JURNAL INFOTEL
Informatics - Telecommunication - Electronics
Website Journal: http://ejournal.st3telkom.ac.id/index.php/infotel
ISSN: 2085-3688; e-ISSN: 2460-0997

Convolutional Neural Networks Based on Raspberry Pi for a Prototype of Vocal Cord Abnormalities Identification

Hertiana Bethaningtyas D.K.[1], M. Agfian Fadillah[2*], Lulu Millatina R.[3], Maiisy Jahja[4], Asep Suhendi[5]
[1,2,3,4,5]Telkom University
[1,2,3,4,5]Jl. Telekomunikasi, Bandung 40257, Indonesia
*Corresponding email: muhammadagfian96@gmail.com

Received 19 June 2020, Revised 24 June 2020, Accepted 22 July 2020

Abstract – This study aims to make a device prototype for identifying vocal cord abnormalities based on Raspberry Pi. This prototype could classify the abnormalities into seven classes, i.e., cysts, granulomas, nodules, normal, papilloma, paralysis, and no vocal cords. The applied method to classify is a deep learning algorithm, mainly using Convolutional Neural Network (CNN). In building the CNN model, we used a statistical method to form a model training scenario, also modified the AlexNet architecture model by optimizing the parameters. The optimized parameters in the test scenario obtained 95.35% accuracy. The CNN model implemented on the Raspberry Pi, and the test results obtained 79.75% accuracy.

Keywords – identification of vocal cord abnormalities, image processing, CNN, Raspberry Pi.

I. INTRODUCTION

High and low tones are needed in communication and singing. These tones depend on the tension of the vocal cord [1]. The vocal cord is the narrowest breathing apparatus [2]. When the area is abnormal, it will show some symptoms also some gripes. Thus, health workers will diagnose these symptoms and gripes. Diagnosis is one of the mattering steps during the examination of the disease. Diagnosis of errors can cause mishandling that increases the chance of death [3]. In the vocal cord cases, the examination process is carried out using a laryngoscopy or a stroboscopy device or both. Stroboscopy is an examination of the condition of vocal cords, such as their anatomy, function, and biomechanism [4]. The observer will check and view the vocal cords using a camera which passed through the nose into the throat and diagnosed the result. However, each observer may give different opinions during the examination process based on their capacities and experiences. Through digital image processing, it can help the observer in determining the conditions as well as abnormalities in the examined vocal cord.

Digital images are formed by a collection of dots called pixels (pixel or picture element) [5]. A digital image is described by a [m, n] matrix in 2D [6]. In a previous study, Bima created a system to help observers detect vocal cord abnormalities using the Moore Neighbor Tracing image processing method [7]. The results obtained an accuracy rate of 85.83% from the 120 tested data. There are only four classes on the tested vocal cord abnormalities, i.e., paralysis, papilloma, granuloma, and nodules/cyst. However, there are still some deficiencies in the system, such as requires a lot of user assistance to obtain the diagnostic results, e.g., rotation, fitting, and multiple grayscale settings. Another study created a similar system as well as using multiple image processing methods [8]–[10]. Both studies could classify the vocal cord conditions and the abnormalities into sixth classes, i.e., normal, nodules, cyst, granulomas, papilloma, and paralysis. They applied the Chan-Vese algorithm to automatically obtain the glottis segmentation area so that the system could run without further initialization. However, all three studies could be performed only if the Personal Computer (PC) has installed MATLAB. Also, it could not perform in real time.

Still, image processing, Luan *et al.* [11] also using Region-Based Convolutional Network (R-FCN) to detect lesion of the larynx and classify the lesion area. They use a sufficient image dataset yet could identify the target. However, they still need some improvement to classify the lesion.

Matava *et al.* using Convolutional Neural Network (CNN) through laryngoscopy video to identify vocal cord real-time [12]. The best CNNs model and architecture that they had performed are ResNet and Inception. However, it could only detect the location of the vocal cord and trachea ring but not to classifying abnormalities from the vocal cord.

