

Implementasi Voice Recognition Pada Aplikasi Alarm Beker Berbasis Android Menggunakan Algoritma MFCC Dan DTW

[Voice Recognition Implementation In Android-Based Beker Alarm Applications Using MFCC And DTW Algorithms]

Nursyahid¹, Bulkis Kanata², Dwi Ratnasari³

^{1,3}Jurusan Teknik Elektro Universitas Mataram

¹nursyahidke@gmail.com, ²uqikanata@unram.ac.id

ABSTRAK

Alarm adalah aplikasi bawaan yang ada pada smartphone namun memiliki keterbatasan fitur, sehingga banyak pengguna yang memilih untuk memasang aplikasi alarm yang bukan bawaan smartphone. Keterbatasan pada alarm yang sudah ada salah satunya adalah kurang fitur yang bervariasi untuk menonaktifkan alarm ketika berbunyi, sehingga penulis membuat aplikasi alarm yang dapat mendukung fitur voice recognition untuk menonaktifkan alarm. Pada penelitian ini penulis menggunakan algoritma Mel Frequency Cepstral Coefficients (MFCC) dan Dynamic Time Warping (DTW) untuk mengenal suara masukan dari pengguna. Algoritma MFCC berfungsi untuk mengekstraksi fitur suara yang kemudian dicocokkan menggunakan algoritma DTW. Berdasarkan hasil pengujian model menggunakan confusion matrix didapatkan nilai rata-rata untuk accuracy, precision, recall, FPR, f1-score masing-masing sebesar 94%, 86%, 85%, 3.5%, dan 85% dari 90 data uji. Pengujian menggunakan blackbox testing dengan tingkat keberhasilan 100%, dan pengujian fungsi voice recognition untuk mematikan alarm didapat tingkat keberhasilan sebesar 89,33% dan ralat 10,66% dari 150 data uji.

Kata kunci : alarm, voice recognition, MFCC, DTW

ABSTRACT

Alarm is a default application on smartphones but has limited features, so many users choose to install an alarm application that is not built into a smartphone. One of the limitations of existing alarms is the lack of varied features to deactivate the alarm when it sounds, so the authors created an alarm application that can support the voice recognition feature to deactivate the alarm. In this study the authors used the Mel Frequency Cepstral Coefficients (MFCC) and Dynamic Time Warping (DTW) algorithms to recognize user input sound. The MFCC algorithm functions to extract sound features which are then matched using the DTW algorithm. Based on the results of testing the model using the confusion matrix, the average values for accuracy, precision, recall, FPR, f1-score were 94%, 86%, 85%, 3.5%, and 85% of the 90 test data, respectively. Testing using blackbox testing with a success rate of 100%, and testing the voice recognition function to turn off the alarm obtained a success rate of 89.33% and an error of 10.66% from 150 test data.

Keywords: alarm, voice recognition, MFCC, DTW

PENDAHULUAN

Perkembangan teknologi informasi yang semakin pesat membuat perangkat smartphone dapat mendukung berbagai aktivitas untuk dilakukan. Selain itu fitur yang ada pada *smartphone* juga semakin beragam dan canggih. Alarm adalah aplikasi bawaan yang ada pada *smartphone* namun memiliki keterbatasan fitur, sehingga banyak pengguna

yang memilih untuk memasang aplikasi alarm yang bukan bawaan smartphone. Keterbatasan pada alarm yang sudah ada salah satunya adalah kurang fitur yang bervariasi untuk menonaktifkan alarm ketika berbunyi, sehingga penulis melakukan penelitian untuk membuat aplikasi alarm beker berbasis Android dengan fitur unggulan menggunakan *voice recognition* untuk mematikan alarm.

