

# Pengenalan Suara Manusia Menggunakan Support Vector Classifier(SVC) Untuk Proses Otentikasi

Florentina Pramita Anindya<sup>1</sup>, Dyah Erny Herwindiati<sup>2</sup>, Novario Jaya Perdana<sup>3</sup>

<sup>1,2,3</sup>Program Studi Teknik Informatika, Fakultas Teknologi Informasi, Universitas Tarumanagara  
Jln. Letjen S. Parman No 1, Jakarta 11440, Indonesia

E-mail : <sup>1</sup>florentina.535180035@stu.ac.co.id, <sup>2</sup>dyahh@fti.untar.ac.id, <sup>3</sup>novariojp@fti.untar.ac.id

## Abstrak

Sistem pengenalan suara SEAUI merupakan salah satu aplikasi website untuk melakukan proses verifikasi suara sebagai salah satu teknik pengamanan tingkat lanjut dengan mengenali pemilik suara setelah melakukan proses login. Sistem ini dibuat untuk diterapkan pada aplikasi website. Sistem ini dibuat dengan menggunakan bahasa pemrograman Python dengan modul framework Flask dan basis data SQL Server Management Studio untuk penyimpanan data. Proses pengenalan suara pada sistem ini menggunakan metode Median Filter, metode ekstraksi fitur suara Mel Frequency Cepstral Coefficients dan metode klasifikasi Support Vector Classifier. Proses pengumpulan data dilakukan dengan teknik kuesioner dan didapatkan data suara dari 25 responden. Dalam tahap pengujian, sistem SEAUI sudah dapat melewati tes pengujian SQL Injection dengan 4 kali percobaan pengujian dan sistem dapat berjalan sesuai fungsionalitas. Data suara yang digunakan sebagai input adalah 1 data suara rekaman pada saat login dan kumpulan data suara yang terdapat dalam basis data. Akurasi tertinggi sistem SEAUI yang didapatkan adalah sebesar 91% untuk 60 kali percobaan pengujian dari 6 akun pengguna.

**Kata Kunci**—Support Vector Classifier, Otentikasi Suara, Speech Recognition

## Abstract

SEAUI voice identification system is a website application to perform a voice verification process as an advance security technique by recognize the owner of the voice after finish a login process. This system will apply in website application. Created with Python language and Flask framework and SQL Server Management Studio for database. Voice recognition is process with Median Filter method, Mel Frequency Cepstral Coefficients extraction method, and Support Vector Classifier for classification method. The data collected from 25 respondent who is filled the questionnaire. In testing stage, SEAUI passed SQL Injection test with 4 times test and complete running according to functionality. Training and testing data is 1 file recorded after submit in verification stage and combined with another audio file from database. Highest accuracy is 91% for 60 times testing from 6 account.

**Keywords**— Support Vector Classifier, Voice Authentication, Speech Recognition

## 1. PENDAHULUAN

Saat ini teknologi memberikan inovasi hebat untuk membantu manusia dalam melakukan beberapa pekerjaan secara bersamaan. Pengenalan ucapan manusia akan suatu aktivitas dapat langsung dieksekusi dengan cepat dengan fitur *voice assistance*. Tetapi untuk pengenalan otoritas suara masih belum banyak implementasi dan dalam proses pengembangan. Pengenalan otoritas

suara dapat menjadi salah satu teknik keamanan yang baru setelah keamanan dengan sidik jari, pengenalan wajah, pengenalan retina dan kata sandi. Keamanan kata sandi menjadi salah satu teknik keamanan yang sangat sering digunakan untuk syarat dalam mengakses suatu akun. Suara manusia dapat menjadi salah satu syarat untuk mengamankan suatu akun dengan menjadi tahap verifikasi suara.

Suara manusia yang didapat akan diolah menjadi angka dan dilakukan ekstraksi suara. *Mel Frequency Cepstral Coefficients* merupakan salah satu teknik yang sering digunakan untuk mengekstraksi ciri suara. Selanjutnya suara tersebut akan diklasifikasi untuk mengetahui kelas yang didapatkan dari suara tersebut. *Support Vector Classifier* merupakan salah satu teknik untuk mengklasifikasi kelas. SVC juga memiliki kernel-kernel yang dapat digunakan sesuai kasus perhitungan yang diolah. *Input* suara akan berupa *file* audio berekstensi WAV. Data suara tersebut diproses dengan bahasa pemrograman Python, dikarenakan pada Python terdapat modul-modul yang mendukung dalam pemrosesan suara.

