Contents list avaliable at Directory of Open Access Journals (DOAJ)

JUTIN : Jurnal Teknik Industri Terintegrasi

Volume 9 Issue 1 2026, Page 1310-1320

ISSN: 2620-8962 (Online)

Journal Homepage: https://journal.universitaspahlawan.ac.id/index.php/jutin/index

Identification of Speech Recognition Using K-Nearest Neighbor Method

Abdullah Hanif[1]✉, Fara Triadi[1], Arsan Kumala Jaya[1], Subhan Hartanto[1], Azhar Basir[2]

[(1)]Politeknik Negeri Samarinda, Samarinda, Indonesia

[(2)]Universitas Muhammadiyah Brebes, Brebes, Indonesia

DOI: 10.31004/jutin.v9i1.56699

✉ Corresponding author:
[abdullahhanif@polnes.ac.id]

Article Info

Keywords:
Speech;
Voice Command;
MFCC;
K-Nearest Neighbor

Abstract

Speech is a part of the human that has unique characteristics so that it can be distinguished from one person with someone else. Speech delivered, has a variety of information so that in its application it can be used to carry out voice commands using speech. In signal processing, Mel Frequency Cepstrum Coefficient (MFCC) is a method used for feature extraction. In this study, MFCC is used as a feature extraction method using Matlab R2017a and K-Nearest Neighbor (KNN) software used to identify and classify voice commands spoken by the speaker using speech pattern patterns obtained from the MFCC. This study uses 10 training data for each voice command word consisting of open, close, message and gallery, and 5 test data for each voice command word. Voice data is used using different words and different speakers. This research yields an accuracy level of 60% in voice Buka, 60% in voice Tutup, 60% in voice Pesan and 65% in voice Galeri.

1. INTRODUCTION

Speech is one way to know and recognize someone's character. Humans can recognize someone through their voice, such as: the speaker's identity, speaking style, accent, emotions and the speaker's health condition. The rapid development of technology has led to the existence of technology in the field of human voice signal processing, namely the introduction of input devices in general, but can use voice commands (Agustini, 2007). The speech signal processing architecture can be seen in Fig. 1.

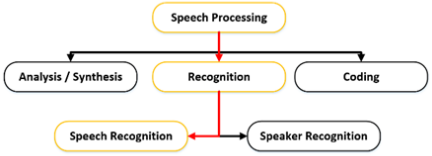

Fig. 1. Speech Recognition

Received 2 January 2026; Received in revised form 8 January 2026 year; Accepted 16 January 2026 Available online 20 January 2026 /

This research identifies the words spoken by the speaker so that they can be used to make it easier for humans to carry out commands using just their voice. Many research has been conducted discussing speech recognition for voice commands, including voice commands to open applications installed on the computer (Andriana, 2013), voice commands to control household electronic equipment (Masykur & Prasetiyowati, 2016), Speech recognition by giving voice commands to open or close automatic doors (Ariyanti, Adi, & Purbawanto, 2018), Control the lights on or off using sound sensors (Putra, Akbar, & Setyawan, 2018), as a security system on the user's smartphone (Sanjaya & Salleh, 2014), implementation of voice recognition using Nepali Language (Rai & Rai, 2014) and as well as speech recognition used to control the movement of the robot arm (Ronando & Irawan, 2012).

In the feature extraction process, there are several parameters used, such as duration (time), spectral and pitch as in the LPC or MFCC feature extraction methods (Gustina, Fadlil, & Umar, 2017). Feature extraction aims to obtain characteristic patterns from each sound data, so that it can facilitate the classification process in categorizing classes (Sinwar & Kaushik, 2014).

2. METHODS

This research to identify the words spoken by the speaker using the Mel Frequency Cepstrum Coefficient (MFCC) feature extraction method (Helmiyah, Riadi, Umar, & Hanif, 2019) and voice command word classification using the K-Nearest Neighbor (K-NN) method (Ronal & Yuliana, 2025). The stages of this research can be seen in Fig. 2.