DASAR TEORI

Voice Recognition

Pengenalan ucapan atau pengenalan suara dalam istilah bahasa Inggrisnya, *voice recognition* adalah suatu pengembangan teknik dan sistem yang memungkinkan komputer untuk menerima masukan berupa kata yang diucapkan. Teknologi ini memungkinkan suatu perangkat untuk mengenali dan memahami kata-kata yang diucapkan dengan cara digitalisasi kata dan mencocokkan sinyal digital tersebut dengan suatu pola tertentu yang tersimpan dalam suatu perangkat. Kata-kata yang diucapkan diubah bentuknya menjadi sinyal digital dengan cara mengubah gelombang suara menjadi sekumpulan angka yang kemudian disesuaikan dengan kode-kode tertentu untuk mengidentifikasi kata-kata tersebut. Hasil dari identifikasi kata yang diucapkan dapat ditampilkan dalam bentuk tulisan atau dapat dibaca oleh perangkat teknologi sebagai sebuah perintah untuk melakukan suatu pekerjaan, misalnya penekanan tombol pada telepon genggam yang dilakukan secara otomatis dengan perintah suara. Alat pengenal ucapan/suara, yang sering disebut dengan *voice recognizer*, membutuhkan sampel kata sebenarnya yang diucapkan dari pengguna. Sampel kata akan didigitalisasi, disimpan ke dalam komputer, dan kemudian digunakan sebagai basis data dalam mencocokkan kata yang diucapkan selanjutnya. Sebagian besar alat pengenal ucapan sifatnya masih tergantung kepada pembicara. Alat yang dibuat pun hanya dapat mengenali kata yang diucapkan dari satu atau dua orang saja (Ritonga, dkk. 2019).

Alarm Beker

Alarm secara umum dapat didefinisikan sebagai bunyi peringatan atau pemberitahuan. Beker adalah jam dengan perangkat yang dapat dibuat terdengar pada saat ditetapkan di awal yang digunakan untuk membangunkan seseorang (Banyumanis dan Fitriana 2017).

Algoritma MFCC dan DTW

MFCC (*Mel-Frequency Cepstral Coefficients*) adalah sebuah teknik

pemrosesan sinyal untuk ekstraksi fitur suara yang paling populer digunakan dalam aplikasi pengenalan suara. Teknik ini banyak digunakan untuk mengubah sinyal suara menjadi vektor fitur yang kemudian dapat digunakan sebagai input untuk model pengenalan suara seperti pengenalan ucapan dan deteksi emosi.

Algoritma MFCC terdiri dari beberapa tahapan yang melibatkan pengubahan sinyal suara dari domain waktu menjadi domain frekuensi dan kemudian menghitung koefisien *cepstral* pada sinyal suara. Tahapan-tahapan tersebut adalah:

1. *DC Removal* ditujukan untuk menghitung rata-rata dari data sampel suara dan mengurangi nilai setiap sampel suara dengan nilai rata-rata tersebut, tujuannya adalah membuang data-data yang tidak dibutuhkan di dalam proses suara latih. Proses *DC Removal* ditampilkan pada Persamaan 2.1

$$y[n] = x[n] - \bar{x}, 0 \leq n \leq N - 1 \quad (1)$$

Dimana:

$y[n]$: sampel sinyal hasil *DC Removal*

\bar{x} : nilai rata-rata sampel sinyal asli

$x[n]$: sampel sinyal asli

N : panjang sinyal

2. *Pra-Emphasis*: Tahap pertama dalam algoritma MFCC adalah *pra-emphasis*, yang bertujuan untuk mengurangi efek dari frekuensi rendah pada sinyal suara. *Pra-emphasis* dilakukan dengan menerapkan *filter high-pass* pada sinyal suara yang dapat meningkatkan kejelasan dan kualitas suara.

$$y[n] = s[n] - \alpha \cdot s[n - 1], 0,9 \leq \alpha \leq 1 \quad (2)$$

Dimana:

$y[n]$: sinyal hasil *pre-emphasis*

α : nilai alpha

$s[n]$: sinyal sebelum *pre-emphasis*

3. *FrameBlocking*: Setelah tahap *pre-emphasis*, sinyal suara dipisahkan menjadi beberapa *frame* yang lebih kecil. Setiap *frame* kemudian diberi label waktu dan diperlakukan secara terpisah dalam algoritma MFCC.

$$\text{jumlah frame} = \frac{l-N}{M} + 1 \quad (3)$$

Dimana:

l : *sample rate*

N : *sample point*(*sample rate* x waktu *framing* (s))

