BAB II
TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Penelitian sebelumnya berisi kajian terhadap studi-studi yang memiliki kesamaan atau kemiripan dengan penelitian ini, dengan tujuan untuk menunjukkan perbedaan, memperbaiki, atau mengembangkan penelitian sebelumnya sehingga dapat ditemukan aspek kebaruan. Oleh karena itu, dalam bagian tinjauan pustaka ini, peneliti menyajikan beberapa hasil dari penelitian yang telah dilakukan sebelumnya, sebagai berikut:

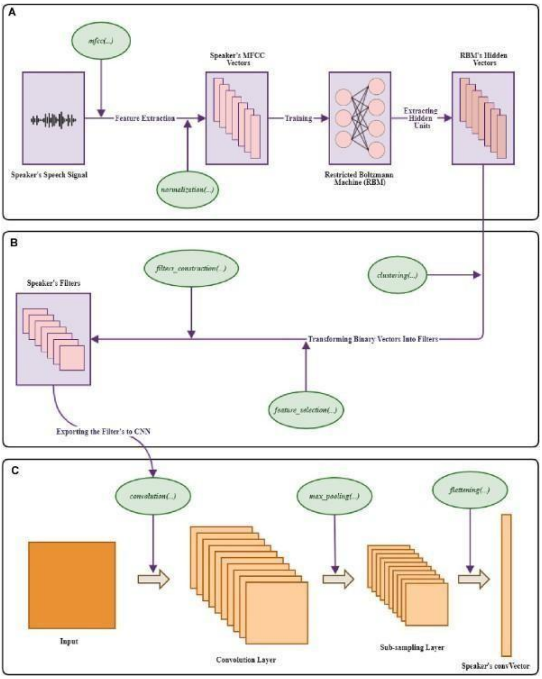
2.1.1 Hourri, S., Nikolov, N.S. & Kharroubi, J. Convolutional neural Network vectors for speaker recognition. Int J Speech Technol 24, 389 400 (2021)

Penelitian ini menjelaskan langkah yang dilakukan untuk mengenali suara bicara manusia. Dengan mengubah sinyal suara diubah menjadi spektrogram, yaitu representasi visual dari sinyal suara dalam bentuk dua dimensi yang memetakan frekuensi (sumbu vertikal), waktu (sumbu horizontal), dan intensitas (warna atau skala abu-abu). CNN dirancang untuk mengolah data berbentuk grid seperti ini, yang memungkinkan model mengenali pola-pola spesifik dalam suara manusia.

Convolutional Neural Networks (CNNs) telah berhasil diterapkan dalam mengenali pembicara dengan mengekstraksi karakteristik unik dari fitur suara mereka, sering kali menggunakan spektrogram sebagai data input. Jaringan ini sangat efektif untuk data dua dimensi, memungkinkan identifikasi pola dan fitur dalam representasi suara, yang sangat penting dalam tugas-tugas pengenalan pembicara maupun suara. (Hourri, S., Nikolov, N. S., & Kharroubi, J. 2021).

Kekurangan utama dari pendekatan ini adalah terlalu berfokus pada intonasi dalam bicara, yang berbeda dengan pola melodi dan ritme pada humming. Pendekatan ini kurang optimal untuk diterapkan pada sistem

deteksi lagu berbasis humming, karena tidak sepenuhnya memperhatikan aspek frekuensi nada dan perubahan temporal yang merupakan elemen kunci dalam melodi.



Gambar 2.1 Alur Deteksi Suara Manusia

2.1.2 Marar, S., Sheikh, F., Swain, D., Joglekar, P. (2020). HummingBased Song Recognition. In: Swain, D., Pattnaik, P., Gupta, P. (eds) Machine Learning and Information Processing.

