Preliminary Study on Indonesian Word Recognition for Elder Companion Robot

M.B.Nugraha[1], Dyah Ayu Anggreini Tuasikal[2], Ni Made Satvika Iswari[3]
Luthfialmas Fakhrizki Irwanto[4]

[1,2,4] Department of Electrical Engineering, Universitas Multimedia Nusantara, Tangerang, Indonesia
[3] Department of Informatics, Universitas Multimedia Nusantara, Tangerang, Indonesia
[1] mb.nugraha@umn.ac.id, [2] dyah.tuasikal@umn.ac.id, [3] satvika@umn.ac.id,
[4] lutfiaamas.fakhriski@student.umn.ac.id

Accepted on 06 July 2022
Approved on 15 July 2022

***Abstract*— Word recognition using deep learning is a simple approach to speech recognition in general. From this word-level recognition, the emotional expression recognition model. The emotion recognition model can be used to describe the important level of action on future planned hardware implementation. This research was conducted using MFCC as the feature extraction method from the audio data and using the CNN-LSTM approach for the emotional expression classifier. The model itself will be implemented into a humanoid robot to become a companion robot for the elderly. The model itself has 67% accuracy for emotion recognition and 97% accuracy for word recognition. However, the model only attained 20% accuracy in real-life testing using the humanoid robot as the model tends to overfitting as a result of the lack of data used in model training.**

***Index Terms*— CNN-LSTM; emotion recognition; MFCC.**

I. Introduction

Speech recognition recently become a popular research subject in Machine Learning and Artificial Intelligence research studies. The speech recognition studies also expand into emotion recognition derived from speech data [1]. This phenomenon occurs because emotion is one of the fundamentals of day-to-day communication between humans. Speech emotion recognition studies grow because researchers expect machines could learn how to distinguish an emotion by audio and visual data [2], [3]. As the machine's capability to recognize emotion in audio or visual data could augment user experience in several areas, especially in services such as automatic call centers, virtual reality games, etc. [4]

Previous studies in speech emotion recognition have already tried different approaches to building a learning network for emotional recognition. Wan [5] uses DTW in emotion recognition. But recently, deep neural networks using CNN and LSTM or RNN are gaining popularity for recognizing human emotion [6][7][8] and compared with established methods like SVM, LSTM and CNN have greater accuracy results [9]. And recently, the researcher also tried to recognize speech and emotion from their respective natural language, as shown in the study of Wan [5], Guiming [10], and Wang [1]. Specific language emotion recognition, in this case, Indonesian also studied by Lasiman [9]. Wunarso et al. [11] even tried to build another dataset for the Indonesian language in their study. As stated by Park et al. [12], deep neural network output really depends on how feature selection is done, and also how pooling and padding are important in improving speech recognition. They also stated that stacking many convolutional layers as they used in their work to create very deep neural networks does not have a great impact on recognition.

Word and emotion recognition technology can be used to build a companion robot for elders. Although in Indonesia the prevalence of loneliness in the elderly is not too high due to the strong eastern culture, symptoms of seniors who come from the middle to upper economy are already living alone without the company of their families. One of the solutions that can be given to this problem is to use a companion robot which is already widely used in Japan and the US. Robot companion is used by seniors who are in the middle to upper economic level and do have the appropriate understanding of technology. The companion robot function emphasizes the response that can be given by the robot based on voice input, so word recognition and speech recognition are important. And an additional feature that is also important is the robot's ability to detect danger or emergencies based on variations in intonation which will be the emotion recognition feature of this robot.

The purpose of this research is to do a preliminary study on emotion recognition in segmented words. Emotion classes were chosen based on a plan of implementation on the robotic system. In previous work, some authors of this paper successfully experimented with the word recognition system using MFCC for feature extraction and implementing the model on CNN. Hence, this work will discuss on speaker-independent word-emotion recognition system, focusing on emotional classification regardless of the speaker. This paper only discussed the method chosen for conducting training and validation of deep learning using CNN and LSTM.