M : $N / 2$

4. *Windowing*: Pada tahap *windowing*, setiap *frame* diberi bobot dengan menggunakan fungsi *window* seperti *hammingwindow*. Fungsi *window* digunakan untuk mengurangi efek dari *overlapping* antar *frame*, sehingga setiap *frame* dapat dianggap sebagai data yang independen. Berikut persamaan *window* (4) dan *hammingwindow* (5).

$$x[n] = x_i[n]w[n] \quad (4)$$

Dimana:

$x[n]$: nilai sampel sinyal hasil *windowing*

$x_i[n]$: nilai sampel sinyal dari *frame* sinyal ke i

$w[n]$: fungsi *hamming window*

$$w[n] = 0,54 - 0,46 \cdot \cos(2\pi n / (N - 1)) \quad (5)$$

Dimana:

n : 1, 2, 3, ..., n

π : 3,14

N : panjang *frame*

5. *Fast Fourier Transform* (FFT): Tahap selanjutnya dalam algoritma MFCC adalah pengubahan sinyal suara dari domain waktu menjadi domain frekuensi menggunakan *Fast Fourier Transform* (FFT). FFT digunakan untuk menghitung amplitudo pada setiap frekuensi dalam setiap *frame* sinyal suara.

$$S(f) = \int_{-\infty}^{\infty} s(t)e^{-j2\pi ft} dt \quad (6)$$

Dimana:

$S(f)$: sinyal dalam domain frekuensi

f : frekuensi (hertz)

$s(t)$: sinyal dalam domain waktu

t : waktu (s)

$s(t)e^{-j2\pi ft}$: konstanta nilai suatu sinyal

π : 3.14

6. *Mel-Frequency Wrapping*: Pada tahap ini, amplitudo pada setiap frekuensi diubah ke dalam skala pita frekuensi *Mel*. Skala pita frekuensi *Mel* digunakan karena manusia cenderung lebih peka terhadap perbedaan frekuensi rendah daripada perbedaan frekuensi tinggi.

$$Mel f = 2595 * \log_{10} \frac{1 + \frac{f}{700}}{(S_i/2)} \quad (7)$$

Dimana:

$Mel f$: sinyal *mel* awal hasil FFT

S_i : sinyal $f: f_0 - f_n$

7. *Discrete Cosine Transform* (DCT) *Cepstrum* biasa digunakan untuk mendapatkan informasi dari suatu sinyal suara yang diucapkan oleh manusia. Pada tahap terakhir pada MFCC ini, spektrum

log mel akan dikonversi menjadi domain waktu menggunakan Discrete Cosine Transform (DCT) menggunakan persamaan berikut.

$$C_i = \sum_{i=1}^M \cos\left(\frac{j(i-1)}{2} \cdot \pi/M\right) \quad (8)$$

Dimana:

C_i : sinyal hasil proses DCT

M : jumlah *filter*

j : jumlah koefisien yang diharapkan

i : jumlah *warping*

X_i : hasil *mel-frequency* sinyal ke i

8. *Mel-Frequency Cepstral Coefficients* (MFCC): Pada tahap terakhir, sejumlah koefisien *cepstral* tertentu dipilih sebagai fitur MFCC. Jumlah koefisien yang dipilih bergantung pada aplikasi dan kebutuhan pengguna.

$$w[n] = \left\{ 1 + L/2 \sin\left(\frac{n\pi}{L}\right) \right\} \quad (9)$$

Dimana:

$w[n]$: hasil MFCC

n : indeks dari *Cepstral Coefficients*

L : jumlah *Cepstral Coefficients*

π : 3.14

Fitur MFCC yang dihasilkan dari algoritma ini telah terbukti efektif dalam aplikasi pengenalan suara dan menjadi standar *de facto* dalam ekstraksi fitur suara. Selain itu, teknik MFCC juga telah digunakan dalam berbagai aplikasi seperti pengenalan wicara, deteksi suara, dan pengenalan musik (Heriyanto, ddk. 2018).