Penelitian ini membahas berbagai teori dan teknik yang diterapkan dalam bidang Machine Learning dan pemrosesan informasi, dengan penekanan khusus pada pengenalan suara melalui humming. Penggunaan humming untuk pengenalan lagu (Humming-Based Song Recognition atau HBSR) adalah sistem yang menerima audio humming sebagai input dan

menganalisis audio tersebut untuk memprediksi lagu yang sesuai dengan input tersebut. Teknik pencarian musik dengan humming berguna ketika kita tidak dapat mengingat lirik lagu, tetapi hanya mengingat melodinya. (Shreerag Marar, Faisal Sheikh, Debabrata Swain, dan Pushkar Joglekar. 2020)

Neural Networks, khususnya Convolutional Neural Networks (CNN), juga menjadi pusat perhatian dalam penelitian ini. CNN digunakan untuk menganalisis fitur audio dan mendeteksi pola suara dari input humming. Meskipun CNN sangat efektif dalam menangkap pola-pola yang kompleks, jaringan syaraf tiruan ini memerlukan daya komputasi yang besar dan seringkali sulit untuk diinterpretasikan.

Keterbatasan dalam penelitian ini membutuhkan jumlah data yang besar untuk melatih model Convolutional Neural Network (CNN). Penelitian ini berfokus pada pengenalan suara pembicara manusia, yang mengharuskan model dilatih menggunakan beragam variasi suara, intonasi, dan aksen dari sejumlah individu.

2.2.3 Henry Hartono., Viny Christanti Mawardi, M.Kom., Janson Hendryli S. Kom. M.Kom. (2021). Perancangan Sistem Pencarian Lagu Indonesia menggunakan Query By Humming Berbasis Long Short- Term Memory.

Penelitian ini membahas sistem pencarian lagu berbasis Query by Humming (QBH) yang bertujuan untuk mengidentifikasi judul lagu berdasarkan input senandung dari pengguna

Pengujian dilakukan dengan menggunakan data humming yang mencakup beberapa lagu Indonesia, di mana model dilatih dan diuji menggunakan berbagai parameter seperti ukuran batch dan jumlah epoch. Hasil pengujian terhadap 14 data dari dua pengguna pria menunjukkan tingkat akurasi sebesar 50% pada pengguna pertama dan 35% pada

pengguna kedua. Hal ini menekankan pentingnya kualitas input suara serta kondisi lingkungan saat proses humming berlangsung.

Namun, sistem ini memiliki beberapa keterbatasan. Akurasi sistem sangat bergantung pada kondisi lingkungan dan kualitas suara yang dihasilkan oleh pengguna. Sistem berfungsi lebih optimal ketika humming dilakukan di lingkungan yang tenang, tetapi performa menurun secara signifikan apabila terdapat gangguan suara atau noise. Selain itu, perbedaan dalam kejelasan nada senandung antar pengguna turut mempengaruhi hasil deteksi lagu.

2.2.4 Fahrizal Adnan , Ilya Amelia, dan Sayyid 'Umar Shiddiq. (2022). Implementasi Voice Recognition Berbasis Machine Learning.

Teknologi pengenalan suara memungkinkan komputer untuk menangkap dan mengenali kata-kata yang diucapkan secara lisan. Teknologi ini memungkinkan komputer untuk memahami bahasa manusia dengan mengubah suara menjadi teks (Adnan, Amelia, & Shiddiq, 2022). Penelitian ini membahas implementasi sistem pengenalan suara berbasis Machine Learning yang menggunakan teknologi Speech Recognition untuk menjadikan audio ucapan sebagai teks tertulis.

Studi ini memanfaatkan library Speech Recognition dan Pyaudio untuk merekam suara secara langsung maupun dari file audio yang telah tersedia. Fokus utama penelitian ini adalah menguji tingkat akurasi dalam proses konversi suara ke bentuk teks, dengan hasil yang menunjukkan akurasi sebesar 0,97 untuk input real-time dan 0,93 saat menggunakan file audio.

Dalam penelitian ini, penulis menjelaskan proses pengenalan suara yang melibatkan beberapa tahapan, termasuk preprocessing, ekstraksi fitur, dan klasifikasi. Namun, terdapat beberapa kekurangan dalam penelitian ini.

Pertama, membutuhkan ukuran dataset yang besar untuk proses pelatihan model.

2.2.5 Ababil Azies Sasilo, Rizal Adi Saputra , Ika Purwanti Ningrum. (2022). Sistem Pengenalan Suara menggunakan Metode Mel Frequency Cepstral Coefficients dan Gaussian Mixture Model

Penelitian ini menitikberatkan pada teknologi pengenalan suara berbasis biometrik yang dirancang untuk mengenali dan membedakan karakteristik vokal individu. Teknologi biometrik ini dipandang memiliki potensi yang besar karena kemampuannya untuk mengidentifikasi individu berdasarkan karakteristik unik dari sinyal suara.

