6	93.6%	90.7%	96.5%

4. Future Works

Applying regularizations and fine-tuning the hyperparameter settings effectively improved learning performance. However, the testing results showed that the model trained on high-quality input achieved the highest accuracy. Thus, work on enhancing images, such as contrast image preprocessing and region of interest segmentation, is required to construct a robust CXR classification model.

Poor-quality images or data biases can interfere with the learning process and degrade the model's generalizability. Therefore, it is important to prepare data with optimal preprocessing, proper dataset selection, and training strategies that consider data variations to avoid overfitting and improve accuracy in real clinical applications.

In addition, other CNN networks with an optimal learning algorithm should be used to achieve high model performance. A self-supervised or reinforcement learning approach may be a good choice for developing robust models.

5. Conclusions

Based on the research findings, it was observed that the dataset in Scheme Two, utilizing the ResNet-101 and EfficientNetB0 architectures, performed well in learning chest X-ray images, enabling effective classification of TB and normal classes. Both models achieved an accuracy of 99.2%, sensitivity of 98.5%, and specificity of 100%. Furthermore, the use of the Shenzhen dataset in Schemes One and Three resulted in overfitting the learning curves, while employing datasets other than Shenzhen

in Scheme Two improved the model's performance. Applying regularization effectively improved the learning performance but did not significantly increase the testing accuracy rate. This demonstrates the significance of using high-quality images to enhance the model's performance. However, it should be noted that simply combining images from different sources to enlarge the dataset did not guarantee improved learning outcomes. Therefore, the best approach to enhance learning performance is to improve the image quality. However, using another deep learning network that has better potential in recognizing medical images with deep hyperparameter settings and regularizations should be performed to provide a good fit and high-performance model.

Contributions of authors: conceptualization – **Khairul Munadi**, methodology – **Muhammad Irhamsyah;** formulation of tasks – **Muhammad Irhamsyah,** contribution analysis – **Khairun Saddami, Khairul Munadi;** model development – **Muhammad Irhamsyah, Qurrata A'yuni;** model verification – **Khairun Saddami;** analysis of results – **Muhammad Irhamsyah, Khairul Munadi;** visualization – **Qurrata A'yuni;** writing original draft preparation – **Muhammad Irhamsyah;** review and editing - **Fitri Arnia, Nasaruddin Nasaruddin.**

Conflict of interest

The authors declare that they have no conflict of interest in relation to this research, whether financial, personal, authorship, or otherwise, that could affect the research and its results presented in this paper.

Funding statement

This work was supported by LPPM Universitas Syiah Kuala under scheme of Penelitian Lektor with Grant No. 343/UN11.2.1/PG.01.03/SPK/PTNBH/2024.

Data availability

Manuscript has no associated data.

Use of artificial intelligence

The authors confirm that they did not use artificial intelligence methods while creating the presented work.

All authors have read and agreed with the published version of this manuscript.

References

1. Miggiano, R., Rizzi, M., & Ferraris, D.M. Mycobacterium tuberculosis Pathogenesis, Infection Prevention and Treatment. *Pathogens*, 2020, vol. 9, iss. 5, article no. 385. DOI: 10.3390/pathogens9050385.

2. Ayaz, M., Shaukat, F., & Raja, G. Ensemble learning based automatic detection of tuberculosis in chest X-ray images using hybrid feature descriptors. *Phys. Eng. Sci. Med.*, 2021, vol. 44, iss 1, pp. 183–194. DOI: 10.1109/ACCESS.2020.3031384.

3. Varshni, D., Kartik, T., Lucky, A., Rahul, N., & Ankush, M. Pneumonia Detection Using CNN based Feature Extraction. *IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT)*. Coimbatore, India: IEEE, 2019, pp. 1–7. DOI: 10.1109/ICECCT.2019.8869364.

4. Toraman, S., Alakus, T. B., & Turkoglu, I. Convolutional capsnet: A novel artificial neural network approach to detect COVID-19 disease from X-ray images using capsule networks. *Chaos Solitons Fractals*, 2020, vol. 140, article no. 110122. DOI: 10.1016/j.chaos.2020.110122.