DTW (*Dynamic Time Warping*) adalah sebuah teknik pencocokan waktu yang digunakan dalam pengolahan sinyal, khususnya dalam pemrosesan sinyal suara

dan pengenalan ucapan. DTW adalah sebuah algoritma yang dapat menentukan tingkat kesamaan antara dua sinyal dengan panjang waktu yang berbeda-beda. Teknik DTW didasarkan pada pemetaan jarak minimum antara dua rangkaian data, sehingga dapat digunakan untuk membandingkan sinyal suara yang tidak sama panjangnya. Algoritma DTW terdiri dari beberapa tahapan, yaitu:

1. Normalisasi: Tahap normalisasi dilakukan pada kedua rangkaian data (yang akan dibandingkan) dengan tujuan menghilangkan perbedaan level energi.
2. Matriks Jarak: Setelah tahap normalisasi, dibangun matriks jarak antara kedua rangkaian data, dengan setiap elemen matriks mewakili jarak antara setiap pasangan titik pada dua rangkaian data.
3. Matriks Akumulasi: Pada tahap ini, dibangun matriks akumulasi yang merepresentasikan jarak minimum untuk setiap pasangan titik pada kedua rangkaian data.
4. Jalur Optimal: Setelah matriks akumulasi terbentuk, dicari jalur optimal yang menghubungkan titik awal dan titik akhir pada kedua rangkaian data. Jalur ini merepresentasikan penyesuaian waktu yang diperlukan untuk menghubungkan kedua rangkaian data dengan jarak minimum.
5. Penentuan Tingkat Kesamaan: Akhirnya, tingkat kesamaan antara kedua rangkaian data dihitung berdasarkan jarak minimum dan jalur optimal yang ditemukan.

Teknik DTW telah digunakan dalam berbagai aplikasi seperti pengenalan ucapan, pengenalan gestur, dan pengenalan gerakan. Keuntungan utama dari DTW adalah kemampuannya untuk membandingkan sinyal suara dengan panjang waktu yang berbeda-beda, sehingga dapat digunakan dalam aplikasi di mana ukuran sinyal suara tidak selalu sama (Handoko dan Kasih 2018).

$$GD_{xy} = LD_{xy} +$$

$$\min(GD_{x-1 y-1}, GD_{x-1 y}, GD_{x y-1}) \quad (10)$$

Dimana:

GD_{xy} : Global Distance (overall distance)

LD_{xy} : Local Distance (Euclidean Distance)

Untuk mencari nilai GD_{xy} atau LD_{xy} dapat menggunakan rumus berikut ini:

$$Dist(x, y) = |x - y| = [((x_1 - y_1)^2 + (x_2 - y_2)^2 + \dots + (x_n - y_n)^2)^{1/2}] \quad (11)$$

Dimana:

$Dist(x, y)$: Jarak Euclidean (GD atau LD)

x : sampel sinyal indeks ke x

y : sampel sinyal indeks ke y

METODOLOGI PENELITIAN

Perancangan Sistem Voice Recognition

Pengumpulan Data

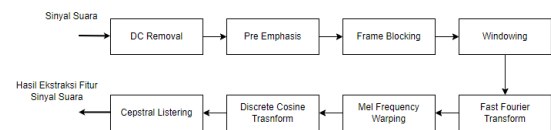
Dalam penelitian ini penulis melakukan proses pengumpulan data dengan teknik angket (kuisisioner) dimana orang yang dijadikan sumber data akan diberikan 5 kalimat motivasi yang akan dibaca sambil merekam suaranya. Setiap kalimat akan di baca 5 kali dengan masing-masing durasi 4 detik untuk setiap kalimat, sehingga setiap partisipan akan merekam suara sebanyak 25 kali. Dalam pengumpulan data ini partisipan sebanyak 6 orang termasuk penulis, hasil rekaman ini akan dijadikan sebagai data latih dalam proses menerapkan algoritma MFCC dan DTW dalam mengenali ucapan atau *voice recognition*.

Pelatihan Pengujian dan Evaluasi Model

Pada tahap ini akan dilakukan proses pelatihan, pengujian dan evaluasi model atau pola suara menggunakan algoritma MFCC dan DTW sehingga dapat diklasifikasikan oleh sistem untuk mematikan alarm.

Pelatihan Model

Pada tahap ini, model dilatih untuk melakukan klasifikasi rekaman suara untuk menentukan apakah rekaman suara merupakan kalimat motivasi pertama, kedua, ketiga, keempat atau kelima menggunakan data latih yang ada. Jumlah total data latih yang digunakan adalah 150 data dengan pembagian masing-masing kalimat motivasi terdiri dari 30 rekaman suara. Berikut dijelaskan tahapan pelatihan yang dilakukan menggunakan algoritma MFCC.