II. Data Collection

In their study, Wunarso et al. [11] try to build an Indonesian speech-emotion database called I-SpeED and use SVM as the classifier method. While Lasiman [9] studied emotion recognition using a feed-forward neural network and LSTM for the Indonesian language.

In this research, the data was collected and built by using multiple audio files with .wav format. Audio files were recorded in Bahasa Indonesia, and based on three emotional expressions, "happy," "sad," and "angry." Speakers were asked to read ten words with their respective expressions. The words are chosen from the robot implementation scenario in future works. All used words are described in Table 1.

TABLE I. Word Used

Words				
Maju *(forward)*	Mundur *(backward)*	Kanan *(right)*	Kiri *(left)*	Tegak *(stand)*
Duduk *(sit)*	Tidur *(sleep)*	Joget *(dance)*	Cari *(find)*	Angkat *(lift)*

The word-emotion database recorded from 100 different speakers consists of 9000 audio files. Each speaker must speak each word 3 times for each emotional expression. Speakers consist of 50 males and 50 females. This approach is necessary because, in the previous study, the training and testing result is biased by serious overfitting caused by a lack of diversity in the database [13]. Also, the general expression of "sad" and "angry" between males and females has different energy and frequency.

Participants were first asked to fill out a simple questionnaire regarding their mood at the time. If the participant is in a very sad or angry condition, the recording will not include the "happy" state and will be rescheduled for another day. To induce the "happy", "sad" and "angry" emotional state, participants were asked to view a video for each emotional state. The videos are purely chosen by the data sample collector and to minimize bias before recording, participants were also asked once again about their emotional state after watching the video.

TABLE II. Voice Samples Gender Distribution

Words	Male	Female
Maju *(forward)*	450	450
Mundur *(backward)*	450	450
Kanan *(right)*	450	450
Kiri *(left)*	450	450
Tegak *(stand)*	450	450
Duduk *(sit)*	450	450
Tidur *(sleep)*	450	450
Joget *(dance)*	450	450
Cari *(find)*	450	450
Angkat *(lift)*	450	450
	4500	4500

III. Methodology

The features from audio data were extracted using the MFCC method before being fed into CNN-LSTM networks to classify audio file samples into three emotion classes. Each audio is preprocessed to have the same ±1000ms length with .wav file format. Shorter data will be added with zeroes, and longer data will be cut to fit in a 1000ms timeframe.

This work will use MFCC for the feature extraction method, as shown by several studies that MFCC has greater output accuracy compared with other methods like DTW, hence MFCC become one of the main methods to process audio samples to be used in deep learning [7][14]. CNN-LSTM is chosen to provide deep learning methods for emotion recognition. A detailed explanation of the processes in this work is given below.

A. MFCC

Preprocessed data will be further processed into MFCC to get the 2D representation of the spectrogram from audio data. This is necessary because the convolutional process in CNN requires all data represented in the image, in this case, a 2D spectrogram image. Using constant value padding, all output vectors from MFCC were fixed in size. Audio files used in the MFCC process have a 16kHz framerate and mono encoding. Output from MFCC extractions are 20x11 vector matrix.

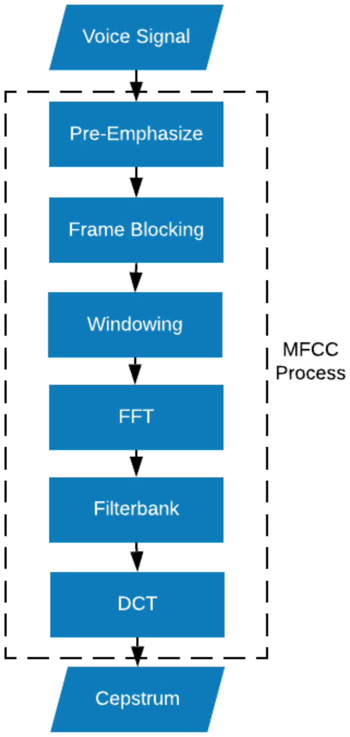


Fig. 1. Typical MFCC Process

B. CNN-LSTM

In this study, CNN-LSTM is used as a means of training, validating, and testing the learning and recognition model built in the research, by using the Keras library in Python. MFCC feature output from the process will be fed into the CNN network. The convolutional method in CNN will extract samples from dataset features provided in MFCC by convoluting the sample to extract diminishing features from the dataset. After the convolutional and pooling process, the fully-connected layer of CNN will be connected to the LSTM layers. The step-by-step of this process will be further explained below.

