

ANALISIS AKURASI CNN PADA DATA OLAH SUARA MANUSIA MENGGUNAKAN PARAMETER KOEFISIEN MFCC DAN MAX LENGTH

Firdaus Noorhadi Rahman¹, Tri Listyorini², Endang Supriyati³

Universitas Muria Kudus

Jl. Lingkar Utara UMK, Gondangmanis, Bae, Kudus

e-mail: fridausrahman08@gmail.com¹, trilistyorini@umk.ac.id², endang.supriyati@umk.ac.id³

Abstrak

Di era digital, pengenalan suara manusia semakin berkembang sebagai salah satu solusi inovatif. Suara dapat digunakan untuk mempermudah dalam berbagai bidang. Penelitian ini bertujuan untuk menganalisis akurasi CNN pada data olah menggunakan parameter koefisien MFCC dan Max Length dalam mengklasifikasikan suara manusia berdasarkan delapan kelas suara yang mencakup tujuh kelas suara orang dan satu kelas suara bebas. Data suara menggunakan data primer dalam format WAV, kemudian diproses melalui dua tahapan preprocessing. Tahap pemrosesan ekstraksi fitur, menggunakan MFCC dengan parameter koefisien dan max length yang bervariasi. Arsitektur model CNN yang dirancang terdiri dari beberapa lapisan utama. Model CNN dilatih menggunakan data yang telah diproses dengan variasi parameter koefisien dan max length untuk mengevaluasi pengaruhnya terhadap tingkat akurasi dalam klasifikasi suara. Evaluasi dilakukan menggunakan Confusion Matrix untuk mengukur performa model yang diuji secara keseluruhan.

Hasil penelitian, analisis akurasi tertinggi sebesar 98,96%, diperoleh dengan parameter koefisien MFCC sebesar 40 dan Max Length 48000. Hal ini menunjukkan bahwa parameter koefisien MFCC dan Max Length berpengaruh terhadap tingkat akurasi model. Semakin besar parameter MFCC maka semakin tinggi tingkat akurasinya. Begitu juga dengan semakin besar Max Length maka semakin tinggi tingkat akurasinya. Kesimpulan dari penelitian ini adalah menekankan pentingnya parameter MFCC dan Max Length (panjang maksimal data) dalam meningkatkan akurasi model CNN.

Kata kunci: Akurasi, CNN, Data Olah, Suara.

Abstract

In the digital era, human voice recognition is growing as one of the innovative solutions. Voice can be used to facilitate in various fields. This research aims to analyze the accuracy of CNN on processing data using MFCC and Max Length coefficient parameters in classifying human voices based on eight voice classes which include seven classes of people's voices and one class of free voice. Voice data using primary data in WAV format, then processed through two preprocessing stages. The feature extraction processing stage uses MFCC with varying coefficient and max length parameters. The designed CNN model architecture consists of several main layers. The CNN model is trained using data that has been processed with varying coefficient and max length parameters to evaluate its effect on the level of accuracy in voice classification. The evaluation is done using Confusion Matrix to measure the overall performance of the tested model.

The highest accuracy analysis of 98.96% was obtained with MFCC coefficient parameter of 40 and Max Length of 48000. This shows that the MFCC coefficient parameter and Max Length affect the accuracy of the model. The greater the MFCC parameter, the higher the accuracy level. Likewise, the greater the Max Length, the higher the level of accuracy. The conclusion of this research is to emphasize the importance of MFCC and Max Length parameters in improving the accuracy of CNN models.

Keywords: Akurasi, CNN, Processing Data, Voice.

1. PENDAHULUAN

Pengenalan suara manusia merupakan salah satu teknologi biometrik yang terus berkembang pesat dan banyak diterapkan dalam berbagai sistem. Biometrik merupakan cabang ilmu yang mempelajari pola karakteristik tertentu untuk mengenali atau mengidentifikasi individu berdasarkan berbagai aspek tubuh manusia. Karakteristik tersebut dapat meliputi ciri perilaku atau fisik, seperti wajah, sidik jari, geometri tangan, iris mata, hingga suara[1]. Seiring dengan kemajuan teknologi, metode berbasis deep learning, khususnya Convolutional Neural Network (CNN) semakin berkembang dan memberikan hasil yang baik

dalam pengolahan data audio. CNN adalah algoritma deep learning yang dirancang untuk mengolah data berupa gambar sebagai inputnya[2]. Supaya CNN dapat mengolah data yang berupa gambar, maka perlu dilakukan pemrosesan pada suara. Pemrosesan suara adalah proses mengubah sinyal suara menjadi bentuk yang mudah dipahami dan informatif. Ekstraksi ciri dan klasifikasi sinyal suara adalah salah satu pendekatan yang digunakan dalam pemrosesan suara, yang berdampak langsung pada proses pengenalan suara[3].