Gambar 1 Proses Mel Frequency Cepstral Coefficient

Pengujian Model

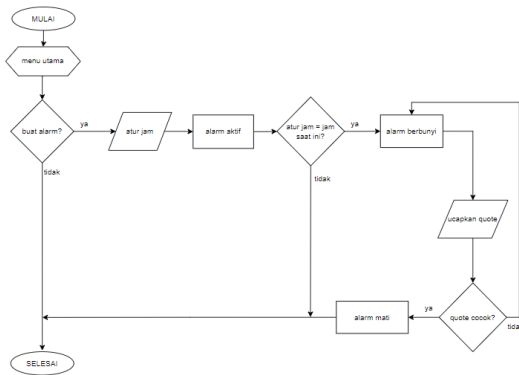
Setelah model selesai dilatih, model akan diberikan data yang belum pernah diolah sebelumnya untuk menguji seberapa baik performa model dalam mengenali pola suara dari masing-masing kalimat. Jumlah total data uji yang digunakan adalah 90 data dengan pembagian masing-masing kalimat terdiri dari 18 rekaman suara.

Evaluasi Model

Proses evaluasi model dilakukan untuk melihat seberapa baik model melakukan klasifikasi pada data yang belum pernah dilihat sebelumnya dan diukur berdasarkan metrik akurasi, precision, recall atau true positive rate, false positive rate, dan F1-score. Nilai-nilai tersebut dapat dihitung menggunakan persamaan 12 sampai 16 pada dasar teori.

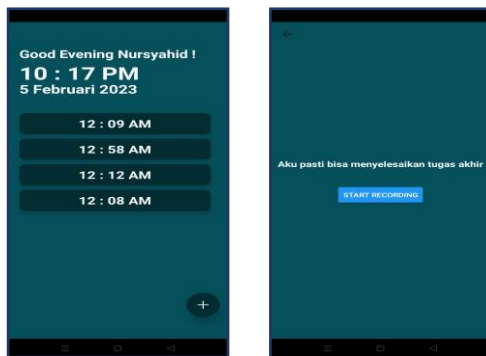
Perancangan Aplikasi

Flowchart



Gambar 2 Flochart aplikasi alarm beker

Desain UI/UX



Gambar 3 Desain UI/UX aplikasi alarm beker

HASIL DAN PEMBAHASAN

Hasil Preprocessing Data

Dataset rekaman suara yang sudah disimpan kemudian melalui beberapa proses untuk memperoleh pola yang diinginkan yaitu berada dalam satu format yang sama. Preprocessing pada data rekaman suara ini dilakukan menggunakan algoritma MFCC dan DTW. MFCC digunakan untuk mengekstraksi sinyal suara menjadi data array yang dapat diolah secara matematis oleh sistem melalui beberapa tahapan ekstraksi. DTW digunakan untuk melakukan pencocokan antara vektor uji dicocokkan dengan vektor suara latih dan didapatkan nilai minimum yang menunjukkan jenis suara dari masukan *user* terhadap aplikasi yang dibangun.

Proses Ekstraksi Sinyal Suara Menggunakan Algoritma MFCC

Algoritma MFCC terdiri dari beberapa tahapan yang melibatkan perubahan sinyal suara dari domain waktu menjadi domain frekuensi dan kemudian menghitung koefisien cepstral pada sinyal suara. Tahapan-tahapan tersebut adalah:

1. Suara dari *user* yang dijadikan sebagai suara latih didapatkan dari proses perekaman atau suara tersebut dipilih dari direktori. Suara tersebut memiliki bentuk (.wav) dengan frekuensi sample 16000 Hz, dari suara tersebut didapatkan nilai berupa *array* suara latih yang akan dilakukan tahap ekstraksi ciri dengan menggunakan MFCC. Contoh sample data *array* dari suara *inputuser* adalah sebagai berikut : [10, 11, 13, 14, 16, 18, 20, 9, 6, 4].
2. *DC Removal* ditujukan untuk menghitung rata-rata dari data sampel suara dan mengurangi nilai setiap sampel suara dengan nilai rata-rata tersebut, tujuannya adalah membuang data-data yang tidak dibutuhkan di dalam proses suara latih.
3. *Pre emphasize* dilakukan untuk mengurangi *noise* pada suara masukan, sehingga tingkat akurasi dari proses ekstraksi ciri dapat ditingkatkan. Berdasarkan *Frameblocking* adalah suatu proses untuk membagi sampel suara menjadi beberapa *frame* atau slot. Proses *windowing* adalah proses untuk mengurangi kesenjangan (discontinuitas) sinyal pada awal dan akhir bingkai setelah proses *frameblocking*.

Fast Fourier Transform (FFT) berfungsi untuk mengubah domain waktu menjadi domain frekuensi.

4. *Mel-Frequency Wrapping*: Pada tahap ini, amplitudo pada setiap frekuensi diubah ke dalam skala pita frekuensi *Mel*.
5. DCT adalah proses untuk menghitung *melspectrum* sehingga menghasilkan representasi yang baik dari spektral suara.
6. Proses *cepstralliftering* merupakan langkah terakhir dari proses utama MFCC berfungsi untuk meningkatkan kualitas pengenalan suara.

Proses Recognition Menggunakan Algoritma DTW

DTW (*Dynamic Time Warping*) adalah sebuah teknik pencocokan waktu yang digunakan dalam pengolahan sinyal, khususnya dalam pemrosesan sinyal suara dan pengenalan ucapan. Berdasarkan persamaan 9 maka dapat dilakukan pencocokan dua buah sinyal suara dengan mencari nilai local distance dan global distance sehingga didapatkan nilai threshold yang akan menjadi acuan dua buah sinyal suara memiliki kemiripan ciri.

Pengujian Sistem Pemrosesan Data

Untuk uji coba sistem digunakan data 90 suara dari 6 orang. Uji coba dilakukan dengan data training suara. Uji coba ini dilakukan dalam keadaan hening. Selanjutnya dilakukan

hitung nilai *accuracy*, *precision*, *recall*, *false positive rate*, dan *F1-score* menggunakan persamaan 12 sampai 16. Dalam mencari nilai-nilai tersebut kita memerlukan nilai *true positive*, *true negative*, *false positive* dan *false negative* yang di dapat dari proses pengujian sistem yang disajikan dalam *confusion matrix* berikut.

		Label Sebenarnya				
		Quote 1	Quote 2	Quote 3	Quote 4	Quote 5
Label Prediksi	Quote 1	15	1	1	0	1
	Quote 2	0	15	3	0	0
	Quote 3	0	1	16	1	0
	Quote 4	0	0	0	17	1
	Quote 5	1	0	1	2	14

Gambar 4 *confusion matrix* hasil prediksi model pada data uji

Tabel 1 Daftar nilai *accuracy*, *precision*, *recall*, *false positive rate*, dan *F1-score*

Class	Accuracy	Precision	Recall	FPR	F1-score
Quote 1	0,95	0,93	0,83	0,013	0,88
Quote 2	0,94	0,88	0,83	0,027	0,85
Quote 3	0,92	0,76	0,88	0,069	0,82
Quote 4	0,95	0,85	0,94	0,041	0,89
Quote 5	0,93	0,87	0,77	0,027	0,82
Average	0,94	0,86	0,85	0,035	0,85

KESIMPULAN DAN SARAN

Kesimpulan

Berdasarkan hasil pengujian model menggunakan *confusion matrix* didapatkan nilai rata-rata untuk *accuracy*, *precision*, *recall*, *FPR*, *f1-score* masing-masing sebesar 94%, 86%, 85%, 3.5%, dan 85% dari 90 data uji. Pengujian menggunakan *blackbox testing* dengan tingkat keberhasilan 100%, dan terakhir pengujian fungsi *voice recognition* untuk mematikan alarm didapat tingkat keberhasilan sebesar 89,33% dan ralat 10,66% dari 150 data uji. Maka dapat ditarik kesimpulan tujuan penelitian untuk membangun aplikasi alarm berkeryang mendukung fitur *voice recognition* berbasis Android yang dapat membantu meningkatkan fitur alarm pada pengguna smartphone, dinyatakan telah dicapai karena mampu memanfaatkan teknologi *voice recognition* dalam proses mematikan notifikasi alarm,

khususnya dalam proses menginput quote, sehingga dibutuhkan usaha dalam mematkan alarm yang dapat meningkatkan RAS (*Reticular Activating System*) untuk memperoleh kesadaran.