Convolution - we used 64 convolutional layers, with three by three kernels, to extract data from the sample with ReLU activation.

MaxPooling - to decrease the samples, we used two-by-two pool sizes with the "same" padding.

Dropout - The dropout method used for regularisation means to reduce the overfitting probability. 0.25 probability used in the Dropout layer

Convolution - 128 convolutional layers, with two by two kernels to further extract diminishing features from convoluted layers.

MaxPooling - another two-by-two pool size is used.

Dropout - same 0.25 probability used in this layer.

Flatten - flatten the output to become a fully connected layer, to make the sure output of CNN will be connected with the LSTM layer.

LSTM - we used two layers of LSTM to acquire information from the output of CNN with "ReLU" activation.

Dense - model will be condensed into 3 classes, with "Softmax" activation.

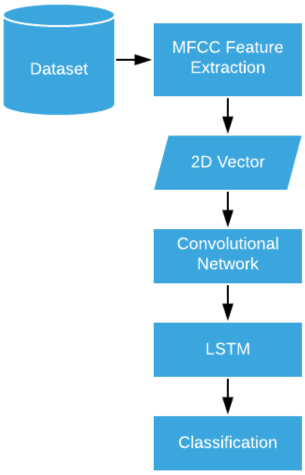


Fig. 2. Feature Extraction and Classification Process

C. CNN-LSTM

The hardware robot used in this research is a humanoid robot from UBTech. This robot will be dismantled and the main CPU in the system will be replaced with a Raspberry Pi board. The Raspberry Pi will be used as a place for running the machine learning model, and where a webcam is connected to the system. The microphone from the webcam will be used to record voice commands, which are then processed with MFCC and classified by CNN-LSTM layers. The result of the classification is a command for the robot, which will then be used as input for the Arduino Uno. This is because the robot's motor itself is driven using Arduino. The illustration of the robot will be shown in Figure 3. The wiring diagram of the robot will be shown in Figure 5.

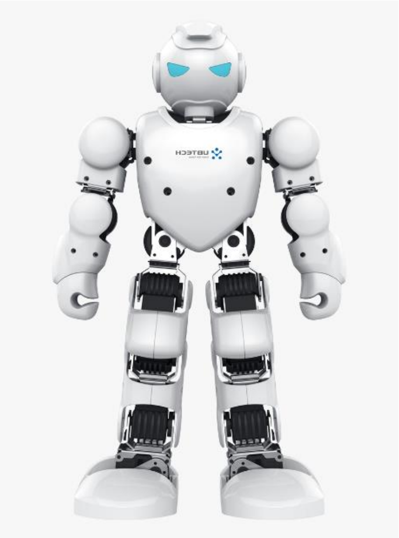


Fig. 3. Humanoid Robot used in Experiment

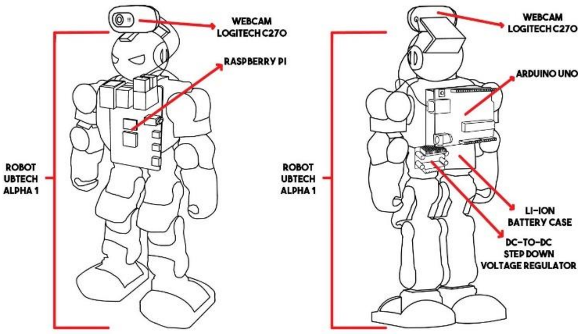


Fig. 4. Robot Design

The servo used in this robot is operated by sending a byte array to the servo. The servo itself is daisy-chained for each limb, so the Tx pin for moving the robot is split into 4 channels. The bytes array itself is consist of 10 bytes which is shown in Table III below.