5. Yusoff, M., Saaidi, M. S. I., Amirul Sadikin Md. Afendi, S Md., & Hassan, A. Tuberculosis X-Ray Images Classification based Dynamic Update Particle Swarm Optimization with CNN. *Journal of Hunan University Natural Sciences*, 2021, vol. 48, iss. 9. ISSN 1674-2974.

6. Rahman,T., Khandakar, A., Abdul Kadir, M., Islam, K. R., Islam, K. F., Mazhar, R., Hamid, T., Islam, M. T., Kashem, S., Ayan, M. A., & Chowdhury, M. E. H. Reliable Tuberculosis Detection Using Chest X-Ray With Deep Learning, Segmentation and Visualization. IEEE Access, 2020, vol. 8, pp. 191586–191601. DOI: 10.1109/ACCESS.2020.3031384.

7. Munadi, K., Muchtar, K., Maulina, N., & Pradhan. B. Image Enhancement for Tuberculosis Detection Using Deep Learning. *IEEE Access*, 2020, vol. 8, pp. 217897–217907. DOI: 10.1109/ACCESS.2020.3041867.

8. Harahap, M., Pasaribu, A. P. S., Sinaga, D. R., Sipangkar, R., & Samuel, S. Classification of Tuberculosis Based on Lung X-Ray Image with Data Science Approach Using Convolutional Neural Network. *Sinkron: Jurnal Dan Penelitian Teknik Informatika*, 2022, vol. 6, iss. 4, pp. 2193–2197. DOI: 10.33395/sinkron.v7i4.11711.

9. Nafisah, S. I., & Muhammad, G. Tuberculosis detection in chest radiograph using convolutional neural network architecture and explainable artificial intelligence. *Neural Comput. Appl.*, 2022, vol. 36, pp. 113-13. DOI: 10.1007/s00521-022-07258-6.

10. Chowdhury, N. K., Kabir, M. A., Rahman, M., & Rezoana, N. ECOVNet: An Ensemble of Deep Convolutional Neural Networks Based on EfficientNet to Detect COVID-19 From Chest X-rays. *PeerJ Comput. Sci.*, 2021, vol. 7, article no. e551. DOI: 10.48550/arXiv.2009.11850.

ISSN 1814-4225 (print)
ISSN 2663-2012 (online)

11. Oloko-Oba, M., & Viriri, S. *Diagnosing Tuberculosis Using Deep Convolutional Neural Network. Image and Signal Processing*, ed. El Moataz A. et al. Cham: Springer International Publishing, 2020, vol. 12119, pp. 151–161. DOI: 10.1007/978-3-030-51935-3 16.

12. Oloko-Oba, M., & Viriri, S. *Ensemble of EfficientNets for the Diagnosis of Tuberculosis. Comput. Intell. Neurosci*, ed. Lo Bosco G, 2021, vol. 2021, pp. 1–12. DOI: 10.1155/2021/9790894.

13. Devasia, J., Goswami, H., Lakshminarayanan, S., Rajaram, M., & Adithan, S. Deep learning classification of active tuberculosis lung zones wise manifestations using chest X-rays: a multi label approach. *Sci. Rep.* 2023, vol. 13, iss 1, article no. 887. DOI: 10.1038/s41598-023-28079-0.

14. Elizar, E., Zulkifley, M. A., & Muharar, R. Scaling and Cutout Data Augmentation for Cardiac Segmentation. *Proceedings of International Conference on Data Science and Applications*, ed. Saraswat M. et al. Singapore: Springer Nature Singapore, 2023, vol. 552, pp. 599–609. DOI: 10.1007/978-981-19-6634-7_42.

15. Li, Y., Han, M., Li, K., Han, Y., & Chen, P. X-Ray Image Enhancement Framework Based on Improved Local Adaptive Contrast Field for Complex Workpieces. *IEEE Trans. Nucl. Sci.*, 2024. vol. 71, iss. 5, pp. 1225–1232. DOI: 10.1109/TNS.2024.3389106.