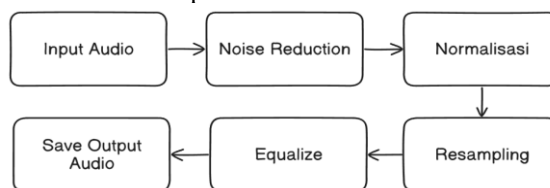
Akurasi model CNN bergantung terhadap berbagai aspek, seperti pengumpulan data, pengolahan data, ekstraksi fitur, arsitektur model dan evaluasi model. Dalam melakukan klasifikasi suara, parameter seperti MFCC dan Max Length menjadi salah satu aspek yang memengaruhi akurasi model. MFCC digunakan sebagai ekstraksi fitur yang bertujuan memperoleh cepstral coefficient dan frame yang berguna dalam proses pengenalan suara untuk meningkatkan akurasi klasifikasi[4]. MFCC melakukan ekstraksi fitur dengan mengubah sinyal suara menjadi variabel. Penelitian ini berfokus analisis akurasi model dalam mengolah data suara manusia dengan menggunakan parameter koefisien MFCC dan Max Length yang bervariasi sebagai parameter dalam pengujian model. Data yang digunakan adalah data primer yang terdiri dari delapan kelas, termasuk tujuh kelas suara orang dan satu kelas suara bebas. Salah satu penelitian yang pernah dilakukan tentang pengolahan data dengan cara melakukan augmentasi pada data, menyimpulkan bahwa augmentasi pada data dapat meningkatkan nilai akurasi terhadap model[5]. Akurasi yang didapatkan dalam penelitian tersebut sebesar 99,06%. Selain itu parameter pada data yang digunakan sebagai input juga berpengaruh terhadap akurasi model CNN.

2. METODE PENELITIAN



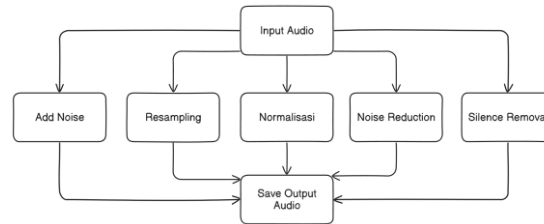
Gambar 1. Alur Tahapan Penelitian.

Data suara yang digunakan dalam penelitian ini merupakan data hasil rekaman suara dari orang yang berbeda. Data diambil secara langsung oleh peneliti, dengan cara pengambilan menggunakan smartphone sebagai alat untuk merekam dalam format WAV. Perekaman dilakukan dengan memperhatikan variasi jarak perekaman, supaya data yang dimiliki dapat bervariasi. Data suara yang dikumpulkan berasal dari tujuh orang, dimana setiap individu mengucapkan sebuah kalimat sebanyak sepuluh kali. Kemudian satu data suara bebas yang diambil perekaman secara acak sebanyak sepuluh kali. Data mentah yang sudah dikumpulkan akan melalui preprocessing atau pengolahan untuk menghasilkan data olah. Ada beberapa tahapan pengolahan yang dilakukan dalam penelitian ini.



Gambar 1. Alur Preprocessing Tahap Pertama

Tahap pertama pengolahan data meliputi proses noise reduction, normalisasi, resampling, dan equalize pada data audio yang sudah dikumpulkan. Audio yang sudah melewati proses tersebut akan disimpan menjadi sebuah file baru, sehingga akan menghasilkan data file audio yang sudah di noise reduction, normalisasi, resampling dan equalize. Hasil file audio yang sudah diolah akan masuk ke tahap pengolahan berikutnya.



Gambar 2. Alur Preprocessing Tahap Kedua

Tahap kedua pengolahan data adalah proses lanjutan dari pengolahan tahap pertama. Tahap kedua melakukan pengolahan meliputi proses add noise, resampling, normalisasi, noise reduction dan silence removal. Setiap proses akan menghasilkan data file audio baru yang sudah melewati masing-masing proses. Data yang sudah melewati preprocessing akan melalui proses ekstraksi fitur.