Based on the background, this study aims to create a device prototype for identifying vocal cord abnormalities based on Raspberry Pi. Raspberry Pi is chosen as its portability that can easily be moved. The devised system using deep learning with the convolutional neural network (CNN) method and could identify in real-time. CNN is usually used in image recognition and pattern detection [13]. CNN can learn how to extract image features once and how to classify them [14]. With this method, parameter settings are no longer performed to obtain predictive results so that it will be more comfortable in terms of software use. This prototype is expected to help doctors diagnose vocal cord conditions, especially vocal cords abnormalities and medical technology development in the future.

II. RESEARCH METHOD

A. Vocal Cord Abnormalities

Vocal cord abnormalities such as cysts, granulomas, nodules, papilloma, and paralysis are structural disorders by some larynx lesions. The observed characteristic while examining the patient is the physical form of the vocal cords in the image. Fig.1 shows the image of vocal cord abnormalities as well as the normal one [7].

B. Prototype Design

There are three main components in the devised prototype system, i.e., camera, Raspberry Pi, and display monitor. Camera's primary function as an image acquisition device. Raspberry Pi then processes the image from the camera and performs image processing. The classification result is then displayed through the display monitor. The prototype design is shown in Fig.2.

C. Data Preparation and Pre-processing

We obtained image data from previous work [7] with 120 images in total. Firstly, the image is pre-processed to get an equal data standard. The pre-processing step is cropping, gray scaling, and resizing to 64 × 64 pixels. For training a CNN model, it requires loads of data. Data augmentation was then performed to increase the owned data [15]. Some of the techniques chosen to perform are width shift, height shift, zoom range, horizontal flip, and rotation range. These techniques can manipulate data without missing any critical information. Total 7100 data that we obtained from the augmentation data process.

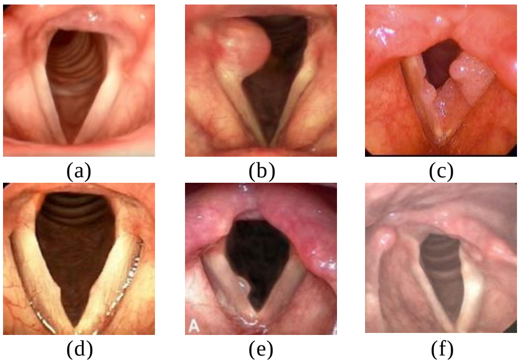


Fig.1. Vocal Cord Conditions; (a) normal, (b) granulomas, (c) papilloma, (d) nodule, (e) cyst, and (f) paralysis

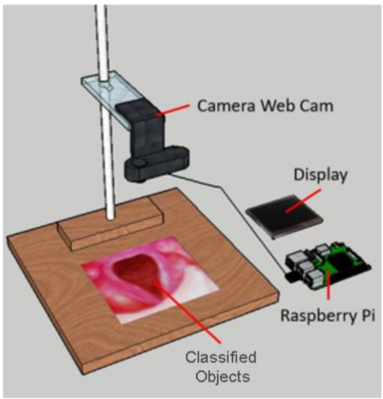
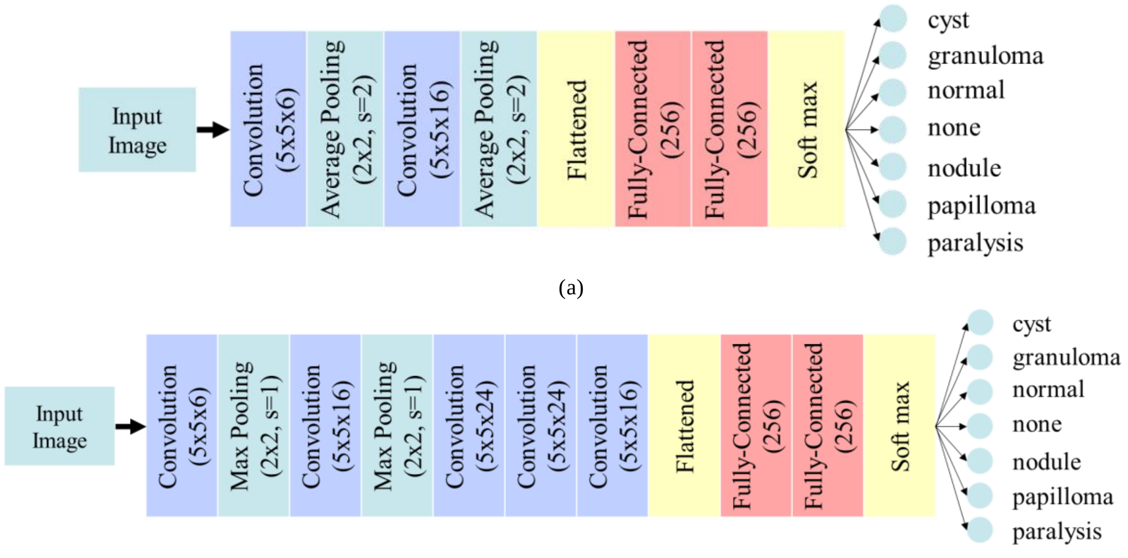