16. Stirenko, S., Kochura, Y., Alienin, O., Rokovyi, O., Gang, P., Zeng, W., & Gordienko, Y. Chest X-Ray Analysis of Tuberculosis by Deep Learning with Segmentation and Augmentation. *IEEE 38th International Conference on Electronics and Nanotechnology (ELNANO)*. Kiev, IEEE, 2018, pp. 422–428. DOI: 10.48550/arXiv.1803.01199.

17. Pasa, F., Golkov, V., Pfeiffer, F., Cremers, D., & Pfeiffer, D. Efficient deep network architectures for fast chest X-ray tuberculosis screening and visualization. *Scientific reports*, 2019, vol. 9, iss. 1, article no. 6268. DOI: 10.1038/s41598-019-42557-4.

18. Inbaraj, Xavier A., Villavicencio, C., Macrohon, J. J., Jeng, J. H., & Hsieh, J. G. A novel machine learning approach for tuberculosis segmentation and prediction using chest-x-ray (CXR) images. *Applied Sciences*, 2021, vol. 11, iss. 19, article no. 9057. DOI: 10.3390/app11199057.

19. Natarajan, S., Sampath, P., Arunachalam, R., Shanmuganathan, V., Dhiman, G., Chakrabati, T., & Margala, M. Early diagnosis and meta-agnostic model visualization of tuberculosis based on radiography images. *Scientific Reports*, 2023, vol. 13, iss. 1, article no. 22803. DOI: 10.1038/s41598-023-49195-x.

20. Jaeger, S., Candemir, S., Antani, S., Wáng, Y.-X ., Lu, P.-X., & Thoma, G. Two public chest X-ray datasets for computer-aided screening of pulmonary diseases. *Quant. Imag. Med. Surgery*, 2014, vol. 4, no. 6, article no. 475. DOI: 10.3978/j.issn.2223-4292.2014.11.20.

21. *Tuberculosis Chest X-rays (Shenzhen)*. Available at: https://www.kaggle.com/datasets/raddar/tuberculosis-chest-xrays-shenzhen. (accessed 10.10.2024).

22. *Tuberculosis Chest X-rays (Montgomery)*. Available at: https://www.kaggle.com/datasets/raddar/tuberculosis-chest-xrays-montgomery. (accessed 10.10.2024).

23. *RSNA Pneumonia Detection Challenge*. Available at: https://kaggle.com/competitions/rsna-pneumonia-detection-challenge. (accessed 10.10.2024).

24. *Drug resistant tuberculosis X-rays*. Available at: https://www.kaggle.com/datasets/raddar/drug-resistant-tuberculosis-xrays. (accessed 10.10.2024).

25. Maeda-Gutiérrez, V., & et al. Comparison of Convolutional Neural Network Architectures for Classification of Tomato Plant Diseases. *Appl. Sci.*, 2020, vol. 10, iss. 4, article no. 1245. DOI: 10.3390/app10041245.

26. Sengupta, S., Basak, S., Saikia, P., Paul, S., Tsalavoutis, V., Ataiah, F., Ravi, F., & Peters, A. A review of deep learning with special emphasis on architectures, applications and recent trends. *Knowl.-Based Syst.*, 2020, vol. 194, article no. 105596. DOI: 10.1016/j.knoys.2020.105596.

27. De Matos, J., Ataky, S., Britto, A., Oliveira, L., & Koerich, A. Machine Learning Methods for Histopathological Image Analysis: A Review. *Electronics*, 2021, vol. 10, iss. 5, article no. 562. DOI: 10.3390/electronics10050562.

28. Simonyan, K., & Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv:1409.1556*. arXiv, 2015, no. arXiv:1409.1556. DOI: 10.48550/arXiv.1409.1556.

29. He, K., Zhang, X., Ren, S., & Sun, J. Deep Residual Learning for Image Recognition. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Las Vegas, NV, USA, IEEE, 2016, pp. 770–778. DOI: 10.1109/CVPR.2016.90.

30. Targ, S., Almeida, D., & Lyman, K. Resnet in Resnet: Generalizing Residual Architectures. *arXiv:1603.08029*. arXiv, 2016. DOI: 10.48550/arXiv.1603.08029.

31. Sarwinda, D., Paradisa, R. H., Bustamam, H., & Anggia, P. Deep Learning in Image Classification using Residual Network (ResNet) Variants for Detection of Colorectal Cancer. *Procedia Comput. Sci.*, 2021, vol. 179, pp. 423–431. DOI: 10.1016/j.procs.2021.01.025.