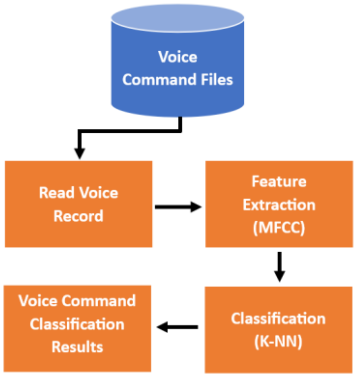

Fig. 2. Research Stages

2.1 Voice Command Files

The voice command file is a storage of voice command data that has been recorded and obtained from the speaker using Matlab R2017a software. Voice data was obtained from four speakers who said four words, namely "Buka", "Tutup", "Pesan", "Galeri". The sound data obtained is divided into two categories of data, namely training data and test data. Training data is used to obtain the characteristics of each spoken word, while test data is used for the word classification testing process.

2.2 Read Audio Record

The data used in this research was recorded from the speaker saying the specified words. Voice command recordings are in the form of voice files with the extension *.wav. Initial processing is carried out so that the Matlab Application can read recorded audio files for the signal conversion process into matrix-shaped values.

2.3 Feature Extraction

The feature extraction used in this research uses the Mel Frequency Cepstrum Coefficient (MFCC) method. In general, the stages in the MFCC feature extraction process can be seen in Fig. 3.

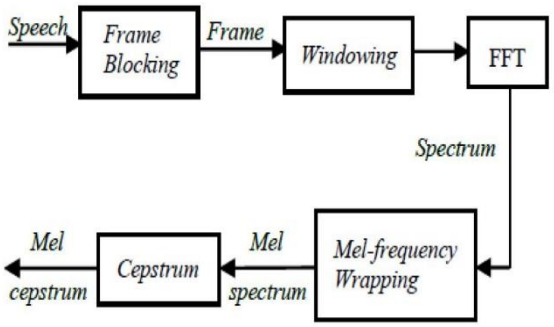

Fig. 3. An Overview of the MFCC Operation

Based on Fig 3, the explanation of the MFCC block diagram is as follows:
Pre Emphasis*:* The initial stage of pre-emphasis filter in the sound signal processing process is required after the processing of sample data is obtained. The purpose of this filtering process is to obtain a better spectral shape of the sound signal frequency. Where the spectral shape which has a high value for a small area will tend to decrease sharply in areas with frequencies >2000 Hz. Pre-emphasis filtering is obtained by input or output relationships in the time domain, as shown in equation (1):

$$y(n)=x(n)-ax(n-1) \quad (1)$$

Where alpha (a) is a pre-emphasis filter determination, with a value of 0.9 < a < 1.0.

Frame Blocking**:** In the frame blocking process, the sound signal that has been obtained will be segmented into several overlapping frames. This process will minimize signal disruption or loss (deletion). Where this process will continue until all existing signals have entered the frame.

Windowing**:** The sound signal that has been processed in the previous process is read every frame and in each frame a windowing process will be carried out using a certain window function. The windowing process is used to reduce unsustainable signals at the beginning and end of each frame. The output resulting from the windowing process is a signal, with an equation in the form of equation (2):

$$y1(n) = x1(n)w(n),\ 0 \le n \le N-1 \quad (2)$$

Where w(n), uses the Hamming window function, so that the equation becomes:

$$w(n) = 0.54-0.46.\ cos\ (2\pi n\ N-1),\ 0 \le n \le N-1 \quad (3)$$

Fast Fourier Transform (FFT)**:** FFT is used to convert each frame with a value of N data samples from the time domain into frequency domain form, as in equation (4).

$$\boldsymbol{Xn = \Sigma\ xk\ N-1\ k=0\ e-2\pi jkn/N} \quad (4)$$

Where n = 0, 1, 2, …, $N-1$ and $j = \sqrt{-1}$.