MFCC dikenal sebagai salah satu pendekatan paling efektif dalam proses ekstraksi fitur dari suara. (Ababil Azies Sasilo, Rizal Adi Saputra, dan Ika Purwanti Ningrum. (2022). Metode MFCC berperan dalam mengubah sinyal suara menjadi data vektor yang mewakili karakteristik frekuensi suara, sedangkan GMM berfungsi sebagai algoritma untuk mengklasifikasikan suara berdasarkan model statistik. Kelemahan dari penelitiaan ini masih kurang efektif dalam mengekstrak data dikarenakan tidak maksimal dalam penggunaan MFCC.

Research gap :

1. Penelitian terdahulu masih berfokus pada deteksi suara dengan intonasi sehingga masih memiliki keterbatasan dalam pengekstrakan suara.

2. Membutuhkan dataset yang sangat besar untuk melakukan model training.

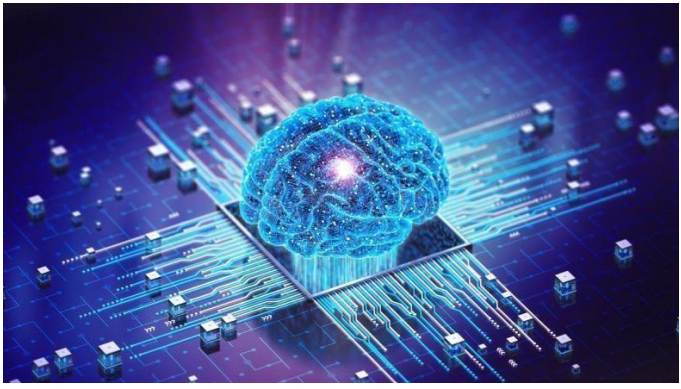
2.2 Teori Terkait

Berikut beberapa teori terkait yang digunakan dalam penelitian ini.

2.2.1 Artificial Intelligence

Harvei Desmon Hutahaean (2016) menjelaskan bahwa istilah Kecerdasan Buatan berasal dari dua kata, yakni 'artificial' yang berarti buatan, dan 'intelligence' yang berarti kecerdasan, berarti meniru kemampuan kognitif manusia agar mesin dapat mengambil keputusan secara mandiri dalam situasi tertentu.

Umumnya, entitas buatan ini berupa sistem komputer yang dirancang untuk meniru proses kognitif manusia. Kecerdasan ini dikembangkan dan diintegrasikan ke dalam mesin, sehingga memungkinkan komputer untuk menjalankan fungsi- fungsi yang pada umumnya melibatkan kecerdasan manusia, contohnya proses pengambilan keputusan., pemecahan masalah, serta adaptasi terhadap situasi tertentu. Oleh karena itu, AI memungkinkan mesin menjalankan tugas-tugas yang dulunya hanya bisa diselesaikan oleh manusia.



Gambar 2.2 Ilustrasi AI

2.2.2 Deep Learning

Deep Learning, bekerja dengan memanfaatkan struktur lapisan-lapisan neural network untuk mengolah data dan menghasilkan pemahaman yang lebih kompleks terhadap informasi tersebut. Istilah "deep" pada Deep Learning merujuk pada penggunaan lapisan-lapisan representasi yang

disusun secara berurutan. Deep Learning umumnya melibatkan jaringan yang terdiri dari puluhan hingga ratusan lapisan, di mana setiap lapisan secara otomatis mempelajari pola-pola dari data latih yang diberikan. Setiap lapisan dalam model ini mampu mengekstraksi fitur yang lebih abstrak dan kompleks dari data seiring dengan bertambahnya kedalaman jaringan.

Dalam Deep Learning, struktur pembelajaran disebut Neural Network, yang terdiri dari sejumlah lapisan yang saling berurutan dan tersusun secara bertingkat. Neural Networks terinspirasi oleh struktur otak manusia dalam bidang neurobiologi, terutama dalam hal kemampuan otak untuk memahami dan memproses informasi. Meskipun konsep ini terinspirasi dari cara kerja otak, model Deep Learning tidak sepenuhnya meniru mekanisme biologis otak manusia. Sejauh ini, tidak terdapat temuan ilmiah yang mendukung bahwa otak manusia bekerja dengan cara yang sama seperti model Deep Learning yang ada saat ini (Chollet, 2018).