Ekstraksi fitur bertujuan untuk mengekstrak karakteristik pada sinyal ucapan untuk setiap kata sehingga dapat membedakan antara berbagai kata yang berbeda[6]. Convolutional Neural Network (CNN) merupakan salah satu model deep learning yang terkenal dan sering digunakan dalam berbagai bidang penelitian, termasuk pengenalan objek, wajah, tulisan tangan, dan suara[7]. CNN bekerja dengan mengekstraksi fitur langsung dari gambar, dan menghilangkan kebutuhan untuk ekstraksi fitur manual, sehingga tidak perlu mengidentifikasi fitur yang digunakan untuk mengklasifikasikan gambar. CNN menggabungkan fitur yang dipelajari dengan data input dan menggunakan lapisan konvolusi 2D, sehingga arsitektur ini cocok untuk memproses data 2D[8].

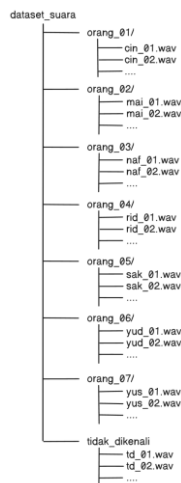
Confusion Matrix adalah metode pengujian yang memberikan gambaran tentang sejauh mana algoritma dalam klasifikasi berhasil melakukan prediksi dengan benar. Confusion Matrix memberikan gambaran tentang performa model dalam melakukan klasifikasi berdasarkan data yang diberikan[9]. Confusion Matrix merepresentasikan hasil dalam bentuk matriks yang dirancang khusus. Dalam kasus klasifikasi biner, informasi yang disajikan mencakup True Positive (TP) sebagai data positif benar, True Negatif (TN) sebagai data negatif benar, False Positive (FP) sebagai data negatif salah, False Negative (FN) sebagai data positif salah[10]. Analisis hasil dapat memberikan pemahaman mengenai hasil penelitian yang dilakukan.

3. HASIL DAN PEMBAHASAN

Hasil dan pembahasan meliputi beberapa tahapan yang akan dibahas mencakup data collection, data preprocessing, extraction features, implementasi model, evaluasi dan analisis hasil.

3.1. Data Collection

Data suara dikumpulkan berjumlah sebanyak 80 file audio, yang terdiri dari perekaman suara manusia sebanyak 70 audio, dan 10 audio adalah perekaman bebas. Suara direkam yang terdiri dari sebuah kalimat yang akan digunakan dalam penelitian ini. Data terdiri dari 8 kelas yaitu orang_01, orang_02, orang_03, orang_04, orang_05, orang_06, orang_07, dan tidak_dikenali.



Gambar 4. Struktur Data Suara dalam Folder

3.2. Data Preprocessing

Data yang sudah dikumpulkan, akan melalui pemrosesan agar data siap untuk digunakan ke dalam model. Pada eksperimen ini, data yang telah dikumpulkan akan diproses beberapa tahapan. Tahap pertama data diproses untuk dilakukan fungsi noise reduction, normalisasi, resampling dan equalize. Hasil dari pemrosesan pertama adalah berupa file audio yang akan di proses ke tahap berikutnya.



Gambar 5. Cuplikan Hasil Tahapan Proses Pertama

Tahap kedua, data akan diperbanyak, dengan cara dilakukan kembali pemrosesan data audio kemudian hasil audio disimpan berupa file audio. Data akan diperbanyak dengan cara melalui proses add noise, resampling, normalisasi, noise reduction dan silence removal. Hasil dari setiap proses tahap kedua yaitu menyimpan dalam bentuk file. Jumlah data yang akan dihasilkan adalah 480 file audio yang telah di proses.



Gambar 6. Hasil Tahapan Proses Kedua

Data yang telah melalui proses pertama dan kedua akan digabungkan menjadi satu folder dataset yang akan digunakan dalam model CNN. Folder dataset berisi seluruh gabungan data tersebut, diberi nama dataset_suara.