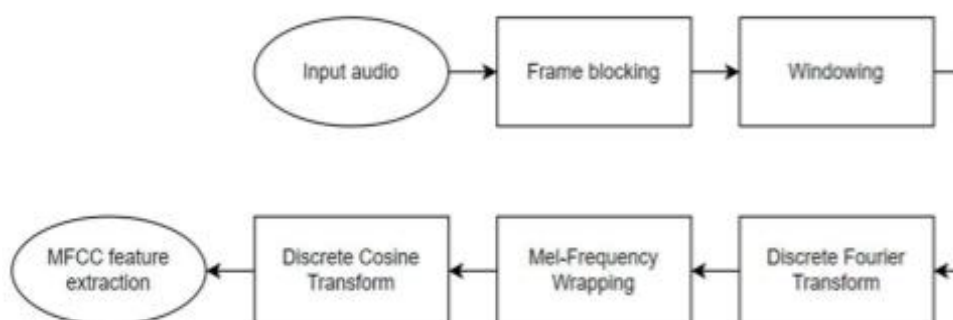
## 2. METODE PENELITIAN

### 2.1 Data

Data yang digunakan dalam penelitian ini didapatkan dari responden dengan menyebarkan angket. Terdapat 2 bagian dalam angket yaitu bagian data pribadi responden seperti nama, usia, email, dan lain-lain, serta bagian pengumpulan data suara responden. Masing-masing responden akan mengunggah 5 data suara dengan mengucapkan kalimat yang sudah ditetapkan pada halaman angket. Data yang didapat berupa data suara berekstensi WAV sebanyak 275 data suara dari 55 responden.

### 2.2 Algoritma

Data suara yang akan diproses akan dilakukan pra-proses sebelum di proses, yaitu dengan metode Median Filter. Median Filter merupakan teknik pemrosesan sinyal non linear yang berguna untuk penekanan noise[1]. Penggunaan Median Filter terhadap data suara dilakukan untuk meminimalisir gangguan yang terdapat dalam sinyal data suara tersebut. Dalam penelitian ini, suara yang sudah diminimalisir gangguannya akan dilakukan teknik ekstraksi ciri suara untuk mengonversi sinyal suara menjadi beberapa parameter[2]. Teknik ekstraksi ciri suara yang digunakan adalah teknik ekstraksi fitur suara *Mel Frequency Cepstral Coefficients* (MFCC) dengan matriks 12 indeks parameter sebagai hasil. Teknik ekstraksi ini cukup terkenal dan sering digunakan untuk mengolah dan menganalisis suara. Teknik ekstraksi suara MFCC ini mampu menangkap karakteristik suara yang sangat penting bagi pengenalan suara dan menghasilkan data yang minim tanpa menghilangkan informasi-informasi penting yang ada[2].



**Gambar 1** Alur *Mel-Frequency Cepstral Coefficients*

Dari Gambar 1 diatas, data *input* yaitu data suara akan masuk ke tahap *frame blocking*. *Spectrum frame* suara akan dihaluskan pada tahap *Windowing* setelah tahap *frame blocking*. Nilai suara tersebut akan diubah dari domain waktu menjadi domain frekuensi[3]. Perubahan nilai suara tersebut akan dilakukan pada tahap *Discrete Fourier Transform* (DFT). Pengembangan algoritma DFT disebut *Fast Fourier Transform* (FFT). Dalam perhitungan DFT, FFT dapat digunakan untuk mempercepat perhitungan. Hasil dari DFT akan dilakukan tahap *Mel Frequency Wrapping* yang berfungsi untuk mengubah frekuensi ke dalam bentuk mel [4]. Dan tahap *Discrete Cosine Transform* merupakan tahap untuk mengubah kembali nilai spektrum mel menjadi nilai dalam domain waktu. Hasil dari MFCC adalah nilai berupa *frame* dan koefisien *cepstral*.