Mel-Frequency Wrapping*:* Human hearing perception of an audible sound frequency cannot be measured on a linear scale. For each sound tone which is the actual frequency, f, which has the unit form Hertz (Hz), "mel" is a pitch in the sound that can be measured on a scale. The mel-frequency scale is a low frequency with a size of <1000 Hz which is linear and a high frequency with a size of >1000 Hz which is logarithmic. The relationship between the Mel scale and frequency in Hz can be shown in equation (5).

$$Fmel = \{2595*log10\ (1 + FHZ\ 700),\ FHZ > 1000\ FHZ,\ FHZ < 1000 \quad (5)$$

The mel-frequency wrapping process for sound signals in the frequency domain uses equation (6).

$$Xi = log10(\Sigma|X(k)|\ N-1k=0\ Hi(k)) \quad (6)$$

2.4 Voice Command Classification

The results of the MFCC feature extraction process are used as training data in recognizing words spoken by speakers. The test data used in this research is 5 voice data for each voice command word. The results of the next test will be classified using the K-Nearest Neighbor method. In the K-NN method, to calculate the closest distance value between sound data to be classified using the Euclidean Distance Formula which can be seen in equation (7).

$$d = \sqrt{(x_2} - x_1)^2 + (y_2 - y_1)^2 \quad (7)$$

Description:
(x_1, y_1) = variable value of the new object being initialized
(x_2, y_2) = variable value of each neighbor in 2 variables

In this study there are more than 2 variables to calculate distance using the Euclidean Distance formula [12], can be seen in equation 8.

$$d = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \quad (8)$$

Description:
d = Euclidean Disatnce
x = The variable value of each neighbor
y = The value of the new object variable
i = Value variable
n = variable amount

3. RESULT AND DISCUSSION

3.1 Read Voice Record

The initial processing stage is changing the voice data recording file into matrix form using 10 voice data for training, which consists of 4 different words, namely "Buka", "Tutup", "Pesan", "Galeri".

```
% Read speech samples, sampling rate and
precision from file
[ speech, fs] = audioread(wav_file);
```

Fig. 4. Read Speech Samples Code

The source code above converts voice command recording data, the results of voice command recording data can be seen in Fig 5-8.