32. Alom, Z., & et al. The History Began from AlexNet: A Comprehensive Survey on Deep Learning Approaches. *arXiv:1803.01164*, 2018. DOI: 10.48550/arXiv.1803.01164.

33. Das, S., & et al. Estimation of Road Boundary for Intelligent Vehicles Based on DeepLabV3+ Architecture. *IEEE Access*, 2021, vol. 9, pp. 121060–121075. DOI: 10.1109/ACCESS.2021.3107353.

34. Tammina, S. Transfer learning using VGG-16 with Deep Convolutional Neural Network for Classifying Images. *Int. J. Sci. Res. Publ. IJSRP*, 2019, vol. 9, iss. 10, article no. p9420. DOI: 10.29322/IJSRP.9.10.2019.p9420.

35. Salian, S. R., & Sawarkar, S. D. Melanoma Skin Lesion Classification Using Improved EfficientnetB3. *Jordanian J. Comput. Inf. Technol.*, 2022, vol. 8, iss. 1, pp. 45-56. DOI: 10.5455/jjcit.71-1636005929.

36. Shoeibi, A., Marjane, K., Mahboobeh, J., Navid, G., Delaram, S., Parisa, M., Ali, K., Roohallah, A., Sadig, H., Assef, Z., Alizadeh, S. Z., Fahime, K., Saeid, N., Rajendra, A. U., & Juan, M. G. Automated detection and forecasting of COVID-19 using deep learning techniques: A review. *Neurocomputing*, 2024, vol. 577, article no. 127317. DOI: 10.1016/j.neucom.2024.127317.

37. Munadi, K., Saddami, K., Oktiana, M., Roslidar., Muchtar, K., Melinda., Muharar, R., Syukri, M., Abidin, T. F., & Arnia, F. A Deep Learning Method for Early Detection of Diabetic Foot Using Decision Fusion and Thermal Images. *Appl. Sci.*, 2022, vol. 12, iss. 15, article no 7524. DOI: 10.3390/app12157524.

38. Tan, M., & Le, Q. V. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. *ArXiv*. 2019, vol. abs/1905.11946. DOI: 10.48550/arXiv.1905.11946.

39. Magdalena, R., Saidah, S., Fuadah, Y. N., Ubaidah, I. D. S., Herman, N., & Ibrahim, N. Convolutional Neural Network for Anemia Detection Based on Conjunctiva Palpebral Images. *Jurnal Teknik Informatika*, 2022, vol. 3, iss. 2, pp. 349-354. DOI: 10.20884/1.jutif.2022.3.2.197.

40. Goodfellow, I., Bengio, Y., & Courville, A. *Deep learning*. Alanna Maldonado, 2023. 804 p.

Received 17.05.2024, Accepted 17.02.2025

ВПЛИВ ВИКОРИСТАННЯ РІЗНИХ РЕНТГЕНОВСЬКИХ ДАНИХ ДЛЯ ВИЯВЛЕННЯ ТУБЕРКУЛЬОЗУ НА ОСНОВІ ГЛИБОКОГО НАВЧАННЯ

Мухаммад Ірхамсіях, Курrата А'юні, Хайрун Саддамі, Насаруддін Насаруддін, Хайрул Мунаді, Фітрі Арнія