Fig.2. Proposed Prototype Design

Validation is a crucial step when forming a model so that the model can generalize new data. One of the validation techniques is hold out. Hold out distribute the data into three parts, i.e., training dataset, validation dataset, and test dataset. Training dataset and validation dataset used during the training process while test dataset used to test the obtained model from the training process. Training dataset, validation dataset, and test dataset are distributed to 80:10:10 data allocation.

D. CNN Architecture

The architecture design model in CNN affects the model performances. Two models are chosen to be trained with the dataset as its history on ImageNet Large Scale Visual Recognition Challenge (ILSVRC). The First model is LeNet-5. LeNet-5 was found by Yann Lecun. This architecture was chosen to be an option in the network architecture. LeNet5 is shown in Fig.3(a). The second model is AlexNet as the winner of ILSVRC 2012, with an acquired error of 16.4% from 1000 images classification [16]. It has eight layers. Because the depth of AlexNet is too severe for running in real-time videos in Raspberry Pi, we modified the architecture with 1/16 reduction for each depth convolution. We changed the input size in the grayscale dimension (shown in Fig.3(b)). This modified AlexNet model later called our proposed model or modified AlexNet.



(a)

(b)

Fig.3. CNN Architecture: (a) LeNet-5 and (b) modified AlexNet/proposed model

E. Training Model

Training parameters also affect the model performances. Several parameters have a significant effect on the model, i.e., input dimensions, epoch, batch size, iterations, dropout, and learning rate. Some parameter observation is needed to get the optimum parameters in the training process and the CNN architecture. Input dimensions, pooling layers type, convolution kernel size, learning rate, dropout, and epoch are the parameters that variated due to the observation in this study. The training model is performed by initiating the value of each parameter, then variate it gradually, and the best value of each parameter will be chosen.

F. Test Scenarios

After getting the optimum model from the training model, the model is tested using the test dataset. In the test process, the prototype had already integrated. A tablet screen is prepared in addition to showing the test dataset. The test process starts by showing the test dataset on the tablet screen. The camera then took a screenshot of the vocal cord images on the tablet screen. Raspberry Pi received the input images then processed it using the proposed CNN model, also give a prediction or classification result and shown by the display monitor. Fig.4 shows the flowchart of the CNN model test scenario.

III. RESULT

A. CNN Architecture Evaluation

Our proposed model/modified AlexNet and LeNet-5 are trained with initiation parameters to get the best architecture model. The initiating parameters are epoch, dropout, learning rate, convolution kernel size, and max pooling type. Both performances are evaluated with their accuracy over epoch value and loss over epoch value. Fig.5 shows the result of the training architecture model, the modified AlexNet, and LeNet-5. Fig.5(a) shows that the loss/error value and validation loss value of both modified AlexNet and LeNet-5 are not significant. It means that both models do not overfit. LeNet-5 has a shorter training time. However, modified AlexNet has a lower loss value than LeNet-5. Also, Fig.5(b) shows that the accuracy of modified AlexNet is higher than LeNet-5.