Tabel 1. Hasil Data Audio Setelah Proses Tahap Pertama dan Kedua

Data Audio	Label Kelas								Jumlah Audio
	orang_01	orang_02	orang_03	orang_04	orang_05	orang_06	orang_07	tidak_dikenali	
dataset_suara	60	60	60	60	60	60	60	60	480

3.3. Extraction Features

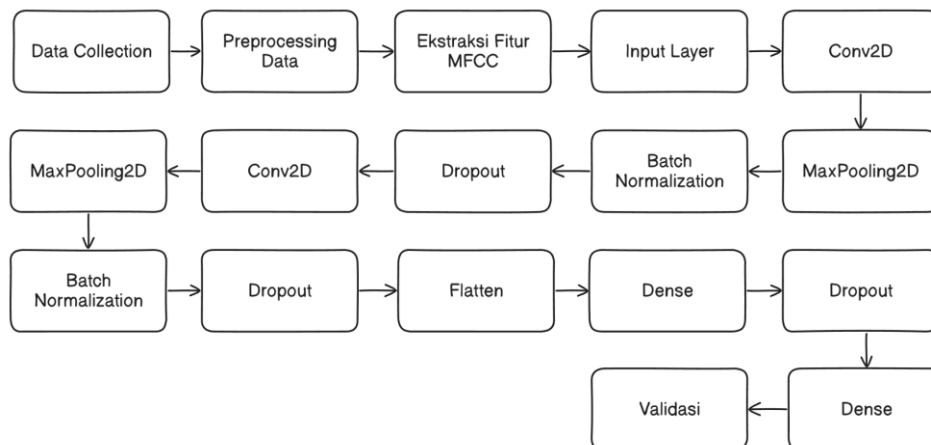
Sebelum memasuki ekstraksi fitur dalam model klasifikasi. Audio diproses melalui pemrosesan seperti Pre-Emphasis Filtering dan Padding atau Truncating. Audio diproses untuk memperkuat frekuensi tinggi dan mengurangi frekuensi rendah serta menyesuaikan panjang sinyal audio agar konsisten untuk setiap file. Penelitian ini, menggunakan ekstraksi fitur MFCC (Mel Frecuency Cepstral Coefficient). MFCC menghasilkan representasi numerik dan karakteristik suara yang akan menjadi input untuk model algoritma. Proses ekstraksi fitur yang digunakan untuk model algoritma CNN menggunakan ekstraksi fitur MFCC dalam bentuk matriks 2D.

```
# Ekstraksi MFCC
mfcc = librosa.feature.mfcc(y=signal, sr=sr, n_mfcc=n_mfcc)
```

Gambar 7. Kode Fungsi Ekstraksi Fitur

3.4. CNN

Data yang digunakan dalam pengujian akan dibagi menjadi data training dan data testing. Pengujian ini membagi data menjadi 80% data training dan 20% data testing. Pengujian CNN pada tahap ini menggunakan arsitektur seperti berikut.



Gambar 8. Arsitektur CNN

Pada penelitian ini, model dimulai dengan layer input yang mendefinisikan bentuk data input berupa gambar 2 dimensi. Layer konvolusi pertama menggunakan 32 filter, dengan setiap filter memiliki kernel berukuran 3×3, dan menggunakan fungsi aktivasi ReLU. Kemudian terdapat MaxPooling2D dengan ukuran 2×2, BatchNormalization untuk menormalkan nilai aktivasi, dan Dropout dengan nilai 0.25. Selanjutnya proses dari layer pertama akan dilanjutkan ke layer berikutnya.

Pada layer konvolusi kedua menggunakan 64 filter, dengan setiap filter memiliki ukuran kernel 3×3 dan fungsi aktivasi ReLU. Kemudian terdapat MaxPooling2D dengan ukuran 2×2, BatchNormalization, dan Dropout dengan nilai 0.25. Setelah semua proses konvolusi selesai, agar dapat berlanjut ke proses berikutnya, maka harus melalui layer Flatten terlebih dahulu.

Setelah melalui layer Flatten, proses akan dilanjutkan ke layer Dense. Layer Dense yang pertama menggunakan 128 neuron dan fungsi aktivasi ReLU. Kemudian dilanjutkan layer Dropout dengan nilai 0.5. Terakhir layer output yang terdiri dari Dense dengan menggunakan fungsi aktivasi softmax. Ada beberapa parameter yang akan digunakan dalam pengujian yaitu nilai koefisien MFCC dan panjang Max Length.