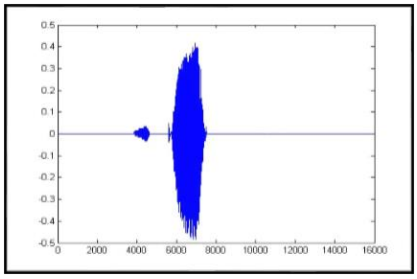

Fig. 5. Speaker 1 Command Record Signal Buka

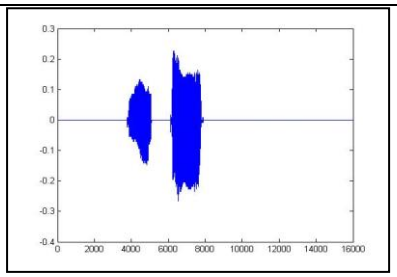

Fig. 6. Speaker 1 Command Record Signal Tutup

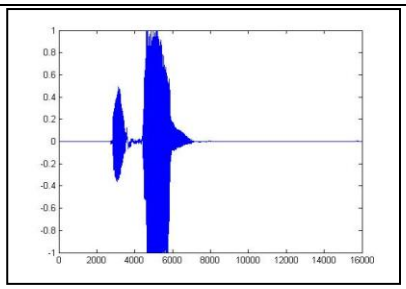

Fig. 7. Speaker 1 Command Record Signal Pesan

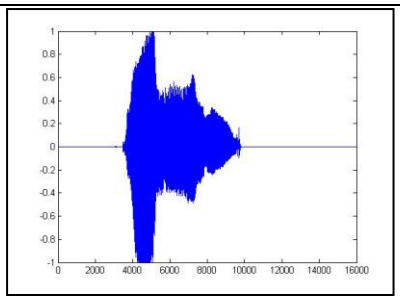

Fig. 8. Speaker 1 Command Record Signal Galeri

3.2 Feature Extraction Voice Result

The feature extraction process for training data uses 10 voice recording data, while for testing data uses 5 voice recording data. The feature extraction process using the MFCC method is carried out with the built-in Matlab function. The source code for the MFCC feature extraction function can be seen in Fig. 9.

```
%% FEATURE EXTRACTION

% Preemphasis filtering (see Eq. (5.1) on p.73 of [1])
speech = filter( [1 -alpha], 1, speech ); % fvtool( [1 -alpha], 1 );

% Framing and windowing (frames as columns)
frames = vec2frames( speech, Nw, Ns, 'cols', window, false );

% Magnitude spectrum computation (as column vectors)
MAG = abs( fft(frames,nfft,1) );

% Triangular filterbank with uniformly spaced filters on mel scale
H = trifbank( M, K, R, fs, hz2mel, mel2hz ); % size of H is M x K

% Filterbank application to unique part of the magnitude spectrum
FBE = H * MAG(1:K,:); % FBE( FBE<1.0 ) = 1.0; % apply mel floor
% DCT matrix computation
DCT = dctm( N, M );

% Conversion of logFBEs to cepstral coefficients through DCT
CC =  DCT * log( FBE );

% Cepstral lifter computation
lifter = ceplifter( N, L );

% Cepstral liftering gives liftered cepstral coefficients
CC = diag( lifter ) * CC; % ~ HTK's MFCCs
```

Fig. 9. Matlab MFCC Function

The results of feature extraction from the training data for each voice recording data can be seen in Fig.10.

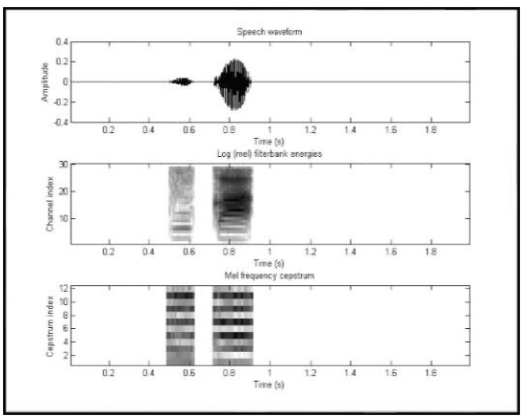

Fig. 10. Feature Extraction Voice Result

The results of the feature extraction process using Matlab R2017a produce values in the form of a 13x199 matrix, then the matrix values will be searched for the mean to form a 13x1 matrix as shown in Table 1.

Table 1. Speaker 1 Command Record Buka Value

Voice Data Characteristic	Value
MFCC1	43.400
MFCC2	-7.750
MFCC3	-14.588
MFCC4	5.400
MFCC5	-7.452
MFCC6	13.228
MFCC7	-17.847
MFCC8	13.893
MFCC9	-21.124
MFCC10	10.096
MFCC11	-11.487
MFCC12	18.729
MFCC13	-21.215

Table 1 shows the mfcc value data that has been obtained in the feature extraction process, because there are large differences in the values of each feature value, then the data normalization process is carried out using Z-Core. Normalization results using Z-Core can be seen in Table 2.

Table 2. Speaker 1 Normalized Buka Value

Voice Data Characteristic	Value
MFCC1	2.251
MFCC2	-0.418
MFCC3	-0.774
MFCC4	0.269
MFCC5	-0.402
MFCC6	0.677
MFCC7	-0.944
MFCC8	0.712
MFCC9	-1.115
MFCC10	0.514
MFCC11	-0.613
MFCC12	-0.944
MFCC13	0.712

After normalization is carried out in the feature extraction process, the next step is to calculate all normalized values for the 4 speakers with the command words open, close, message and gallery. The normalized values from all sound data tests for open words can be seen in Table 3.