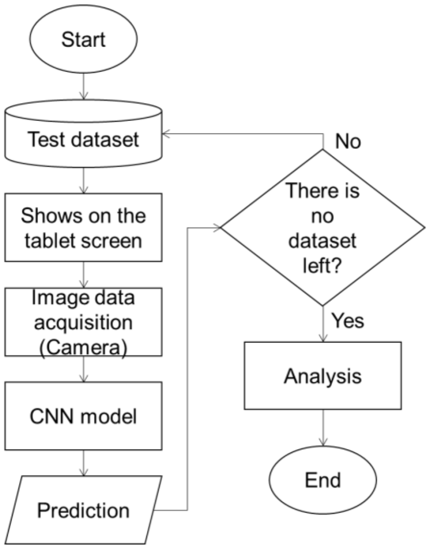
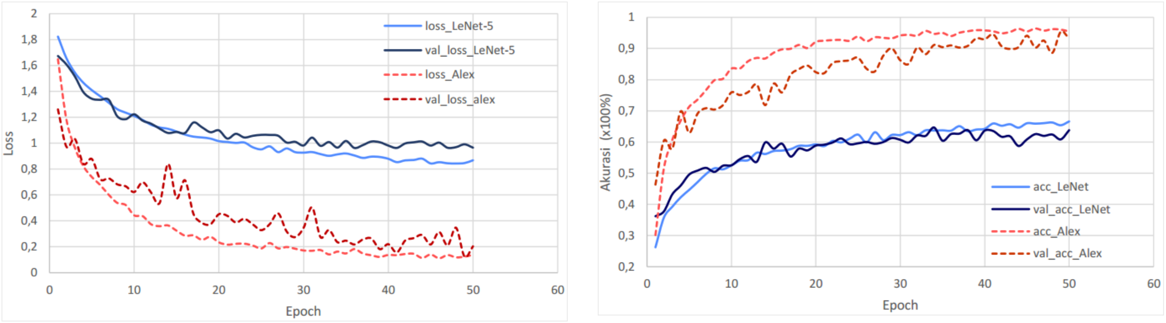

Start
Test dataset
Shows on the tablet screen
Image data acquisition (Camera)
CNN model
Prediction
There is no dataset left?
No
Yes
Analysis
End

Fig.4. Flowchart of CNN Model Test Scenario

Furthermore, the details are shown in Table 1. Thus, it can be said that the modified AlexNet has better performances than LeNet-5. The modified AlexNet could explore more complex/detail features (low-level features) as it has more in-depth architecture than LeNet-5. So that modified AlexNet is chosen as the applied CNN architecture in this study.



(a) (b)

Fig.5. Result of Architecture Performances: (a) loss chart over epoch value and (b) accuracy chart over epoch value

Table 1. Comparison of The Modified Alexnet and LeNet-5 Performances

Model	Training time (s)	Loss	Val. loss	Val. accuracy (%)
AlexNet	3961	0.138	0.204	93.24
LeNet-5	2588	0.868	0.966	63.80

B. *Results of Training Model*

Training is carried out using a laptop with data acquisition derived from previous work [7], and its distribution has been explained in Subsection II.C. Statistical methods were applied to do tuning parameters for getting the optimum parameters. This section provides the results of observation training model.

a) Input Dimensions

Input dimensions could affect model accuracy because it determines the amount input of information. If there is only a small amount of information, it may lose some vital information. However, too much information could make the computation higher and severe to run in Raspberry Pi.

We variated the input dimensions to 28×28, 32×32, and 64×64 pixels. The result shows in Table 2 and shown that model performances using 64×64 pixels input dimensions are the best cause it has the highest validation accuracy value and the lowest loss and validation loss value. The larger size would take more time to compute because it needs more layers to explore the image. However, the smaller size input image may lose some information and get a lower accuracy.

Table 2. Model Performance with Variated Input Dimensions

Input image	Training time (s)	Loss	Val. loss	Val. accuracy (%)
28×28	1485	0.164	0.318	88.17
32×32	1796	0.172	0.212	91.83
64×64	6153	0.138	0.204	93.24

b) Pooling Layer

Pooling layer is a reduction data dimension process. It would reduce the sensitivity model in noise and variations. There are two methods to do this process, using an average or maximum value from kernel window. We found that in this case study average method provides the best result. However, the average method provides the same result in other study cases. Furthermore, the details shown in Table 3.