Saran

Aplikasi ini masih jauh dari sempurna, untuk menciptakan sebuah aplikasi yang baik tentu perlu dilakukan pengembangan baik dari sisi manfaat maupun dari sisi kerja sistem, berikut beberapa saran bagi yang ingin mengembangkan aplikasi yang mungkin dapat menambah nilai dari aplikasi nantinya:

1. Aplikasi alarm beker ini perlu untuk sepenuhnya menggunakan voicerecognition untuk memberikan perintah pada aplikasi seperti membuat alarm baru, edit dan menghapus alarm yang sudah dibuat.
2. Mengimplementasikan pelatihan voice recognition pada aplikasi alarm sehingga tidak perlu dilatih secara terpisah.
3. Menambahkan algoritma lainnya agar dapat meningkatkan tingkat akurasi voice recognition untuk menonaktifkan alarm.

DAFTAR RUJUKAN

- Aini, Nur, Desi Andreswari, and Funny Farady C. 2022. "Aplikasi Belajar Daring Bahasa Inggris Kelas VII SMP Menggunakan Teknologi *Voice Recognition* Berbasis Android (Studi Kasus: SMP Negeri 17 Bengkulu)." *Rekursif: Jurnal Informatika* 10(1): 81–95.
- Banyumanis, Ihya Ulumuddin &, and Devi Fitriana. 2017. "Aplikasi Alarm Beker Berbasis Android Untuk Mengacak Pertanyaan." *Jurnal Ilmu Teknik dan Komputer* 1(1): 1–10.
- Eisa, Ivani Resti, and Derisma. 2021. "Implementasi Voice Recognition Dan Sensor Ultrasonik Pada Televisi." *Chipset* 2(02): 1–6.
- Eisenman, Bonnie. 2015. "Learning React Native: Building Native Mobile Apps with JavaScript". Hyderabad: Tutorials Point India Private Limited.
- Händel, K. 1972. "Alkoholwirkung in Der Resorptionsphase." *Therapie der Gegenwart* 111(5): 756-757 passim.
- Handoko, Destian Tri, and Patmi Kasih. 2018. "Destian Tri Handoko Voice Recognition Untuk Sistem Keamanan PC Menggunakan Metode MFCC Dan DTW." *Generation Journal* 2(1): 57.
- Heriyanto, Heriyanto, Sri Hartati, and Agfianto Eko Putra. 2018. "Ekstraksi Ciri Mel Frequency Cepstral Coefficient (Mfcc) Dan Rerata Coefficient Untuk Pengecekan Bacaan Al-Qur'an." *Telematika* 15(2): 99.
- Pratama, Andre. 2017. "JavaScript Uncover – Panduan Belajar JavaScript Untuk Pemula." : 1–3.
- Putra, Muhamad M I et al. 2020. "Implementasi Speech Recognition Pada Aplikasi Pembelajaran Bahasa Inggris Untuk Anak." *Jurnal Teknik Informatika* 15(4): 247–56.
- Ritonga, Muhammad Rizki, Nurul Fadillah, and Liza Fitria. 2019. "Sistem Kendali Peralatan Elektronik Rumah Tangga Melalui Media Wireless Fidelity Menggunakan Voice Recognition Secara Real Time." *InfoTekJar (Jurnal Nasional Informatika dan Teknologi Jaringan)* 3(2): 1–7.
- Setyoningrum, Tri Yuli, Kabupaten Blora, and Jawa Tengah. 2022. "PENGEMBANGAN APLIKASI THERA UNTUK." 5(1): 104–16.
- Wahid, Aceng Abdul. 2020. "Analisis Metode Waterfall Untuk Pengembangan Sistem Informasi." *Jurnal Ilmu-ilmu Informatika dan Manajemen STMIK* (November): 1–5.