Table 3. All Normalized Buka Value

Voice Data	Speaker			
	Speaker 1	Speaker 2	Speaker 3	Speaker 4
MFCC1	2.198	2.286	2.327	2.356
MFCC2	-0.498	-0.452	-0.378	-0.517
MFCC3	-0.674	-0.442	-0.584	-0.560
MFCC4	0.167	0.323	0.292	0.240
MFCC5	-0.670	-0.687	-0.739	-0.696
MFCC6	0.694	0.642	0.598	0.665
MFCC7	-0.939	-0.923	-0.889	-0.782
MFCC8	0.696	0.797	0.623	0.754
MFCC9	-1.034	-0.936	-0.964	-1.024
MFCC10	0.790	0.653	0.806	0.543
MFCC11	-0.739	-1.048	-0.979	-0.886
MFCC12	0.891	0.632	0.676	0.733

Voice Data	Speaker			
	Speaker 1	Speaker 2	Speaker 3	Speaker 4
MFCC13	-0.883	-0.846	-0.789	-0.825

The normalized cri extraction results are then made into a plot to display the characteristics of each sound word. The form of the feature extraction pattern for the word open that has been successfully obtained can be seen in Fig.11.

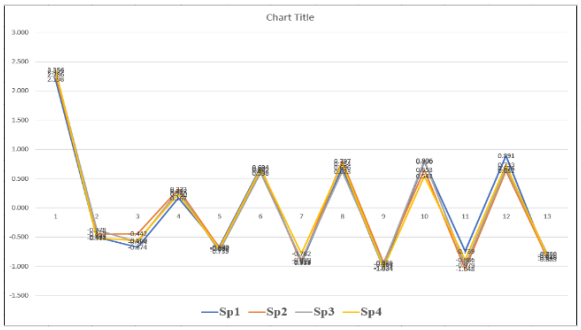

Fig. 11. Characteristic Pattern Command Buka

The differences in the characteristic patterns of each voice that have been tested can be seen in Fig. 12.

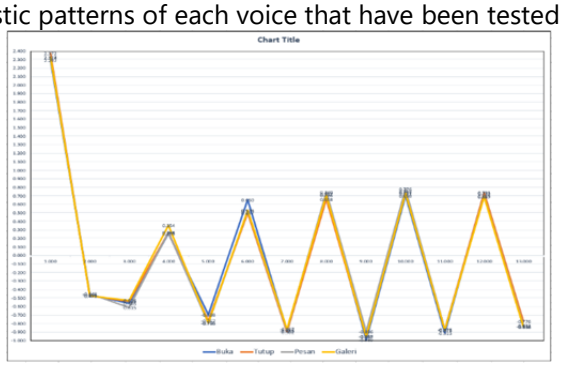

Fig. 12. Differences of Voice Patterns

3.3 Classification Result

The feature extraction process for training data uses 10 voice recording data, while for testing data uses 5 voice recording data. The feature extraction process using the MFCC method is carried out with the built-in Matlab function. The source code for the MFCC feature extraction function can be seen in Figure 8.

The classification result is the process of determining which voice recording data falls into the word class open, close, message or gallery. Class determination is obtained from the classification process using the K-Nearest Neighbor (K-NN) method. The steps required to obtain classification results are as follows.

Determining of K Parameter: The initial stage of the classification process using the K-NN method is determining the K parameter, the K parameter used is 4. Where K is the number of closest values.

Calculating the Closest Distance to New Data: The test data carried out for the classification process is carried out on Speaker 1 data Buka, like use Normalization Result with Z-core in Table 2. The results of calculating the closest distance of the Buka words spoken by 4 speakers can be seen in Table 4.