Table 3. Model Performance with Variated Pooling Layer Types

Pooling layer	Training time (s)	Loss	Val. loss	Val. accuracy (%)
Max	6153	0.138	0.204	93.24
Average	6000	0.086	0.112	95.35

c) Convolution Kernel Size

Convolution kernel size affects the number of learning parameters. Commonly, the applied kernel size is 3 × 3, 5 × 5, and 7 × 7. The performance result shown in Table 4, we found that 5 × 5 and 7 × 7 kernel size has a similar validation accuracy, which higher than 3×3. Also, the smallest kernel size has the highest loss. It because smaller filters may collect more information and able to distinguish features at low-level yet require more in-depth architecture. While larger kernel size has a spacious area of observation and hard to differentiate detailed characteristics, thus we choose 5 × 5 as the convolution kernel size.

Table 4. Model Performance with Variated Convolution Kernel Sizes

Kernel size	Training time (s)	Loss	Val. loss	Val. accuracy (%)
3×3	4260	0.139	0.153	93.66
5×5	6000	0.086	0.112	95.35
7×7	8039	0.1013	0.141	95.63

d) Learning Rate

Learning rate adjusts how responsive the updated weight. A higher learning rate may reach the convergence point in a shorter time. Nevertheless, if the value is too big, then the weight alteration over error value will become too responsive and do not reach the convergence point. Otherwise, a lower learning rate may longer reach the convergence point and has a higher probability of reaching the convergence point.

We variated the learning rate to 0.01, 0.001, and 0.0001 and found that the stable learning rate value was 0.001 (shown in Table 5). Learning rate value at 0.01 failed to reach the convergence point as it has too high value, while learning rate 0.0001 needs more epoch to reach the convergence point. So that we choose 0.001 as the learning rate value.

Table 5. Model Performance with Variated Learning Rates

Learn-ing rate	**Training time (s)**	**Loss**	**Val. loss**	**Val. accuracy (%)**
0.01	5632	1.947	1.974	12.11
0.001	6000	0.086	0.112	95.35
0.0001	5623	0.257	0.316	87.04

e) Dropout

As regulate technique, dropout will deactivate some neurons, thus decreasing the overfitting trained model. In the initiation parameters, we used 0.03 as a dropout value. Based on the varying value, we found that the highest accuracy performed by the training model without dropout (shown in Table 6). However, we chose 0.03 as dropout value cause its differences between loss and validation loss is smaller than without dropout.

Table 6. Model Performance with Variated Dropout

Drop-out	**Training time (s)**	**Loss**	**Val. loss**	**Val. accuracy (%)**
With-out	4578	0.068	0.112	95.92
0.3	6000	0.086	0.112	95.35

f) Epoch

Epoch is one cycle of forward pass and backward pass for all of the datasets. We variated the epoch value to 10, 30, and 50. Based on Table 7, the highest validation accuracy is reached when the epoch value is 50. The lower value of epoch will be causing the weight not optimal, yet the model may not classify correctly, while an excessive epoch will merely be causing weights to memorize the training data. Thus, they may not recognize the characteristic of test dataset correctly.

Table 7. Model Performance with Variated Epoch

Epoch	**Training time (s)**	**Loss**	**Val. loss**	**Val. accuracy (%)**
10	1250	0.399	0.583	77.12
30	2999	0.104	0.186	93.52
50	6000	0.086	0.112	95.35

After tuning the parameters, we got the optimum parameters: input dimensions 64× 64 pixels, using average pooling layer, convolution kernel size 5×5, learning rate 0.001, not using any dropout value, and epoch 50. The result of training model performance before and after optimizing the parameters shown in Table 8. The optimized parameters model obtained 95.35% and have better performances than the model before optimizing the parameters.

Table 8. Comparison Between Before and After Optimizing The Parameter

Parameters	**Before optimization**	**After optimization**
Training time (seconds)	6153	6000
Loss	0.1377	0.0856
Validation loss	0.2037	0.1225
Validation accuracy (%)	93.24	95.35

C. *Results of CNN Model Test Scenario in Raspberry Pi*

After getting the training model and parameters optimized, the test is performed using a prototype and deployed the training model to Raspberry Pi. The confusion matrix of the tested model in Raspberry Pi shown in Table 9. It shows that nodules are hard to predict, yet granuloma is easier to predict correctly using the CNN model. Based on the table, from the 711 data, the CNN model could predict 567 data correctly. The accuracy count as follows,

$$accuracy = \frac{num.\, correct\ prediction}{num.\ data} \times 100\% \quad (1)$$

Thus, the accuracy is 79.75%. The test scenario was performed by the built prototype as shown in Fig.6. Also, the examples of classify results shown in Fig.7.