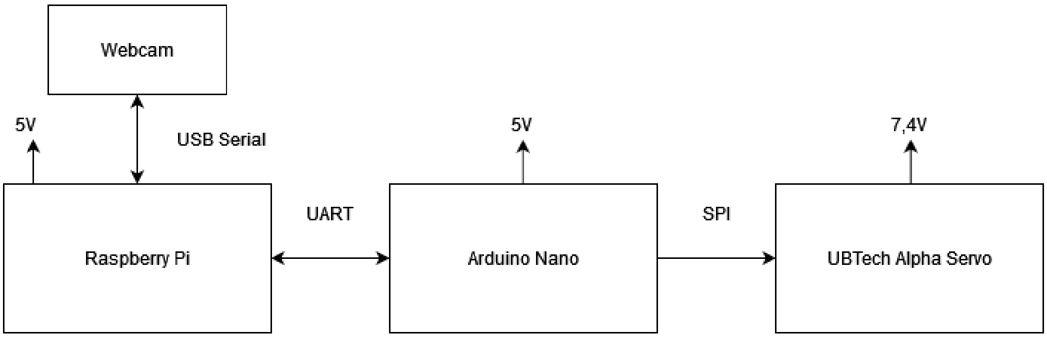


Fig. 5. System Block Diagram

The servo ID is already predetermined by the manufacturer. Op Mode is like the “servo.attach” command when using the Arduino Servo library, so the servo will be initialized and energized when “attached” and de-energized when “detached”. Mode 1 is for the “attach” function and Mode 2 is for the “detach” function. The degree is the value of desired servo degree, ranging from 0 to 180. Duration is for determining how fast the servo must attain the desired degree.

TABLE III. ROBOT MOVEMENT COMMAND

Byte 1	Header
Byte 2	Header
Byte 3	Servo ID
Byte 4	Op Mode
Byte 5	Degree
Byte 6	Duration
Byte 7	N/A
Byte 8	N/A
Byte 9	Checksum
Byte 10	End of array

IV. EXPERIMENT AND RESULTS

A. *Word Recognition*

For word recognition and classification, the CNN-LSTM network is used as a means for training and testing the machine learning model. This classification is done by using the Keras library in Python. On the CNN side, 128 convolution layers were used, with a kernel size of 2x2 matrix and ReLU activation. And then, the kernel will be pooled using the 2D MaxPooling technique with a 2x2 matrix size.

As this work uses limited data on non-linear hidden layers of the deep neural network, the tendency for overfitting to occur is high [15]. So Dropout method is used to prevent overfitting to occur. The Dropout coefficient used is 0.25.

The LSTM layers in the model are engaged by defining CNN layers within the “TimeDistributed” function from Keras. After building the layer, the model will be completed by using the “Dense” function to build a Fully Connected Layer (FC Layer). The final result will be compiled using the Adadelta optimizer. After compilation, the model will be tested for 150 epochs using 80% audio samples from the data set for training and 20% for testing.

TABLE IV. TRAINING AND TESTING RESULT

Epoch	Training Accuracy	Test Accuracy
10	47,21%	38,06%
25	62,19%	41,12%
50	68,86%	44,19%
75	73,57%	45,91%
100	85,67%	57,56%
125	90,36%	61,33%
150	89,19%	66,94%

For the accuracy of training and testing from all emotion datasets, depicted in Fig. 3. The training process from 80% of data, peaked at 90,36%. And the testing process used 20% of the data and peaked at 67%.