Table 4. Euclidean Result Voice Buka From 4 Spoken

Data	Speaker 1	Speaker 2	Speaker 3	Speaker 4
1	0.747	0.906	0.986	0.825
2	0.919	0.635	0.728	0.795
3	1.075	0.623	0.671	1.061
4	1.121	0.897	0.837	0.813
5	1.017	0.516	1.003	0.948
6	1.237	1.228	1.063	1.089
7	0.968	0.503	1.038	0.785
8	0.699	0.983	1.205	0.568

Data	Speaker 1	Speaker 2	Speaker 3	Speaker 4
9	1.013	1.126	0.971	0.852
10	0.722	0.940	0.977	0.635

Next, sort the Euclidean data that has been obtained from all speakers, sorting is done from the smallest value to the largest value. The results of sorting in ascending order can be seen in Table 5.

Table 5. Euclidean Sorting Result Data

Data Speaker	Euclidean Distance	Smallest Distance Ranking
Speaker 1	0.503	1
Speaker 1	0.516	2
Speaker 4	0.568	3
Speaker 1	0.623	4
Speaker 4	0.635	5
Speaker 1	0.635	6
Speaker 3	0.671	7
Speaker 2	0.699	8
Speaker 2	0.722	9
Speaker 3	0.728	10
Speaker 2	0.747	11
Speaker 4	0.785	12
Speaker 4	0.795	13
Speaker 4	0.813	14
Speaker 4	0.825	15
Speaker 3	0.837	16
Speaker 4	0.852	17
Speaker 1	0.897	18
Speaker 1	0.906	19
Speaker 2	0.919	20

The final classification process using K-NN is to categorize the number of closest values that have been determined in the first stage, where K= 4. The results of the 4 data that have been determined with the closest distance can be seen in Table 6.

Table 6. Euclidean Sorting Final Result Data

Data Speaker	Euclidean Distance	Smallest Distance Ranking
Speaker 1	0.503	1
Speaker 1	0.516	2
Speaker 4	0.568	3
Speaker 1	0.623	4

The test results using Buka Speaker 1 data have succeeded in identifying the voice class Buka. The K-NN concept tested in this research can be seen in Fig. 13.

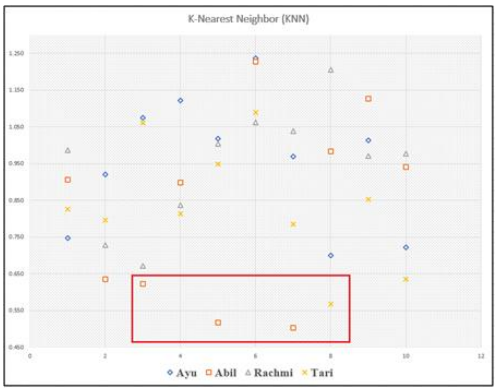

Fig. 13. Concept Classification K-NN Result

The results of testing data in the classification process for 20 sound data can be seen in Table 7.

Table 7. Number of Closest Distance Values

Data	Voice Command			
	Buka	Tutup	Pesan	Galeri
Speaker 1	3	2	3	3
Speaker 2	4	3	3	4
Speaker 3	3	4	2	3
Speaker 4	2	3	4	3

The accuracy of the test results for each word can be seen in Fig. 14.

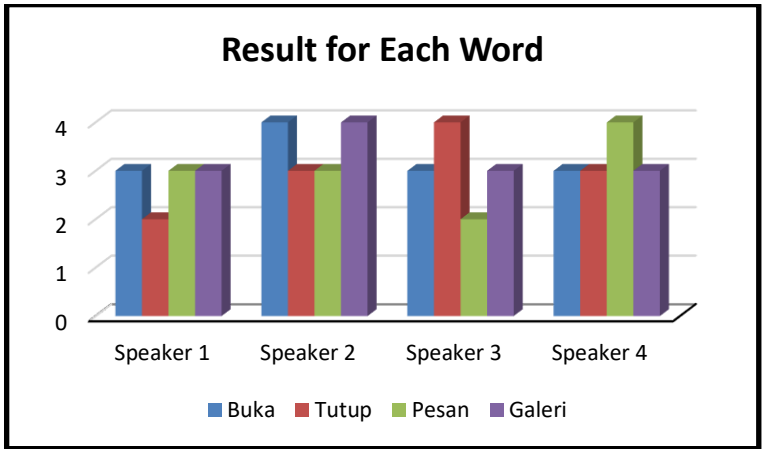

Fig. 14. Accuracy Graphic Voice

4. CONCLUSION

Speech recognition uses the MFCC method for the feature extraction process and the K-NN method for the classification process to be able to identify the words spoken by the speaker. This research resulted in accuracy in the voice of Buka 60%, the voice of Tutup 60%, the voice of Pesan 60% and the voice of Galeri 65%.