Предмет статті, що ознаки туберкульозу важко вивчити візуально. Тому для розпізнавання рентгенівських зображень застосовується комп'ютерна система, заснована на глибокому навчанні. Багато робіт було проведено в цій області, але ще не досягнуто високого рівня точності. **Метою** нашої роботи — є виявлення ефекту використання різних наборів даних для розробки моделі глибокого навчання. **Використаний** метод полягає в розробці моделі глибокого навчання згорткової нейронної мережі (CNN) з використанням трансферного навчання для класифікації рентгенівських зображень на бінарні класи норми та туберкульозу (ТБ). В якості архітектури CNN використовуються попередньо навчені мережі ResNet та EfficientNet, а також їхні варіанти. Попередньо навчена мережа була навчена на наборі даних, отриманих з чотирьох різних джерел: Шеньчжень, Монтгомері, RSNA CXR та Білорусь. Набір даних розділено на три схеми: Перша схема складається з набору даних з Шеньчженя з низькоякісними рентгенівськими знімками; друга схема - це набори даних з Монтгомері, RSNA та Білорусі, які показують хороший контраст у зазначеній області туберкульозу; і третя схема містить набори даних з усіх джерел, щоб дозволити вивчити більше наборів даних. Для підвищення ефективності навчання також було застосовано регуляризацію доповнення, відсіювання та L2 регуляризацію. Було отримано такі результати: моделі краще працювали з високоякісними рентгенівськими знімками набору даних Схеми два, але не з великим набором даних Схеми три. Щодо продуктивності мережі, моделі, отримані за допомогою ResNet-101 та EfficientNetB0, перевершують інші завдяки гарному навчанню та здатності розпізнавати рентгенівські зображення з точністю 99,2%. **Підсумовуючи**, найкращим підходом до підвищення ефективності навчання є використання високоякісних вхідних даних та застосування регуляризацій.

Ключові слова: туберкульоз (ТБ); Коволюційна нейронна мережа (CNN); ResNet; EfficientNet.

Мухаммад Ірхамсія – ступінь магістра, докторська програма, Інженерна школа, аспірантура, Universitas Syiah Kuala, Banda Aceh, Індонезія.

Qurrata A'yuni – ступінь бакалавра, магістр, факультет електротехніки та комп'ютерної інженерії, Universitas Syiah Kuala, Banda Aceh, Індонезія.

ISSN 1814-4225 (print)
ISSN 2663-2012 (online)

Khairun Saddami – доктор технічних наук, доцент, кафедра електротехніки та комп'ютерної інженерії, Universitas Syiah Kuala, Banda Aceh, Індонезія.

Насаруддін Насаруддін – доктор технічних наук, професор кафедри електротехніки та комп'ютерної інженерії, Університет Сія Куала, Банда Ачех, Індонезія.

Хайрул Мунаді – доктор технічних наук, професор, кафедра електротехніки та комп'ютерної інженерії, Uni-versitas Syiah Kuala, Banda Aceh, Індонезія.

Фітрі Арнія – доктор технічних наук, професор кафедри електротехніки та комп'ютерної інженерії, Університет Сія Куала, Банда Ачех, Індонезія.

Muhammad Irhamsyah – Master degree, Doctoral Program, School of Engineering, Post Graduate Program, Universitas Syiah Kuala, Banda Aceh, Indonesia,
e-mail: irham.ee@usk.ac.id, ORCID: 0000-0002-7726-6283.

Qurrata A'yuni – Bachelor degree, Master Student, Department of Electrical and Computer Engineering, Universitas Syiah Kuala, Banda Aceh, Indonesia,
e-mail: qurrataayu@mhs.usk.ac.id.

Khairun Saddami – Doctor of Engineering, Assistant Professor, Department of Electrical and Computer Engineering, Universitas Syiah Kuala, Banda Aceh, Indonesia,
e-mail: khairun.saddami@usk.ac.id, ORCID: 0000-0001-8344-339X, Scopus Author ID: 57189353973.

Nasaruddin Nasaruddin – Doctor of Engineering, Professor, Department of Electrical and Computer Engineering, Universitas Syiah Kuala, Banda Aceh, Indonesia,
e-mail: nasaruddin@usk.ac.id, ORCID: 0000-0002-2933-1562, Scopus Author ID: 55663355200.

Khairul Munadi – Doctor of Engineering, Professor, Department of Electrical and Computer Engineering, Universitas Syiah Kuala, Banda Aceh, Indonesia,
e-mail: khairul.munadi@usk.ac.id, ORCID: 0000-0002-7507-9476, Scopus Author ID: 6508326697.

Fitri Arnia – Doctor of Engineering, Professor, Department of Electrical and Computer Engineering, Universitas Syiah Kuala, Banda Aceh, Indonesia,
e-mail: f.arnia@usk.ac.id, ORCID: 0000-0001-6020-1275, Scopus Author ID: 14027791000.