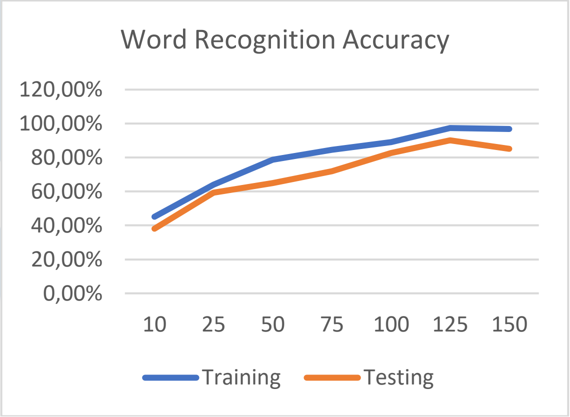


Fig. 6. Accuracy of Training and Testing

B. *Emotion Recognition*

Experiments in this research were conducted in 3 scenarios to fully validate the classification accuracy. All scenarios were conducted with 80:20 split data ratios. The parameter in CNN-LSTM networks used in the scenarios is ReLU activation CNN networks. For LSTM, the Softmax activation method is used for the final output. 150 epoch set for all scenarios with final compilation using the "Adadelta" optimizer.

1. *Scenario 1* - test the accuracy of the "happy" expression on the model.

2. *Scenario 2* - test the accuracy of the "sad" expression on the model.

3. *Scenario 3* - test the accuracy of the "angry" expression on the model.

Scenarios 1 to 3 will try to recognize which audio sample is classified as an emotional expression and to check the accuracy compared to the null model or undefined expression. The averaged result of "happy," "sad," and "angry" is shown in Table V. For the accuracy of training and testing from all emotion datasets, depicted in Fig. 7. The training process from 80% of data, peaked at 90,36%. And the testing process used 20% of the data and peaked at 67%

TABLE V. RESULT OF "HAPPY", "SAD", "ANGRY"

Emotion	Precision	Recall	f-1 Score
Happy	0,78	0,52	0,65
Sad	0,61	0,57	0,66
Angry	0,69	0,46	0,67

For the final testing scenario, all emotional expressions are processed together within the CNN-LSTM network. From this process, the output of this process can be depicted in a random confusion table, shown in Table VI.

TABLE VI. RANDOM SAMPLE CONFUSION RESULT

	Prediction			
Actual	Happy	Sad	Angry	Undefined
Happy	73	5	7	15
Sad	3	65	18	14
Angry	9	11	67	13

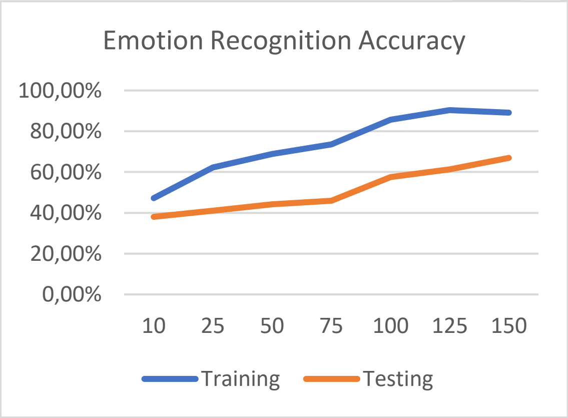


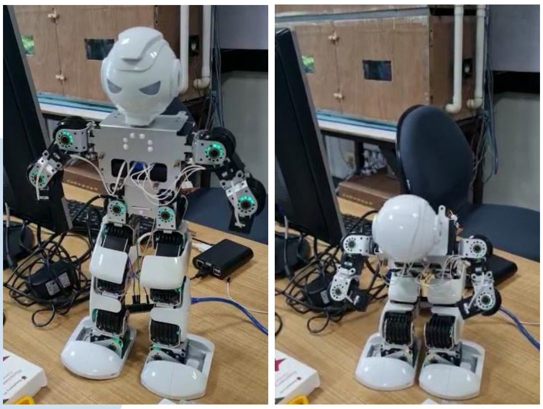
Fig. 7. Accuracy of Training and Testing

C. Robot Movement

The movement of the robot will depend on the classification results that have been obtained. The classification results that have been successfully obtained from the Raspberry Pi will be sent to the microcontroller. The robot will be driven by using serial communication between the microcontroller and the robot's servo motor. The movement of the robot being tested is for 4 types of movement, namely "Tegak", "Duduk", "Kanan", and "Kiri". This type of movement was chosen because this type of movement allows it to be carried out without analyzing the balance of the robot's movement. The first test is carried out using the data used in the machine learning model as input. The first experiment result is shown in Table VII.

TABLE VII. PRE-RECORDED INPUT TEST RESULT

No	Pre-Recorded Input	Classification Result	Robot Movement
1	Duduk	Duduk	Duduk
2	Tegak	Tegak	Tegak
3	Duduk	Duduk	Duduk
4	Tegak	Tegak	Tegak
5	Kanan	Kanan	Kanan
6	Kanan	Kanan	Kanan
7	Kiri	Kiri	Kiri
8	Duduk	Duduk	Duduk
9	Tegak	Tegak	Tegak
10	Kanan	Kanan	Kanan



(a) (b)

Fig. 8. Robot Movement for (a) Tegak, (b) Duduk

And then for the final testing scenario, the input data for the robot uses a new voice input recorded using a webcam microphone. The user must speak within 10-20 cm of the webcam. This is because the webcam microphone was not good enough to capture the user's voice. The result of the final experiment is shown in Table VIII below.

TABLE VIII. REAL-TIME INPUT TEST RESULT

No	Pre-Recorded Input	Classification Result	Robot Movement
1	Duduk	Duduk	Duduk
2	Tegak	Tegak	Tegak
3	Duduk	Duduk	Duduk
4	Tegak	Tegak	Tegak
5	Kanan	Kanan	Kanan
6	Kanan	Kanan	Kanan
7	Kiri	Kiri	Kiri
8	Duduk	Duduk	Duduk
9	Tegak	Tegak	Tegak
10	Kanan	Kanan	Kanan

From Table VIII above, mostly the results of the model classification resulted in the word "Duduk". This causes the robot to do more "Duduk" movements even though the user gives the "Right" and "Left" commands to the robot. This could be caused by the model experiencing overfitting because the results of the input using pre-recorded data indicate the model can work well according to the input given.

V. Conclusion

Our preliminary study on word-level emotion recognition results shows that this model has an acceptable performance. And from the confusion matrix result shows that the model accuracy is around 65%. These results should be improved in future works by adding more data and also using and comparing different layers of CNN and LSTM to determine how deep the network should be used for emotion recognition based on a self-built database like this. For robot implementation, the pre-recorded scenario shows the model can satisfy the movement classification with 100% classification, but when the model is introduced with new input data, it fails miserably as most input is classified as “Duduk” with accuracy only 20% from 10 data. This result is strong evidence for showing this model has an overfitting tendency. This is the main issue that must be solved in future works.

Acknowledgment

This study is supported by Universitas Multimedia Nusantara under an internal research grant.

References

[1] J. Wang and Z. Han, “Research on Speech Emotion Recognition Technology based on Deep and Shallow Neural Network,” *2019 Chinese Control Conf.*, pp. 3555–3558, 2019.

[2] P. Tzirakis, J. Zhang, and W. Schuller, “End-to-End Speech Emotion Recognition Using Deep Neural Networks,” *IEEE Int. Conf. Acoust. Speech, Signal Process. 2018*, pp. 5089–5093, 2018.

[3] K. Tarunika, R. B. Pradeeba, and P. Aruna, “Applying Machine Learning Techniques for Speech Emotion Recognition,” *2018 9th Int. Conf. Comput. Commun. Netw. Technol. ICCCNT 2018*, pp. 1–5, 2018, doi: 10.1109/ICCCNT.2018.8494104.

[4] B. T. Atmaja and M. Akagi, “Speech Emotion Recognition Based on Speech Segment Using LSTM with Attention Model,” *2019 IEEE Int. Conf. Signals Syst.*, pp. 40–44, 2019, doi: 10.1109/icsigsys.2019.8811080.

[5] C. Wan and L. Liu, “Research of speech emotion recognition based on embedded system,” *ICCSE 2010 - 5th Int. Conf. Comput. Sci. Educ. Final Progr. B. Abstr.*, pp. 1129–1133, 2010, doi: 10.1109/ICCSE.2010.5593692.

[6] Y. J. Lee, C. Y. Park, and H. J. Choi, “Word-Level Emotion Embedding Based on Semi-Supervised Learning for Emotional Classification in Dialogue,” *2019 IEEE Int. Conf. Big Data Smart Comput. BigComp 2019 - Proc.*, pp. 1–4, 2019, doi: 10.1109/BIGCOMP.2019.8679196.

[7] S. Basu, J. Chakraborty, and M. Aftabuddin, “Emotion Recognition From Speech Using Convolutional Neural Network With Recurrent Neural Network Architecture,” *ICCES 2017 - 2nd Int. Conf. Commun. Electron. Syst.*, pp. 333–336, 2017.

[8] D. Palaz, M. Magimai-Doss, and R. Collobert, “Convolutional Neural Networks-Based Continuous Speech Recognition Using Raw Speech Signal,” *ICASSP, IEEE Int. Conf. Acoust. Speech Signal Process. - Proc.*, pp. 4295–4299, 2015, doi: 10.1109/ICASSP.2015.7178781.

[9] J. J. Lasiman and D. P. Lestari, “Speech Emotion Recognition for Indonesian Language Using Long Short-Term Memory,” *2018 Int. Conf. Comput. Control. Informatics its Appl. Recent Challenges Mach. Learn. Comput. Appl. IC3INA 2018 - Proceeding*, pp. 40–43, 2019, doi: 10.1109/IC3INA.2018.8629525.

[10] D. Guiming, W. Xia, W. Guangyan, Z. Yan, and L. Dan, “Speech recognition based on convolutional neural networks,” pp. 708–711, 2017, doi: 10.1109/siprocess.2016.7888355.

[11] N. B. Wunarso and Y. E. Soelistio, “Towards Indonesian speech-emotion automatic recognition (I-SpEAR),” *Proc. 2017 4th Int. Conf. New Media Stud. CONMEDIA 2017*, vol. 2018-Janua, pp. 98–101, 2018, doi: 10.1109/CONMEDIA.2017.8266038.

[12] S. Park, Y. Jeong, M. S. Kim, and H. S. Kim, “Linear prediction-based dereverberation with very deep convolutional neural networks for reverberant speech recognition,” *Int. Conf. Electron. Inf. Commun. ICEIC 2018*, vol. 2018-Janua, no. 1, pp. 1–2, 2018, doi: 10.23919/ELINFOCOM.2018.8330593.

[13] D. A. A. Tuasikal, M. B. Nugraha, E. Yudhatama, A. S. A. S. Muharom, and M. Pura, “Word Recognition For Color Classification Using Convolutional Neural Network,” in *Proceedings of 2019 5th International Conference on New Media Studies, CONMEDIA 2019*, Oct. 2019, pp. 228–231, doi: 10.1109/CONMEDIA46929.2019.8981852.

[14] A. I. S. M. Ayu and K. K. Karyono, “Audio detection (Audition): Android based sound detection application for hearing-impaired using AdaBoostM1 classifier with REPTree weaklearner,” in *2014 Asia-Pacific Conference on Computer Aided System Engineering (APCASE)*, Feb. 2014, pp. 136–140, doi: 10.1109/APCASE.2014.6924487.

[15] A. Singh, K. K. Srivastava, and H. Murugan, “Speech emotion recognition using convolutional neural network (CNN),” *Int. J. Psychosoc. Rehabil.*, vol. 24, no. 8, pp. 2408–2416, 2020, doi: 10.37200/IJPR/V24I8/PR280260.