The identification process in this research did not obtain high accuracy results, because the voice data recorded by the user was limited to using Matlab R2017a software, and could not use other recording devices. Including the influence of the surrounding environmental conditions when recording voice data which is not good, thus affecting the results of the recorded data and during the speech recognition identification process.

5. REFERENCES

Agustini, K. (2007). Biometrik Suara dengan Transformasi Wavelet berbasis Othogonal Daubenchies. *Gematek Jurnal Teknik Komputer*, 50-56.

Andriana, A. D. (2013). Perangkat Lunak untuk Membuka Aplikasi pada Komputer dengan Perintah Suara menggunakan Metode Mel Frequency Cepstrum Coefficients. *Jurnal Ilmiah Komputer dan Informatika (KOMPUTA)*, 21-26.

Ariyanti, S., Adi, S. S., & Purbawanto, S. (2018). Sistem Buka Tutup Pintu Otomatis berbasis Suara Manusia. *Electronics, Informatics, and Vocational Education (ELINVO)*, 83-91.

Gustina, S., Fadlil, A., & Umar, R. (2017). Sistem Identifikasi Jamur Menggunakan Metode Ekstraksi. *Techno.COM*, 378-386.

Helmiyah, S., Riadi, I., Umar, R., & Hanif, A. (2019). Ekstraksi Fitur Pengenalan Emosi berdasarkan Ucapan menggunakan Linear Predictor Ceptral Coefficient dan Mel Frequency Cepstrum Coefficients. *Jurnal Mobile and Forensics (MF)*, 48-56.

Masykur, F., & Prasetiyowati, F. (2016). Aplikasi Rumah Pintar (Smart Home) Pengendali Peralatan Elektronik Rumah Tangga berbasis Web. *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK)*, 51-58.

Putra, M. K., Akbar, S. R., & Setyawan, G. E. (2018). Perancangan Sistem Keamanan pada Smart Home menggunakan Voice Command dengan Konektivitas Bluetooth. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 7417-7426.

Rai, N., & Rai, B. (2014). An ANN Based Mobile Robot Control Through Voice Command Recognition Using Nepali Language. *International Journal of Applied Control, Electrical and Electronics Engineering (IJACEEE)*, 13-22.

Ronal, & Yuliana. (2025). Penerapan Algoritma K-Nearest Neighbor (KNN) dalam Penerjemahan Bahasa Isyarat bagi Penyandang Disabilitas Tunarungu. *Jurnal Pusat Akses Kajian Teknologi Artificial Intelligence (Jurnal Pustaka AI)*, 30-34.

Ronando, E., & Irawan, M. I. (2012). Pengenalan Ucapan Kata Sebagai Pengendali Gerakan Robot Lengan Secara Real-Time dengan Metode Linear Predictive Coding – Neuro Fuzzy. *Jurnal Sains dan Seni ITS*, 51-56.

Sanjaya, M., & Salleh, Z. (2014). Implementasi Pengenalan Pola Suara Menggunakan Mel-Frequency Cepstrum Coefficients(MFCC) dan Adaptive Neuro-Fuzzy Inferense System(ANFIS) sebagai Kontrol Lampu Otomatis. *Al-Hazen Jurnal of Physics*, 43-54.

Sinwar, D., & Kaushik, R. (2014). Study of Euclidean and Manhattan Distance Metrics using Simple K-Means Clustering. *International Journal for Research in Applied Science and Engineering Technology*, 270-274.