https://doi.org/10.20895/infotel.v12i3.487

Table 9. Confusion Matrix of The Tested Model in Raspberry Pi

		Prediction						
		Cyst	**Granuloma**	**Nodule**	**None**	**Normal**	**Papilloma**	**Paralysis**
Actual	**Cyst**	99	4	14	4	0	0	3
	Granuloma	3	92	2	4	0	0	1
	Nodule	31	0	44	3	2	0	9
	None	0	0	0	101	0	1	0
	Normal	4	4	12	2	86	0	1
	Papilloma	7	1	2	10	0	57	9
	Paralysis	1	1	0	8	0	1	88

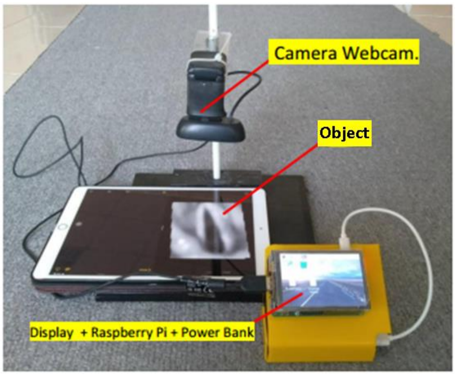


Fig. 4. The Built Prototype

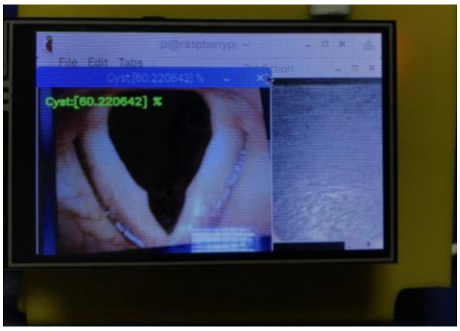


Fig.5. Screenshot of Display Monitor

IV. DISCUSSION

The causes of the test accuracy only get 79.75% is the built prototype acquire data from the tablet screen. In that process, much noise arises from the environment, such as the brightness of the light.

Specifically, the CNN model performance for each class is shown in Table 3. A 79.75% accuracy value means that almost all data is predicted correctly. Furthermore, in predicting each class, CNN model perfomance can be reviewed from recall and precision value.

In the table, recall value is higher than precision at paralysis, cyst, and none classes. For example, in none class, the model could classify all none datasets correctly, even though there are other datasets classified as none. The illustration can be seen in Fig.8(a). Otherwise, precision value is higher from recall at normal and papilloma classes. For example, in normal classes, the model could classify the normal dataset only some parts of the dataset. The illustration can be seen in Fig.8(b).

When the recall and precision values are equally high, it shows that the classification result is great as in granuloma classes, whereas nodules' recall and precision values are low, which shows that the modified and studied CNN model not suitable to classify nodules.

Table 10. Performance Result of Raspberry Pi

	Raspberry Pi		
Class	**Precision**	**Recall**	**Accuracy**
Cyst	68.28	79.84	79.75%
Granuloma	90.57	90.57	
Nodule	59.46	49.44	
None	76.52	99.02	
Normal	97.73	78.90	
Papilloma	96.61	66.28	
Paralysis	79.28	88.89	

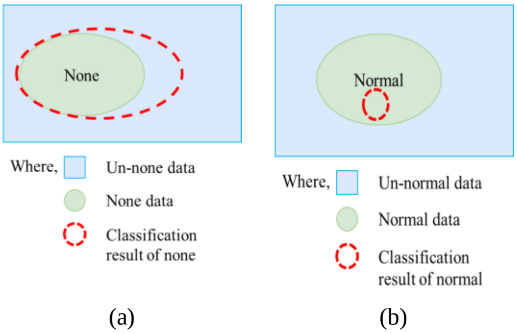


Fig. 6. Illustration of Recall and Precision Relation; (a) recall higher than precision and (b) precision higher than recall

https://doi.org/10.20895/infotel.v12i3.487

V. CONCLUSION

This study has developed a CNN model that is applied to the prototype using Raspberry Pi. Of the two training models considered, LeNet-5 and modified AlexNet, modified AlexNet was chosen as the training model because of its value of loss and validation loss is small without showing overfitting. We also optimized the training parameters of the modified AlexNet model. The performance of the training model with optimized parameters got 95.35% accuracy while the CNN model performance in Raspberry Pi got 79.75%. The accuracy value is smaller than the performance of training because of environmental noises when acquiring images on the prototype.