Gambar 2.3 Ilustrasi Deep Learning

2.2.3 Voice Recognition

Teknologi ini beroperasi dengan mengubah suara yang diterima menjadi sinyal digital, yang kemudian dicocokkan dengan pola- pola tertentu yang telah tersimpan dalam basis data. Voice recognition dapat diaplikasikan dalam berbagai bidang, seperti perintah suara, asisten virtual, hingga pengenalan melodi dari humming yang dihasilkan oleh manusia. Sistem ini semakin banyak digunakan karena memberikan pengalaman interaksi yang lebih alami dan intuitif antara manusia dan mesin.

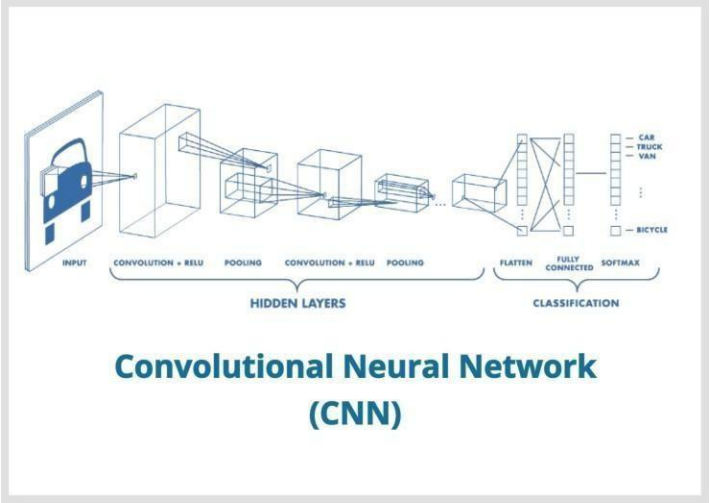
Dalam sistem pendeteksi lagu berbasis humming, teknologi voice recognition memiliki peran yang signifikan dalam mengidentifikasi suara humming dari pengguna. Suara humming yang direkam diubah menjadi sinyal digital, yang kemudian dianalisis untuk menemukan pola melodi yang khas. Pola melodi ini dibandingkan dengan pola melodi dari lagu-lagu yang tersimpan di dalam basis data. Sistem ini dirancang untuk mengenali lagu meskipun inputnya hanya berupa humming tanpa lirik, yang menimbulkan tantangan tersendiri dibandingkan dengan pengenalan suara berbasis kata.

Gambar 2.4 Gambaran Voice Recognition

2.2.4 Convolutional Neural Network

Efektif dalam pengenalan suara, dengan kemampuan ekstraksi fitur otomatis dari data input. Dalam implementasinya, sinyal audio diubah menjadi spektrogram dan diproses melalui lapisan CNN (konvolusional, pooling, fully connected). Metode ini unggul dalam akurasi dan ketahanan terhadap variasi ucapan.

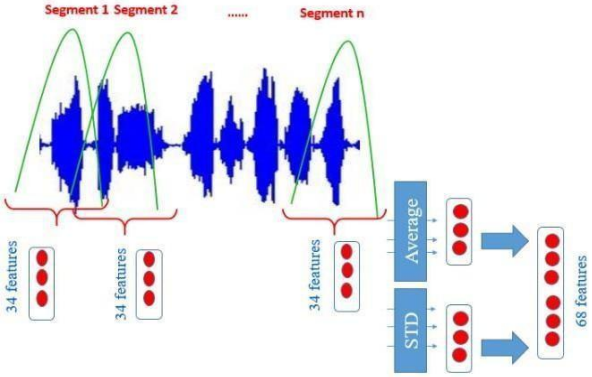
CNN juga digunakan untuk pengklasifikasian berbagai jenis suara, yang mengekstrak fitur suara menjadi bentuk visual (heatmap) untuk diproses lebih lanjut oleh CNN.



Gambar 2.5 Struktur CNN

2.2.5 *Feature Extraction*

Proses yang memuat ciri-ciri penting dari data untuk keperluan analisis lebih lanjut. