DenseNet-CNN Architectural Model for Detection of Abnormality of Acute Pulmonary Edema

Cynthia Hayat
Correspondence: cynthia.hayat@ukrida.ac.id
Department of Information System
Krida Wacana Christian University
West Jakarta, Indonesia

Abstract-Acute pulmonary edema (EPA) is a condition of emergency respiratory distress that results from the sudden and rapid build-up of fluid into the lungs. Rapid screening of EPA patients is necessary so that radiologists can make the prognosis as early as possible. In addition, reliance on the expert's knowledge of reasoning also hinders the diagnostic process. This research was conducted by developing an architectural model for machine learning systems with a deep learning approach. With the concept of representative learning, the denseNet-CNN algorithm connects each layer to another by means of a feed-forward. The data used is Image CXR-14 specifically labeled pulmonary edema pathology. The size of each CXR-14 image is 1024 × 1024 with a value of 8 bits grayscale. The size of each CXR-14 image is 1024 × 1024 with a value of 8 bits grayscale. The architectural model development stages consist of the preparation stage, data resampling, data training and data testing. Optimizer parameters used are Adam's optimizer, learning rate of 0.0001 and weight decay = 1e-5 and the loss used is binary cross entropy. The resulting mean AUROC analysis showed the sensitivity value of the 10% dataset was 71.493% and the specificity value of 10.011% was obtained at the second hold of the k-fold cross validation method after holdout validation, so that the resulting model was valid. The detection system developed from the denseNet-CNN model is expected to help radiologists identify abnormalities in CXR images quickly, precisely, and consistently. The denseNet-CNN model is also developed in the form of a heatmap visualization by localizing the features you are looking for. With localization in the form of a heat map, detection of pathological abnormalities of PEA is easier to do and to be recognized.

Keywords: denseNet-CNN, EPA, abnormality detection

1. Introduction

Acute pulmonary edema (EPA) is a condition of urgency characterized by a rapid respiratory emergency caused by displacement and accumulation of fluid in the lungs. According to the source of the cause, EPA is divided into two types, namely cardiogenic EPA and noncardiogenic EPA. The high prevalence rate of EPA can be seen from a study involving approximately 600 hospitals in Europe, Latin America, and Australia. Data show EPA is present in 37% of patients with acute heart failure [1][2].

Fast and precise screening for patients with EPA cases is needed so that doctors can make the prognosis as early as possible. One of the initial screenings performed on EPA patients is by doing a chest X-ray examination, known as ChestX-Ray (CXR) screening. Generally, the reading and diagnosis of CXR images is performed by a radiologist by comparing the CXR images of EPA patients with normal CXR images to detect abnormalities [3 [4].

Dependence on the knowledge of an expert radiologist regarding the principles of anatomy, physiology, and pathology is a factor that can hinder making a diagnosis as early as possible [5]. Another difficulty in the process of detecting CXR images is the difficulty of expert radiologists in developing consistent reasoning techniques in reading CXR images while considering all common chest diseases that require a long time to diagnose a CXR image [6].

One solution to solve this problem is to develop a machine learning system. Amit Kumar Jaiswal's research, entitled Identifying Pneumonia in Chest X-Rays: A Deep Learning Approach uses the Mask-RCNN method, used a deep neural network that combines global and local features for pixel-based segmentation. The identification model proposed in the study achieved reasonably good performance after evaluating a dataset of chest radiographs depicting potential causes of pneumonia [7].

The approach recommended in this study is to use deep learning (DL) convolutional neural network models. The DL approach adopts the way the human brain works

in managing data as representative learning which is then classified into layers. Research by Huang, Liu, Weinberger, & van der Maaten conducted in 2017 shows the importance of layer depth, better accuracy, and efficient training will be achieved if CNN has a closer connection between the layers that are close to the input and the layers that are close to it. with output [8]. DenseNet has an architecture that connects each layer to another in a feed-forward manner. Densenet itself has several advantages, namely: reducing the vanishing-gradient problem, strengthening feature propagation, reusing features, and reducing the number of parameters [9][10][11].

The computer-assisted detection system is expected to help radiologists identify abnormalities in CXR images quickly, precisely, and consistently. Computers can be taught to read and process a very large number of CXR image scans in a short amount of time. to confirm the results found by the radiologist and potentially identify other findings that may have been found. The resulting model is an artificial intelligence mechanism that can direct radiologists to make better diagnostic decisions for patients.

2.   Methods

a.   Data collection

The CXR images used in the CNN architectural model developed in the study were obtained from the public dataset of NIH Clinical Center, a clinical research hospital for the National Institutes of Health based in Maryland - United States, called ChestX-ray14 (CXR-14). This public dataset is the largest CXR dataset available to the public, containing 112,120 anonymous CXR front view images derived from 30,805 patients including patients with advanced lung disease. Each dataset summary page also contains the license terms and citation requirements which can be accessed on TCIA datasets from Cloud Storage, BigQuery, or by using the NIH Clinical Center's Cloud Healthcare API [12][13]. One of the CXR-14 images used is shown in Figure 1.
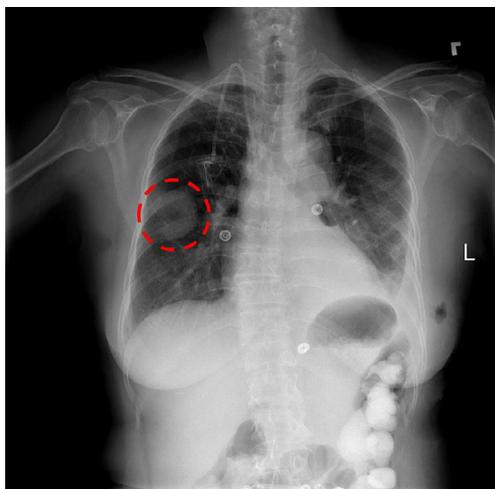

Figure 1. CXR image that identifies lung masses

The size of each CXR-14 image is 1024 × 1024 with a value of 8 bits grayscale as shown in table 1.

Table 1. Number of positive labels per pathology in the CXR-14 dataset

No.	Pathology	Number of Positive Labels	Percentage of the amount
1	Edema	2,303	2.05%

The number of positive labels for pulmonary edema was 2,303 with the assumption that there would be redundancy of patient data between existing pathology labels.

b.   DenseNet CNN Procedure

The proposed denseNet-CNN model development procedure is as follows:

Step 1: The preparation stage

In this preparation stage, analysis is carried out first to adjust the Deep Learning Framework to be applied with the availability of hardware in conducting training, validation and testing of the CNN model to be developed, as for the specifications of the hardware to be used are as follows:

- Desktop CPU: Intel (R) Core (TM) i3-8100 CPU @ 3.60GHz
- Memory: 8052MB
- VGA: NVIDIA Corporation GP107 [GeForce GTX 1050] Storage Media: 256GB Solid State Disk

Step 2: Re-Sampling Data

Modifications to the algorithm applied to the CXR-14 PEA dataset resulted in an underfit model due to imbalance data, therefore re-sampling the CXR-14 dataset, in order to provide valid results. The process of re-sampling this dataset begins with training on less data, which in this study was determined as much as 10% of the total dataset, which includes: all data with a positive PE label found on CXR-14, and the rest randomly chooses data labeled PE negative. [14] Re-sampling the dataset obtained is separated systematically using Python coding into training data, validation data and testing data. Resampling Dataset for Hold-out validation is illustrated in Figur 2 & 3.
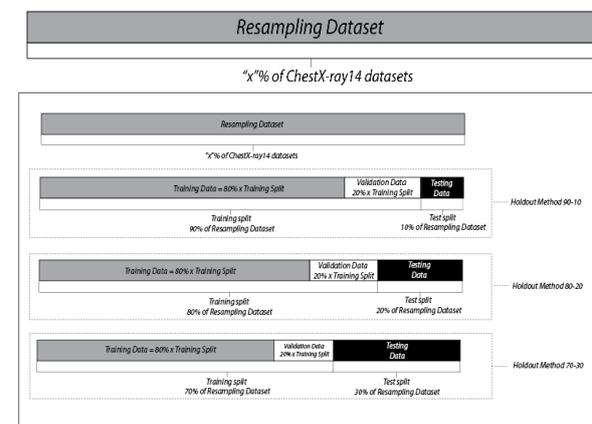

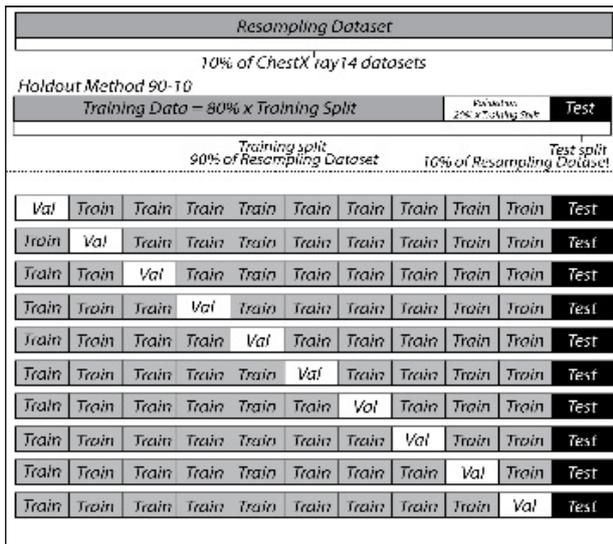**Figure 2. Resampling Dataset for Hold-out validation**

Figure 3. Resampling Dataset for Hold-out validation

Step 3: Training Stage

Furthermore, experiments and validations were carried out on the CheXNet scripting sources found on the github repository portal, this study determines the implementation of the PyTorch DL Framework made by Andrey G. (zoogzog) on the link https://github.com/zoogzog/chexnet . Referring to the selected scripting, some environmental elements for the DL framework need to be prepared first, namely using: [12]

- Operating System: Ubuntu 17.10
- Compilers: C (GCC) 7.2.0
- CUDA compilation tools: NVIDIA CUDA, V8.0.61

Experiments were carried out by replicating Python coding in the PyTorch environment that implements training and validation functions so that the model is in the form of a file that will be used in the testing stage. The optimizer used is Adam's algorithm as a standard parameter that can produce output quickly. The learning rate used is 0.0001, the weight decay is 1e-5, and the loss used is the cross entropy loss.

Step 4: Testing Phase

Furthermore, the testing process is carried out to produce AUROC performance against the PEA pathology contained in CXR-14. The distribution of the amount of training, validation, and testing data used in scripping for the DL framework is shown in table 2.

Table 2. Distribution of Total Data Training, Validation and Testing on Scripting

Image CXR	Training process	Validation process	Testing process
amount	78,468 data	11,219 data	22,433 data

Step 5: The resulting output

The AUROC performance obtained at this preparation stage is then compared with the results obtained by Rajpurkar et.al., in their journals as a reference in proceeding to the research implementation stage, as shown in table 3[15].

Table 3. Comparison of the results of testing the DenseNet algorithm between the results of Rajpurkar et.al., and the results of testing scripts for pulmonary edema

Patology	CheXNet-14	Result
Acute Pulmonary Edema	0.8878	0.9017
AUROC mean	0.841	0.8508

CNN's DenseNet Architecture Model Modifications

The architectural model developed using the DenseNet CNN algorithm which has advantages in implementing the depth of a layer, better accuracy, and efficient training will be achieved if CNN has a closer connection between the layer close to the input and the layer close to the output. Densenet CNN has an architecture that connects each layer to another by means of feed-forward [16][17][18].

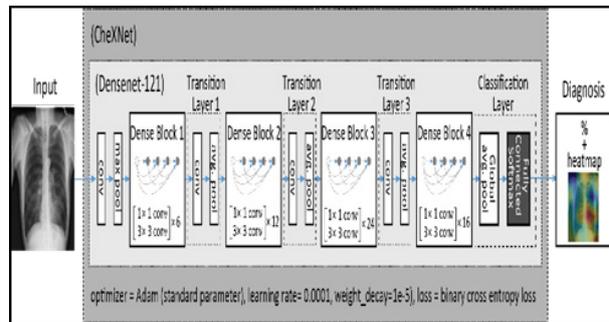The modification of CNN's DenseNet architectural model in this study can be seen in Figure 4.

Figure 4. CNN's DenseNet Architectural Model for EPA Disease Abnormality Detection

Generally, the CNN architecture consists of the input layer, hidden layer and output layer stages. CNN's DenseNet architecture above by skipping processes in hidden layers. The effect of the skip hidden layer is that the loss value from the previous layer can be carried to the next network so that the loss after the skip process is a combination of loss with the loss brought before by the skip connection. In this study, the skip connection added to the CNN architecture with the DenseNet algorithm can minimize the vanishing-gradient problem so that feature propagation can be strengthened, repetitive features on features, and reduces the use of parameters in the architectural model being developed.

Adam Optimizer was chosen because it is the most popular algorithm and it can produce output faster and better than other methods. The optimization done by Adam can be used as a substitute for the classic stochastic gradient descent procedure so that it can update the network weight repetitively on features based on training data. The stochastic gradient descent function maintains one learning rate for all weight updates used, where the learning rate will not change during the training process. [19] [20]

The learning rate used is 0.0001. The learning rate is maintained for each network weight (parameter) and is adapted separately as learning unfolds. With Adam's algorithm it is also possible to calculate individual adaptive learning rates for parameters different from the predicted first and second moments of the gradient. [21]

In the classification layer, a full connected softmax and a global average pool will be generated where the diagnoses of how many abnormalities are detected in the CXR-14 heatmap image.

2. Result

a. Holdout Validation at 10% Chest X-ray 14 Dataset

The distribution of the dataset for this method is carried out in accordance with the value of the distribution of training, validation and testing data shown in table 4, with the method scheme illustrated in Figure 5 with a value of x = 10.

Table 4. Holdout Validation Method Dataset Resampling Label on 10% Dataset CXR-14

Method	Data Training	Data Validation	Data Testing
Holdout 90-10	8.072	2.018	1.121
Holdout 80-20	7.175	1.793	2.242
Holdout 70-30	6.278	1.569	3.363

The training is carried out using the same Python programming as the 100% dataset training. The best results of modeling the three holdout validation methods at 10% of the ChestX-ray14 dataset are shown in table 4. The best mean AUROC value achieved is 0.9114 with Binary Cross Entropy (BCELoss) between the target and the output is 0.3238 which is obtained from the 90- division. 10. , determining the amount of training data, validation and schematic testing is shown in Figure 5. Graphic images of BCELoss on each batch of epoch validation are shown in Figure 6.

The results of model accuracy with three hold-out validation methods at 10% of the CXR-14 dataset are shown in table 5.
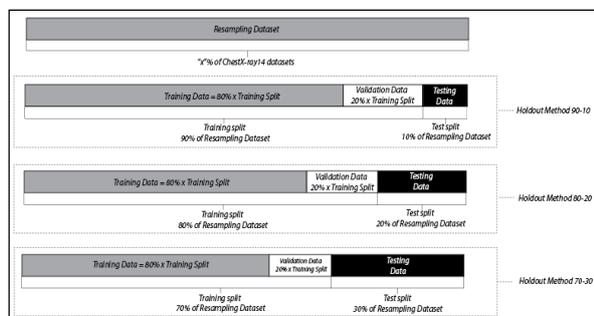

Figure 5. Schematic of determining the training data, validation and testing of the hold-out validation method
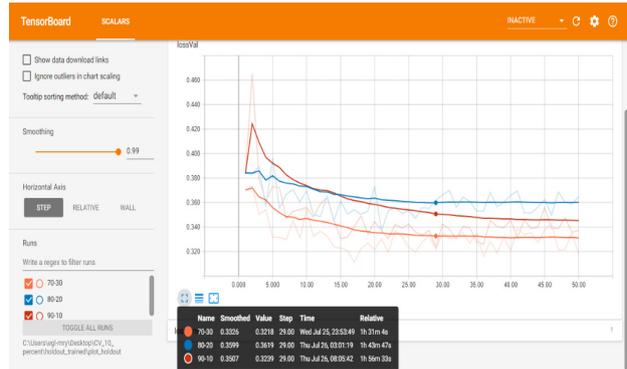

Figure 6. Graph BCELoss holdout validation method on 10% of ChestX-ray14 dataset

Table 5. Holdout Validation Model Accuracy Results on 10% Chestx-Ray 14 Dataset

Method	AUROC mean	BCELoss	Epoch#
Holdout 90-10	**0.9114**	**0.3238**	**29**
Holdout 80-20	0.8992	0.3369	21
Holdout 70-30	0.8940	0.3110	17

The number of positive PE data labels and negative PE data labels on the 90-10 holdout validation method at 10% of the ChestX-ray14 dataset is mentioned in table 6.

Table 6. Number of PE detection labels on Modified 90-10 Holdout Validation Dataset on 10% Chestx-Ray 14 Dataset

	Training	Validation	Testing	Total
Label Positif	1.692	389	222	2.303
Label Negatif	6.380	6.380	1.629	899
Total	8.072	8.072	2.018	1.121

b. Application of the K-fold cross validation over holdout validation method on 10% of the ChestX-ray 14

Furthermore, the K-fold cross validation over holdout validation method is applied, with a value of k = 10 (10 fold) based on the best 10% ChestX-ray14 dataset that has been obtained from the holdout validation method previously carried out, namely in the 90-10 division, determining the amount of data. training, validation and schematic testing are shown in Figure 7.

The results obtained from the modeling method of 10 fold cross validation over holdout validation on 10% of the ChestX-ray14 dataset are shown in Table 7. The best mean AUROC produced was 0.9164, with a BCELoss value of 0.3167, which was achieved by the 2nd fold in the 42nd batch epoch validation of the 50 defined epoch batches. Graphic of BCELoss achievement in each epoch batch is shown in Figure 8.
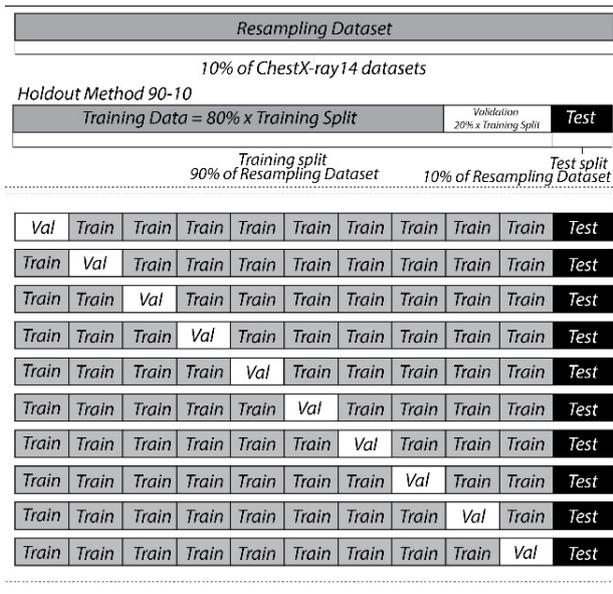
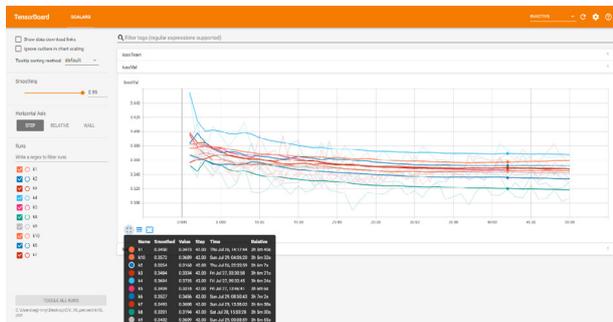**Figure 7. Schematic of determining training data, validation and testing of the k-fold cross validation method**

Figure 8. Graph of BCELoss k-fold validation method on 10% of the ChestX-ray 14 dataset

Table 7. Modeling Accuracy of the K-Fold Cross Validation Method Over Holdout Validation on the CXR-14 10% Dataset

Method	AUROC mean	BCELoss	Epoch#
Fold-1	0.9082	0.3225	27
Fold-2	**0.9164**	**0.3167**	**42**
Fold-3	0.9154	0.3274	24
Fold-4	0.9079	0.3484	19
Fold-5	0.9082	0.3226	18
Fold-6	0.9056	0.3249	25
Fold-7	0.9100	0.3291	17
Fold-8	0.9125	0.2894	22
Fold-9	0.9157	0.3190	25
Fold-10	0.9118	0.3319	26

c.   AUROC analysis of 10% ChestX-ray 14 dataset

The area under receiver operating characteristics (AUROC) is a performance metric that you can use to evaluate a classification model. The analysis was to

determine the sensitivity of the best model produced, namely the second fold of the k-fold cross validation method after holdout validation. The calculation of the mean AUROC is shown in Figure 9 and a slice of the output dataset is shown in Table 8.

Table 8. AUROC calculation for a positive label and a negative label on the CXR-14 image

Line #	AUROC	Line #	AUROC
1	0.9152	1	0.0244
2	0.8589	2	0.6133
3	0.8710	3	0.0576
…	…	…	…
220	0.8219	1796	0.0054
221	0.2529	1797	0.0417
222	0.7647	1798	0.1170
Pieces of AUROC calculation data for detection of positive PE labels in 10% of the CXR-14 dataset		Pieces of AUROC calculation data for detection of positive PE labels in 10% of the CXR-14 dataset	

Figure 9. The results of AUROC calculations on 10% of the ChestX-ray 14 dataset

From the calculation results, it is known that the sensitivity value has increased significantly so that it is above the cut-off value, with compensation there is an increase in the specificity value, which means that the resulting model has increased the likelihood of getting a true positive value when detecting a CXR-14 image with a PE label. positive is better, although there is a decrease in the confidence value of the negative PEA label training results.

With the results of the analysis obtained, it is concluded that the resulting model is valid, with a better

sensitivity value obtained from the 10% dataset of 71.493% and a specificity value of 10.011%. The comparison of the model test results is shown in table 9.

Table 9. Model Measurement Results

	10% dataset on (k-fold 2)
Total Dataset	11.211
Data Training	8.072
Data Validation	2.018
Data Testing	1.121
AUROC mean	0.9164
Sensitivity	71.493%
Specificity	10.011%

The research continues to the next stage by bringing the best model file obtained, from modeling 80-20 holdout validation on 10% of the CXR-14 dataset as a model for testing CXR input in detecting PEA pathology by displaying a heat map to visualize areas on CXR-14, where disease is present, by using the Class Activation Mapping. Implementing this application can assist medical personnel in localizing the features they looking for.

Figure 10. The result of heat map visualization on the resulting CNN Dense-Net model

3.   Conclusion

The valid model was obtained in the 6th modification of the second k-fold with a mean AUROC value of 0.9164, a better sensitivity value was obtained from the 10% dataset of 71.493% and a specificity value of 10.011%. Optimizer parameters used are Adam's optimizer, learning rate of 0.0001 and weight decay = 1e-5 and the loss used is binary cross entropy.

The denseNet-CNN model is also developed in the form of a heatmap visualization by localizing the features to be searched. The architecture is able to recognize predictive information with localization in the form of a heat map. This makes easier to detect and recognize PEA pathologic abnormalities.

Reference

[1]   S. H. Rampengan, "EDEMA PARU KARDIOGENIK AKUT," J. BIOMEDIK, 2014, doi: 10.35790/jbm.6.3.2014.6320.

[2]   H. Nendrastuti and M. Soetomo, "Edema Paru Akut Kardiogenik Dan Non Kardiogenik," Maj. Kedokt. Respirasi, 2010.

[3]   M. Irawaty, "Penatalaksanaan Edema Paru pada Kasus VSD dan Sepsis VAP Treatment of Lung Oedema in VSD and VAP Sepsis," Anest. Crit. Care, 2010.

[4]   A. Jatu and Lusiana, "Peranan Epitel Alveoli pada Edema Paru Non-kardiogenik," Ckd, 2015.

[5]   C. Hayat and B. Abian, "The modeling of artificial neural network of early diagnosis for malnutrition with backpropagation method," 2018, doi: 10.1109/IAC.2018.8780505.

[6]   K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," 2016, doi: 10.1109/CVPR.2016.90.

[7]   A. K. Jaiswal, P. Tiwari, S. Kumar, D. Gupta, A. Khanna, and J. J. P. C. Rodrigues, "Identifying pneumonia in chest X-rays: A deep learning approach," Meas. J. Int. Meas. Confed., 2019, doi: 10.1016/j.measurement.2019.05.076.

[8]   G. Huang, S. Liu, L. Van Der Maaten, and K. Q. Weinberger, "CondenseNet: An Efficient DenseNet Using Learned Group Convolutions," 2018, doi: 10.1109/CVPR.2018.00291.

[9]   G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," 2017, doi: 10.1109/CVPR.2017.243.

[10]   T. NURHIKMAT, "IMPLEMENTASI DEEP LEARNING UNTUK IMAGE CLASSIFICATION MENGGUNAKAN ALGORITMA CONVOLUTIONAL NEURAL NETWORK (CNN) PADA CITRA WAYANG GOLEK," UNIVERSITAS ISLAM INDONESIA. 2018.

[11]   S. Albawi, T. A. Mohammed, and S. Al-Zawi, "Understanding of a convolutional neural network," 2018, doi: 10.1109/ICEngTechnol.2017.8308186.

[12]   X. Wang, Y. Peng, L. Lu, Z. Lu, M. Bagheri, and R. M. Summers, "ChestX-ray8: Hospital-

scale chest X-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases," 2017, doi: 10.1109/CVPR.2017.369.

[13] E. T. Nader, "Chest X-ray interpretation," in Perioperative Assessment of the Maxillofacial Surgery Patient: Problem-based Patient Management, 2018.

[14] G. Huang, S. Liu, L. Van Der Maaten, and K. Q. Weinberger, "CondenseNet: An efficient densenet using learned group convolutions," arXiv. 2017.

[15] S. Ramiz and M. Rajpurkar, "Pulmonary Embolism in Children," Pediatric Clinics of North America. 2018, doi: 10.1016/j.pcl.2018.02.002.

[16] Y. Bengio, "Learning deep architectures for AI," Found. Trends Mach. Learn., 2009, doi: 10.1561/2200000006.

[17] P. Rajpurkar et al., "CheXNet: Radiologist-level pneumonia detection on chest X-rays with deep learning," arXiv. 2017.

[18] N. Kalchbrenner, E. Grefenstette, and P. Blunsom, "A convolutional neural network for modelling sentences," 2014, doi: 10.3115/v1/p14-1062.

[19] Y. Zhang, J. Gao, and H. Zhou, "Breeds Classification with Deep Convolutional Neural Network," 2020, doi: 10.1145/3383972.3383975.

[20] C. Neural and N. Accelerator, "ISAAC : A Convolutional Neural Network Accelerator with I n- S itu A nalog A rithmetic in C rossbars," Iscas, 2016.

[21] R. M. PRASMATIO, B. Rahmat, and I. Yuniar, "Deteksi Dan Pengenalan Ikan Menggunakan Algoritma Convolutional Neural Network," J. Inform. dan Sist. Inf., 2020.

Appendix 1. List of model test results

	Jumlah Dataset	Data Training	Data Validation	Data Testing	Prediksi *Pathology*	AUROC mean	Hasil	*Sensitivity*	*Specificity*	Keterangan
Persiapan	112.120	78,468	11,219	22,433	14	0.8508	*Valid*	-	-	CheXNet
Modifikasi 1	112.120	78,468	11,219	22,433	1	0.8973	*Underfit*	-	-	100% dataset
Modifikasi 2	11.211	8072	2.018	1.121	1	0.9114	*Valid*	-	-	10% dataset - Holdout 90-10
Modifikasi 3	11210	7.175	1.793	2.242	1	0.8992	*Valid*	-	-	10% dataset - Holdout 80-20
Modifikasi 4	11210	6.278	1.569	3363	1	0.8940	*Valid*	-	-	10% dataset - Holdout 70-30
Modifikasi 5	11.211	8.072	2.018	1.121	1	0.9082	*Valid*	-	-	10% dataset - k-fold 1
Modifikasi 6	11.211	8.072	2.018	1.121	1	0.9164	*Valid*	71.493%	10.011%	10% dataset - k-fold 2
Modifikasi 7	11.211	8.072	2.018	1.121	1	0.9154	*Valid*	-	-	10% dataset - k-fold 3
Modifikasi 8	11.211	8.072	2.018	1.121	1	0.9079	*Valid*	-	-	10% dataset - k-fold 4
Modifikasi 9	11.211	8.072	2.018	1.121	1	0.9082	*Valid*	-	-	10% dataset - k-fold 5
Modifikasi 10	11.211	8.072	2.018	1.121	1	0.9056	*Valid*	-	-	10% dataset - k-fold 6
Modifikasi 11	11.211	8.072	2.018	1.121	1	0.9100	*Valid*	-	-	10% dataset - k-fold 7
Modifikasi 12	11.211	8.072	2.018	1.121	1	0.9125	*Valid*	-	-	10% dataset - k-fold 8
Modifikasi 13	11.211	8.072	2.018	1.121	1	0.9157	*Valid*	-	-	10% dataset - k-fold 9
Modifikasi 14	11.211	8.072	2.018	1.121	1	0.9118	*Valid*	-	-	10% dataset - k-fold 10