ACKNOWLEDGMENT

Thank to Dustira Hospital Cimahi which kindly lent the data for this study.

REFERENCES

[1] M. Ester and A. S. Augustinus, *Sistem pernapasan dan sistem kardiovaskular*. 1999.

[2] Anies, *Seri kesehatan umum pencegahan dini gangguan kesehatan: berbagai penyakit dan gangguan kesehatan yang perlu diwaspadai dan dicegah secara dini.* Jakarta: Pt Elex Media Komputindo, 2005.

[3] D. E. Newman-Toker *Et Al.*, "Serious misdiagnosis-related harms in malpractice claims: the 'big three'– vascular events, infections, and cancers," *Diagnosis*, vol. 6, pp. 227–240, 2019.

[4] Docdoc, "What is stroboscopy: overview, benefits, and expected results," 2020. Https:==Www:Docdoc:Com=Medicalinformation=Procedures=Stroboscopy.

[5] A. Kadir and A. Susanto, *Teori dan aplikasi pengolahan citra digital.* Yogyakarta: Andi, 2013.

[6] Young, I. T., Gerbrands, J. J., Vliet, and L. J. Van., *Fundamentals Of Image Processing.* 2007.

[7] B. G. L. Pubiyangga, "Identifikasi kondisi pita suara untuk deteksi kelainan pita suara dengan metode moore neighbor tracing," *Univ. Telkom*, P. 2016.

[8] H. Bethaningtyas, S. Suwandi, and C. D. Anggraini, "Sistem klasifikasi kondisi pita suara dengan metode decision tree," *J. Nas. Tek. Elektro Dan Teknol. Inf.*, vol. 8, no. 2, pp. 168, 2019, Doi: 10.22146/Jnteti.V8i2.506.

[9] H. Bethanigtyas, Suwandi, and C. D. Anggraini, "Classification System Vocal Cords Disease Using Digital Image Processing," *Proc. - 2019 Ieee Int. Conf. Ind. 4.0, Artif. Intell. Commun. Technol. Iaict 2019*, no. C, pp. 129–132, 2019, Doi: 10.1109/Iciaict.2019.8784832.

[10] C. D. Anggraini, "Identifikasi otomatis kelainan pada pita suara menggunakan teknologi pengolahan citra digital.," Telkom University, 2019.

[11] B. Luan, Y. Sun, C. Tong, Y. Liu, and H. Liu, "R-FCN based laryngeal lesion detection," *Proc. - 2019 12th Int. Symp. Comput. Intell. Des. Isc. 2019*, pp. 128–131, 2019, Doi: 10.1109/Iscid.2019.10112.

[12] C. Matava, E. Pankiv, S. Raisbeck, M. Caldeira, and F. Alam, "A convolutional neural network for real time classification, identification, and labelling of vocal cord and ttacheal using laryngoscopy and bronchoscopy video," *J. Med. Syst.*, vol. 44, no. 2, 2020, Doi: 10.1007/S10916-019-1481-4.

[13] S. Albawi, T. A. M. Mohammed, and S. Alzawi, "A data-driven approach to precipitation parameterizations using convolutional encoder-decoder neural networks pablo," *IEEE*, 2017, [Online]. Available: https://wiki.tum.de/display/lfdv/Layers+of+a+Convolutional+Neural+Network.

[14] Z. C. Horn, L. Auret, J. T. McCoy, C. Aldrich, and B. M. Herbst, "Performance of convolutional neural networks for feature extraction in froth flotation sensing," *IFAC-PapersOnLine*, vol. 50, no. 2, pp. 13–18, 2017, doi: 10.1016/j.ifacol.2017.12.003.

[15] B. Santosa and A. Umam, *Data mining dan big data analysis*, 2nd ed. Penerbit Penebar Media Pustaka: Yogyakarta., 2018.

[16] Chennupati and Sumanth., "Hierarchical decomposition of large deep networks.," Rochester Institute of Technology, 2016.