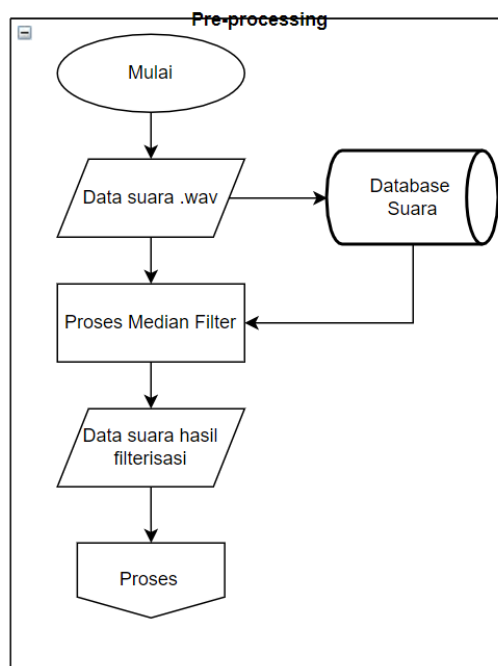
Setelah didapatkan koefisien hasil ekstraksi fitur suara MFCC, suara akan masuk ke tahap klasifikasi untuk menentukan pemilik dari suara yang diolah. Teknik klasifikasi yang digunakan adalah teknik *Support Vector Classifier* (SVC) yang dikembangkan oleh Vladimir N Vapnik dan Alexey Ya Chervonenkis pada tahun 1963. Dalam artikel “*Voice Identification Using Classification Algorithms*” yang ditulis oleh Orken Mamyrbayev, algoritma SVC mendapatkan akurasi sebesar 90% dalam analisis perbandingan beberapa algoritma untuk mengidentifikasi suara dengan *machine learning*[5]. SVC merupakan suatu teknik untuk menemukan *hyperplane* untuk memisahkan dua set kelas yang berbeda[6]. Terdapat kernel-kernel yang bisa digunakan untuk memaksimalkan margin dari *hyperplane* tersebut, seperti kernel linear, kernel sigmoid, kernel polinomial dan kernel fungsi basis radial. Pada penelitian ini, digunakan 2 kernel, yaitu kernel linear dan kernel polinomial. Berikut persamaan dari kernel linear dan kernel polinomial dapat dilihat pada persamaan (1) dan (2):

$$K(x_i, x_j) = x_i^T x_j \quad (1)$$

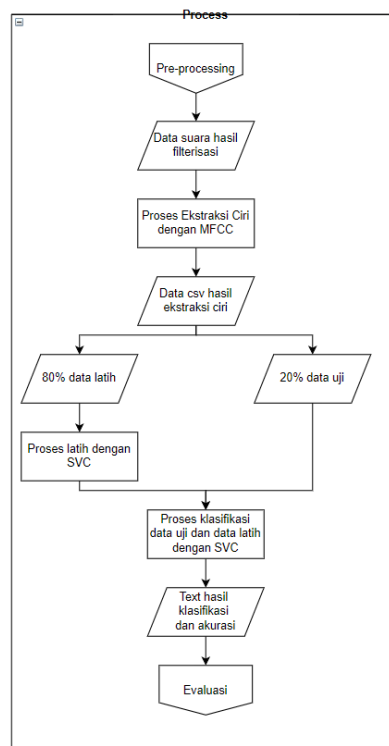
$$K(x_i, x_j) = (x_i^T x_j + r)^d \quad (2)$$

### 2.3 Rancangan

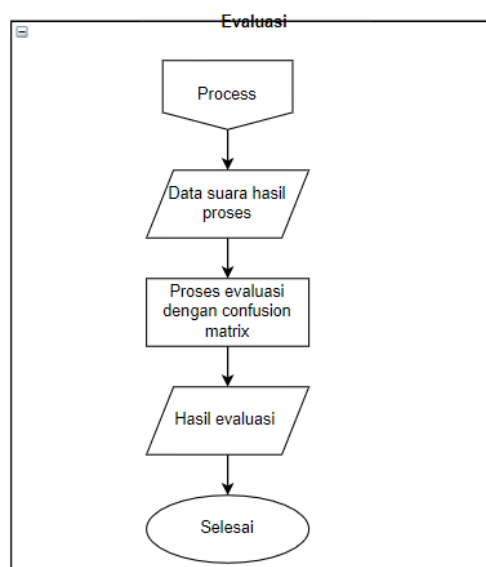
Terdapat 3 bagian alur yang akan digunakan, yaitu bagian pra-proses, bagian proses dan bagian evaluasi seperti pada Gambar 2, Gambar 3 dan Gambar 4 dibawah ini.



**Gambar 2** Flowchart pra-proses data suara



Gambar 3 Flowchart proses data suara



Gambar 4 Flowchart evaluasi proses data suara

Dari Gambar 2 diatas, data yang digunakan akan dikumpulkan dari penyebaran angket sesuai dengan yang telah dijelaskan pada subbab 2.1. Penelitian ini akan menggunakan data suara wav yang akan dilakukan pra-proses untuk meminimalisir gangguan dengan metode Median Filter. Setelah data di minimalisir gangguannya, selanjutnya data suara akan diekstraksi ciri dengan fitur *Mel Frequency Cepstral Coefficients* dan menghasilkan matriks 12 indeks koefisien MFCC.

Pada penelitian ini, kernel yang digunakan adalah kernel linear dan kernel polinomial untuk memaksimalkan nilai margin dari *hyperplane*. Dari masing-masing pengguna akan dibuatkan modelnya masing-masing, sehingga data suara akan dikelompokkan untuk masing-masing pengguna. Kelompok data akan diproses dengan MFCC dan dibagi menjadi data latih dan data uji. Data uji yang digunakan adalah 2 data suara, yaitu 1 data suara dari pengguna yang asli dan 1 data suara dari pengguna lain yang ada dalam kelompok data tersebut. Data latih adalah data suara yang tidak dipilih sebagai data uji.

Data tersebut akan dilatih dengan kernel linear dan kernel polinomial dengan parameter yang sudah diatur. Setelah pelatihan data model yang terbentuk dihitung nilai akurasi pelatihan dengan mendapatkan nilai rata-rata hasil *k-fold cross validation*. *K-fold cross validation* merupakan salah satu metode pengambilan ulang sampel untuk mengevaluasi model hasil *machine learning* dalam data sampel yang terbatas[7]. Sehingga kernel yang digunakan adalah jika hasil akurasi pelatihan dengan kernel linear lebih besar dari 50% dan lebih besar dari nilai akurasi pelatihan dengan kernel polinomial maka kernel linear akan dipilih. Tetapi jika nilai akurasi pelatihan dengan kernel polinomial lebih besar dari 50% dan lebih besar dari nilai akurasi pelatihan dengan kernel linear, maka kernel polinomial akan dipilih. Dan jika nilai akurasi dari kedua kernel kurang dari 50% maka akan dilakukan pencarian parameter yang sesuai dengan metode *Grid Search* dan setelah didapatkan hasil parameter yang sesuai, data latih akan dilatih kembali dan dilakukan pengujian data suara tersebut. Hasil yang didapatkan dari klasifikasi data suara adalah 1 atau 0, dengan keterangan 1 menjadi data positif dan 0 menjadi data negatif.

#### 2.4 Metode evaluasi

Metode evaluasi yang digunakan adalah *Confusion Matrix*. *Confusion matrix* akan membandingkan hasil prediksi dengan hasil sebenarnya untuk mendapatkan jumlah data yang memiliki hasil yang berbeda dari hasil sebenarnya. Terdapat 4 kombinasi dari hasil prediksi dengan hasil sebenarnya, yaitu *True Positif*, *True Negatif*, *False Positif* dan *False Negatif*[8].

		True Class	
		Positive	Negative
Predicted Class	Positive	TP	FP
	Negative	FN	TN

**Gambar 5** *Confusion Matrix* (Sumber: Joydwip Mohajon, [Confusion Matrix for Your Multi-Class Machine Learning Model](https://towardsdatascience.com/confusion-matrix-for-your-multi-class-machine-learning-model-ff9aa3bf7826), <https://towardsdatascience.com/confusion-matrix-for-your-multi-class-machine-learning-model-ff9aa3bf7826>, 8 Agustus 2021.)

Dari hasil perbandingan data prediksi dengan data sebenarnya, dapat dihitung nilai akurasi, presisi, *recall*, *F-1 Score* dengan formula sebagai berikut:

$$Accuracy = \frac{TP+TN}{TP+FP+FN+TN} \quad (3)$$

$$Precision = \frac{TP}{TP+FP} \quad (4)$$

$$Recall = \frac{TP}{TP+FN} \quad (5)$$

$$F - 1 \text{ Score} = \frac{2*Recall*Precision}{Recall+Precision} \quad (6)$$

### 3. HASIL DAN PEMBAHASAN

Untuk mengidentifikasi pemilik suara dengan menggunakan teknik klasifikasi *Support Vector Classifier* (SVC), tentunya data suara diperoleh dari pengguna yang merekam suara dengan mengucapkan kalimat yang sudah disediakan. Pada saat perekaman suara, pengucapan kalimat oleh pengguna akan dilakukan pengecekan terhadap kesamaan kalimat antara kalimat yang disediakan dan kalimat yang diucapkan dengan menggunakan metode *speech recognition* dalam bahasa Indonesia. Google Speech to Text API digunakan untuk mengenali kalimat atau kata yang diucapkan oleh pengguna. Dilakukan pengujian terhadap *speech recognition* untuk mengetahui seberapa akurat pengenalan kalimat yang diucapkan. Pengujian dilakukan oleh 3 pengguna dengan mengucapkan kalimat yang berbeda sebanyak 3 kali. Pengujian *speech to text* API dapat dilihat pada Tabel 1.

**Tabel 1.** Pengujian Google Speech to Text API

Percobaan	Pembicara	Kalimat	Hasil <i>speech to text</i>
1	Pembicara 1	Apakah kamu baik-baik saja	Apakah kamu baik-baik saja
2		Semangka dan melon	-
3		Tertawa bahagia	Tertawa bahagia
4	Pembicara 2	Bayi masih sangat muda	Bayi masih sangat muda
5		Kucing berjalan diatas pagar	-
6		Kakak senang dengan hobinya	Kak senang dengan hobinya
7	Pembicara 3	Semut berkumpul dengan kawannya	Semut berkumpul dengan kawannya
8		Anginnya kencang sekali	Anginnya kencang sekali
9		Adik suka bermain	Adik suka bermain

Dari Tabel 1 diatas, terdapat 3 kali percobaan dari 9 kali percobaan dimana Google Speech to Text gagal mendeteksi kalimat yang diucapkan dengan benar. Selanjutnya, rekaman suara tersebut akan diolah dengan menggunakan metode Median Filter sebagai filterisasi terhadap gangguan yang ada pada sinyal suara. Mengekstrak ciri suara merupakan salah satu proses yang akan dilakukan setelah metode Median Filter, yaitu dengan menggunakan metode MFCC.

Data suara yang sudah diproses menggunakan Median Filter dan MFCC selanjutnya akan dilatih dan diuji dengan menggunakan kernel linear dan kernel polinomial. Tahap ini dilakukan kepada 6 pengguna dan masing-masing pengguna akan dibuatkan modelnya sendiri. Berikut hasil pelatihan data suara dapat dilihat pada Tabel 2.

**Tabel 2.** Pelatihan model uji kernel linear dan kernel polinomial

Model	Jumlah data latih berlabel 0	Jumlah data latih berlabel 1	Akurasi kernel linear	Akurasi kernel polinomial
Model 1 (Pengguna 1)	8	8	57%	50%
Model 2 (Pengguna 2)	6	6	88%	88%
Model 3 (Pengguna 3)	10	10	52%	60%
Model 4 (Pengguna 4)	10	10	60%	60%
Model 5 (Pengguna 5)	7	7	60%	60%
Model 6 (Pengguna 6)	5	5	75%	75%

Dari Tabel 2 hasil pelatihan data, kernel yang dipilih adalah kernel dengan nilai akurasi tertinggi. Dan dilakukan pengujian data sebanyak 10 kali percobaan untuk 1 model dengan skema pengujian yang sama. Hasil pengujian dapat dilihat pada Tabel 3.

**Tabel 3** Pengujian klasifikasi data suara

Percobaan ke-	Akun	Suasana	Pembicara	Hasil
1	Pengguna 1	Ramai	Pengguna 1	Berhasil
2		Tidak terlalu ramai	Pengguna 1	Berhasil
3		Tidak terlalu ramai	Pengguna 1	Berhasil
4		Ramai	Pengguna 1	Berhasil
5		Ramai sekali	Pengguna 1	Berhasil
6		Ramai	Pengguna 2	Gagal
7		Ramai	Pengguna 3	Gagal
8		Ramai	Pengguna 4	Gagal
9		Tidak terlalu ramai	Pengguna 5	Gagal
10		Tidak terlalu ramai	Pengguna 6	Gagal
11	Pengguna 2	Tidak terlalu ramai	Pengguna 2	Berhasil
12		Tidak terlalu ramai	Pengguna 2	Berhasil
13		Tidak terlalu ramai	Pengguna 2	Berhasil
14		Tidak terlalu ramai	Pengguna 2	Berhasil
15		Tidak terlalu ramai	Pengguna 2	Berhasil
16		Tidak terlalu ramai	Pengguna 1	Gagal
17		Tidak terlalu ramai	Pengguna 3	Gagal
18		Tidak terlalu ramai	Pengguna 4	Gagal
19		Sepi	Pengguna 5	Gagal
20		Sepi	Pengguna 6	Gagal
21	Pengguna 3	Sepi	Pengguna 3	Berhasil
22		Tidak terlalu ramai	Pengguna 3	Berhasil
23		Tidak terlalu ramai	Pengguna 3	Berhasil
24		Ramai	Pengguna 3	Berhasil
25		Tidak terlalu ramai	Pengguna 3	Berhasil
26		Tidak terlalu ramai	Pengguna 1	Gagal
27		Sepi	Pengguna 2	Gagal
28		Tidak terlalu ramai	Pengguna 4	Gagal
29		Ramai	Pengguna 5	Gagal
30		Sepi	Pengguna 6	Gagal
31	Pengguna 4	Tidak terlalu ramai	Pengguna 4	Berhasil
32		Tidak terlalu ramai	Pengguna 4	Berhasil
33		Tidak terlalu ramai	Pengguna 4	Berhasil
34		Sepi	Pengguna 4	Berhasil
35		Sepi	Pengguna 4	Berhasil
36	Pengguna 4	Ramai	Pengguna 1	Gagal
37		Sepi	Pengguna 2	Gagal
38		Tidak terlalu ramai	Pengguna 3	Berhasil
39		Sepi	Pengguna 5	Gagal
40		Sepi	Pengguna 6	Berhasil
41	Pengguna 5	Tidak terlalu ramai	Pengguna 5	Berhasil
42		Tidak terlalu ramai	Pengguna 5	Berhasil
43		Tidak terlalu ramai	Pengguna 5	Berhasil
44		Tidak terlalu ramai	Pengguna 5	Berhasil
45		Tidak terlalu ramai	Pengguna 5	Berhasil
46		Ramai sekali	Pengguna 1	Gagal
47		Ramai	Pengguna 2	Gagal
48		Tidak terlalu ramai	Pengguna 3	Berhasil

Percobaan ke-	Akun	Suasana	Pembicara	Hasil
49	Pengguna 6	Sepi	Pengguna 4	Berhasil
50		Sepi	Pengguna 6	Gagal
51		Sepi	Pengguna 6	Berhasil
52		Sepi	Pengguna 6	Berhasil
53		Sepi	Pengguna 6	Berhasil
54		Sepi	Pengguna 6	Berhasil
55		Sepi	Pengguna 6	Berhasil
56		Tidak terlalu ramai	Pengguna 1	Gagal
57		Tidak terlalu ramai	Pengguna 2	Berhasil
58		Tidak terlalu ramai	Pengguna 3	Gagal
59		Sepi	Pengguna 4	Gagal
60		Sepi	Pengguna 5	Gagal

Dari hasil pengujian Tabel 3, selanjutnya dapat dinilai hasil klasifikasi yang telah didapat dengan menggunakan metode *confusion matrix*, seperti pada Tabel 4.

**Tabel 4** Visualisasi *confusion matrix*

Prediksi/Aktual	1	0
1	30	5
0	0	25

Pada Tabel 4 diatas, menunjukkan bahwa dari 60 kali percobaan, didapatkan 30 kali percobaan program berhasil mengenali suara asli pengguna (*True Positive*), 25 kali percobaan program berhasil menolak suara pengguna lain (*True Negative*), dan 5 kali percobaan program salah mengenali suara pengguna lain sebagai suara dari pemilik akun (*False Positive*).

**Tabel 5** Hasil *confusion matrix*

Nama	Hasil
Akurasi	91,67%
Presisi	85,7%
Recall	100%
F-1 Score	92,3%

#### 4. KESIMPULAN

Berdasarkan penelitian yang telah dilakukan, dapat diambil kesimpulan bahwa program untuk mengidentifikasi pemilik suara dengan MFCC dan SVC kernel linear dan kernel polinomial dapat mengklasifikasi suara dengan cukup baik dan didapatkan tingkat akurasi sebesar 91,67%. Metode ini dapat menjadi salah satu alternatif untuk menjadi teknik keamanan dalam menjaga akun pengguna dengan mengidentifikasi pemilik suara pada saat proses *login*.

#### UCAPAN TERIMA KASIH

Ucapan terima kasih yang sebesar-besarnya diberikan oleh penulis kepada para responden atas ketersediaan dalam mengisi angket untuk tujuan pengumpulan data suara.



## DAFTAR PUSTAKA

- [1] Justusson, B.I., 1981, Median Filtering: Statistical Properties. In: Two-Dimensional Digital Signal Processing II, Topics in Applied Physics, vol 43, <https://link.springer.com/chapter/10.1007/BFb0057597>, diakses pada tanggal 27 Agustus 2021.
- [2] Manunggal, H. S., 2005, Perancangan dan Pembuatan Perangkat Lunak Pengenalan Suara Pembicara dengan Menggunakan Analisa MFCC Feature Extraction, <http://repository.petra.ac.id/7329/>, diakses pada tanggal 28 Agustus 2021.
- [3] Ismail, A., Idris, M. Y. I., Noor, N. M., Razak, Z., Yusoff, Z. M., 2014, MFCC-VQ Approach for Qalqalah Tajweed Rule Checking. [https://www.researchgate.net/publication/279113646\\_MFCC-VQ\\_approach\\_for\\_Qalqalah\\_Tajweed\\_rule\\_checking](https://www.researchgate.net/publication/279113646_MFCC-VQ_approach_for_Qalqalah_Tajweed_rule_checking), diakses pada tanggal 27 Agustus 2021.
- [4] Laha, D. dan Mandal, P., 2007, Handbook of Computational Intelligence in Manufacturing and Production Manajemen, <https://www.semanticscholar.org/paper/Handbook-of-Computational-Intelligencein-and-Laha-Mandal/0eda16abb58ab3f5ba5846b99c5dc96b7bf1ac59>, diakses pada tanggal 27 Agustus 2021.
- [5] Mamyrbayev, O., Mekebayev, N., Turdalyuly, M., Oshanova, N., Medeni, T. I., Yessentay, A., 2019, Voice Identification Using Classification Algorithms, <https://www.intechopen.com/chapters/68705>, diakses pada tanggal 27 Agustus 2021.
- [6] Vapnik, V. dan Cortes, C., 1995, Support Vector Networks. Machine Learning, <https://link.springer.com/content/pdf/10.1007/BF00994018.pdf>, diakses pada tanggal 27 Agustus 2021.
- [7] Brownlee, J., 2018, A Gentle Introduction to k-fold Cross-Validation, <https://machinelearningmastery.com/k-fold-cross-validation/>, diakses pada tanggal 12 Oktober 2021.
- [8] Anggreany, M. S., Confusion Matrix, <https://socs.binus.ac.id/2020/11/01/confusion-matrix/>, diakses pada tanggal 7 Agustus 2021.