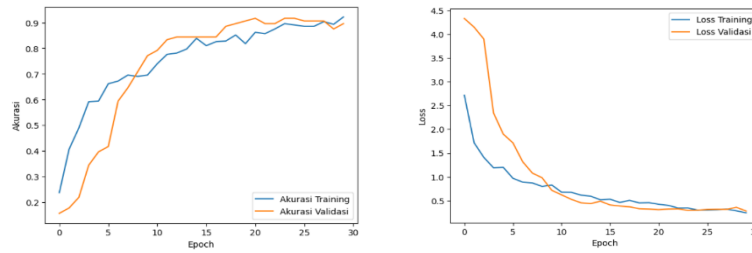
Pada tahap kompilasi, model menggunakan optimizer adam, loss function categorical_crossentropy yang sesuai untuk multi kelas, dan metric accuracy. Penelitian ini menggunakan epoch sebanyak 30 dan batch size sebesar 32, yang masing-masing bertujuan untuk mengontrol jumlah iterasi dan ukuran data yang diproses dalam pelatihan.

Tabel 2. Parameter Pengujian

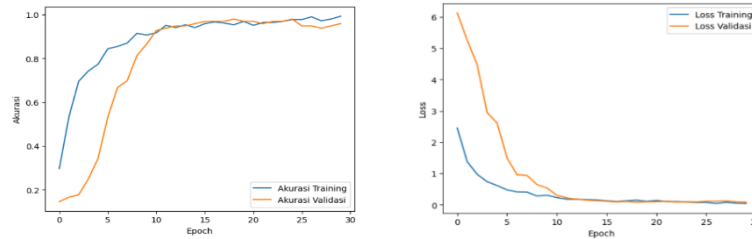
Koefisien MFCC		Max Length		
13	40	16000	32000	48000

Tabel 3. Akurasi Hasil Pengujian

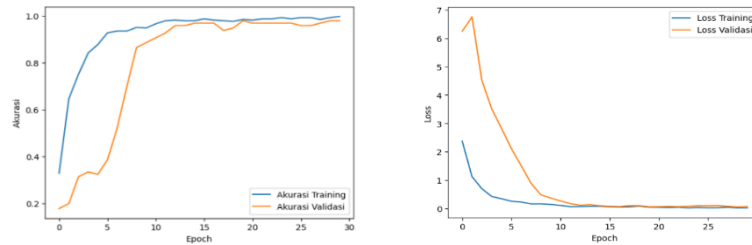
Parameter		Akurasi
Koefisien MFCC	Max Length	
13	16000	89,58%
13	32000	95,83%
13	48000	97,92%
40	16000	94,79%
40	32000	96,88%
40	48000	98,96%



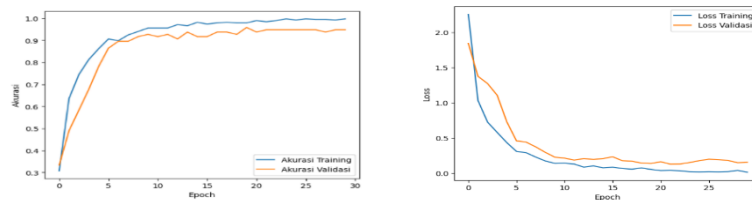
Gambar 9. Grafik Akurasi dan Loss Parameter Koefisien MFCC 13, Max Length 16000



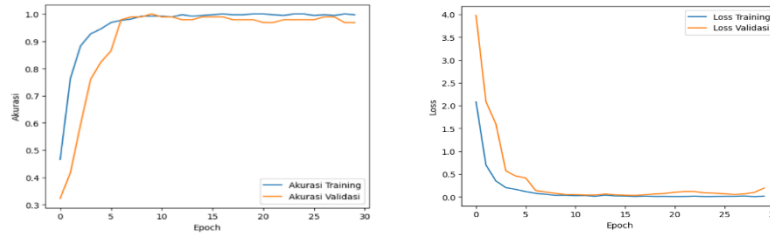
Gambar 10. Grafik Akurasi dan Loss Parameter Koefisien MFCC 13, Max Length 32000



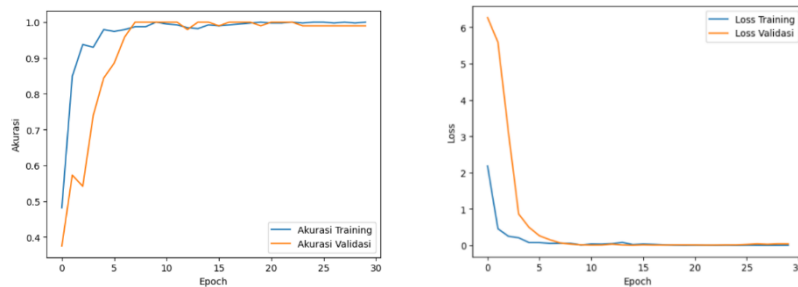
Gambar 11. Grafik Akurasi dan Loss Parameter Koefisien MFCC 13, Max Length 48000



Gambar 12. Grafik Akurasi dan Loss Parameter Koefisien MFCC 40, Max Length 16000

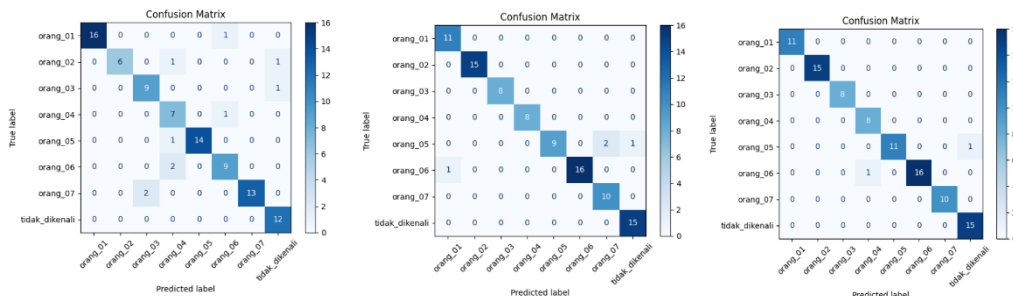


Gambar 13. Grafik Akurasi dan Loss Parameter Koefisien MFCC 40, Max Length 32000

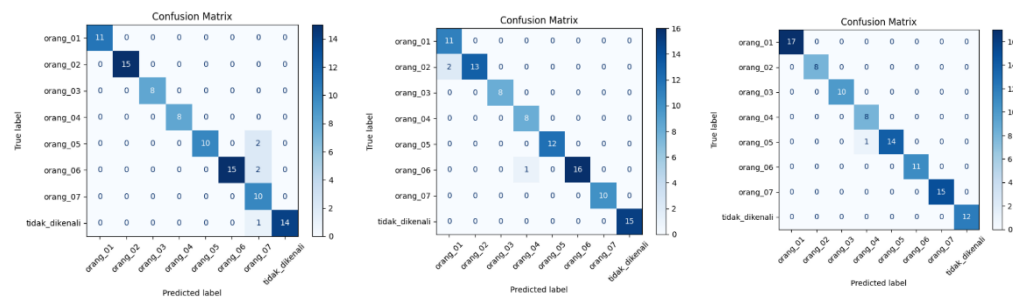


Gambar 14. Grafik Akurasi dan Loss Parameter Koefisien MFCC 40, Max Length 48000

3.5. Evaluasi Hasil



Gambar 15. Koefisien MFCC 13, Max Length 16000 (Kiri), 32000 (Tengah), 48000 (Kanan)



Gambar 16. Koefisien MFCC 40, Max Length 16000 (Kiri), 32000 (Tengah), 48000 (Kanan)

3.6. Analisis Hasil

Dari pengujian yang telah dilakukan Confusion Matrix pada evaluasi model dengan parameter koefisien MFCC 13 dan Max Length 16000 menunjukkan hasil akurasi model mencapai 89,58%. Data yang diuji berjumlah 96 data, dimana sebanyak 86 data berhasil diprediksi dengan benar sesuai dengan kelasnya, sementara 10 data lainnya salah diprediksi. Confusion Matrix dengan parameter koefisien MFCC 13 dan Max Length 32000 menunjukkan hasil akurasi model mencapai 95,83%. Data yang diuji berjumlah 96 data, dimana model berhasil memprediksi sebanyak 92 data dengan benar sesuai dengan kelasnya, sementara 4 data lainnya salah diprediksi. Confusion Matrix dengan parameter koefisien MFCC 13 dan Max Length 48000 menunjukkan hasil akurasi mencapai 97,92%. Data yang diuji berjumlah 96 data, dimana sebanyak 94 data berhasil diprediksi dengan benar, sementara 2 data lainnya salah diprediksi.

Confusion Matrix pada evaluasi model dengan parameter koefisien MFCC 40 dan Max Length 16000 menunjukkan hasil akurasi model mencapai 94,79%. Data yang diuji berjumlah 96 data, dimana model berhasil memprediksi sebanyak 91 data dengan benar, sementara 5 data lainnya salah diprediksi. Confusion Matrix dengan parameter koefisien MFCC 40 dan Max Length 32000 menunjukkan hasil akurasi model mencapai 96,88%. Data yang diuji berjumlah 96 data, dimana model berhasil memprediksi sebanyak 93 data dengan benar, sementara 3 data lainnya salah diprediksi. Confusion Matrix dengan parameter koefisien MFCC 40 dan Max Length 48000 menunjukkan hasil akurasi model mencapai 98,96%. Data yang diuji berjumlah 96 data, dimana model berhasil memprediksi sebanyak 95 data dengan benar, sementara 1 data lainnya salah diprediksi.

Tabel 4. Analisis Koefisien MFCC Sama, Max Length Berbeda

Parameter		Akurasi	Parameter		Akurasi
Koefisien MFCC	Max Length		Koefisien MFCC	Max Length	
13	16000	89,58%	40	16000	94,79%
13	32000	95,83%	40	32000	96,88%
13	48000	97,92%	40	48000	98,96%

Berdasarkan Tabel 4 pada koefisien MFCC 13, akurasi meningkat seiring dengan bertambahnya nilai *Max Length*. *Max Length* 16000 memiliki akurasi sebesar 89,58%, menunjukkan performa yang cukup rendah dibandingkan *Max Length* yang lebih panjang. Hal ini terjadi karena panjang sinyal yang pendek tidak mampu merepresentasikan informasi temporal dengan cukup baik. *Max Length* 32000 memiliki akurasi sebesar 95,83%, menunjukkan akurasi meningkat daripada sebelumnya. Hal ini menunjukkan bahwa penambahan panjang sinyal membantu model menangkap lebih banyak pola atau fitur penting dari data. *Max Length* 48000 memiliki akurasi yang lebih tinggi dibandingkan 2 pengujian sebelumnya. Akurasi dengan *Max Length* 48000 mencapai 97,92%, menunjukkan panjang sinyal ini memberikan informasi temporal yang lebih lengkap sehingga model dapat mengenali pola.

Pada koefisien MFCC 40, pola yang sama terlihat seperti MFCC 13, yaitu akurasi meningkat dengan bertambahnya nilai *Max Length*. *Max Length* 16000 memiliki akurasi sebesar 94,97%, lebih tinggi dibandingkan *Max Length* 16000 pada MFCC 13. *Max Length* 32000 memiliki akurasi sebesar 96,88%, mengindikasikan bahwa panjang sinyal menangkap lebih banyak informasi dari sebelumnya. *Max Length* 48000 memiliki akurasi tertinggi yaitu mencapai 98,96%. Hal ini menunjukkan bahwa kombinasi panjang sinyal yang besar dan jumlah koefisien tinggi menghasilkan performa terbaik.

Tabel 5. Analisis Koefisien MFCC Berbeda, Max Length Sama

Parameter		Akurasi	Parameter		Akurasi	Parameter		Akurasi
Koefisien MFCC	Max Length		Koefisien MFCC	Max Length		Koefisien MFCC	Max Length	
13	16000	89,58%	13	32000	95,83%	13	48000	97,92%
40	16000	94,79%	40	32000	96,88%	40	48000	98,96%

Berdasarkan Tabel 5 pada *Max Length* 16000, untuk koefisien MFCC 13 memiliki akurasi sebesar 89,58%, sedangkan untuk koefisien MFCC 40 memiliki akurasi sebesar 94,79%. Pada *Max Length* 32000, untuk koefisien MFCC 13 memiliki akurasi sebesar 95,83%, sedangkan untuk koefisien MFCC 40 memiliki akurasi sebesar 96,88%. Pada *Max Length* 48000, untuk koefisien MFCC 13 memiliki akurasi sebesar 97,92%, sedangkan untuk koefisien MFCC 40 memiliki akurasi sebesar 98,96%. Hasil ini menunjukkan bahwa lebih banyak koefisien MFCC, seperti MFCC 40 dibandingkan dengan MFCC 13, membantu meningkatkan akurasi model karena dapat menangkap lebih banyak informasi.

4. KESIMPULAN

Hasil dari penelitian ini menunjukkan bahwa variasi parameter koefisien *Mel Frequency Cepstral Coefficient* (MFCC) dan *Max Length* (panjang maksimal data) memengaruhi tingkat akurasi model *Convolutional Neural Network* (CNN) secara signifikan. Data yang digunakan merupakan data primer dengan delapan kelas, meliputi tujuh kelas suara orang, dan satu kelas suara bebas yang telah melalui

beberapa tahapan *preprocessing*. Pengolahan data memiliki peran penting dalam mempersiapkan data untuk digunakan pada model, yang dapat berdampak terhadap akurasi model.

Hasil mengungkap bahwa kombinasi parameter koefisien MFCC dan *Max Length* data secara signifikan memengaruhi tingkat akurasi model CNN. Model menunjukkan performa terbaik pada parameter tertentu, yang mencerminkan atau menegaskan pentingnya pemilihan parameter yang optimal dalam proses pelatihan. Selain itu jumlah koefisien MFCC yang lebih besar memberikan representasi frekuensi yang lebih detail, sementara *Max Length* (panjang maksimal data) memengaruhi kemampuan model dalam menangkap informasi temporal. Kesimpulannya, parameter koefisien MFCC dan *Max Length* berpengaruh terhadap akurasi model CNN.

DAFTAR PUSTAKA

- [1] Widodo YF, Sunardi S, Fadlil A. Identifikasi Suara Pada Sistem Presensi Karyawan Dengan Metode Ekstraksi MFCC. *J-SAKTI (Jurnal Sains Komput Dan Inform 2019;3:115*. <https://doi.org/10.30645/j-sakti.v3i1.107>.
- [2] Scraping W, Recognition S. Identifikasi Audio Ancaman Menggunakan Metode Convolutional Neural Network Threat Audio Identification Using Convolutional Neural Network Method 2022;10:446–52. <https://doi.org/10.26418/justin.v10i4.52433>.
- [3] Putera TA, Prasetyo BH. Sistem Deteksi Depresi melalui Pengenalan Pola Suara dengan Mengimplementasikan Metode Prosody Analysis. *J Pengemb Teknol Inf ... 2024;1:1–10*.
- [4] Oktaviani M, Sutanto TE, Mahmudi M. Klasifikasi Usia Berdasarkan Suara Dengan Ekstraksi Ciri Mel Frequency Cepstral Coefficients Menggunakan Support Vector Machine. *Kesatria J Penerapan Sist Inf (Komputer Dan Manajemen) 2023;4:901–7*.
- [5] Ahmad Khairul Umam. Augmentasi data pada model klasifikasi jenis vokal menggunakan CNN dengan fitur ekstraksi mel spectrogram. 2023.
- [6] Helmiyah S, Fadlil A, Yudhana A, Informatika MT, Dahlan UA. Pengenalan Pola Emosi Manusia Berdasarkan Ucapan Menggunakan Ekstraksi Fitur Mel-Frequency Cepstral Coefficients (MFCC) Speech Based Emotion Pattern Recognition Using Mel- Frequency Cepstral Coefficients (MFCC) Feature Extraction 2018;4:372–81.
- [7] Yohannes Y, Wijaya R. Klasifikasi Makna Tangisan Bayi Menggunakan CNN Berdasarkan Kombinasi Fitur MFCC dan DWT. *JATISI (Jurnal Tek Inform Dan Sist Informasi) 2021;8:599–610*. <https://doi.org/10.35957/jatisi.v8i2.470>.
- [8] Rendi Nurcahyo, Mohammad Iqbal. Pengenalan Emosi Pembicara Menggunakan Convolutional Neural Networks. *J RESTI (Rekayasa Sist Dan Teknol Informasi) 2022;6:115–22*. <https://doi.org/10.29207/resti.v6i1.3726>.
- [9] Erinsyah MF, Karenina V, Wibowo DS. Klasifikasi Rentang Usia Dan Gender Dengan Deteksi Suara Menggunakan Metode Deep Learning Algoritma CNN (Convolutional Neural Network) Classification With Gender Voice Recognition Detection Deep Learning Method Using CNN (Convolutional Neural Network) 2023;12. <https://doi.org/10.34010/komputika.v12i2.10516>.
- [10] Farida LN, Bahri S. *Komputika : Jurnal Sistem Komputer Klasifikasi Gagal Jantung Menggunakan Metode SVM (Support Vector Machine) Classification of Heart Failure using the SVM (Support Vector Machine) Method 2025;13:0–7*. <https://doi.org/10.34010/komputika.v13i2.11330>.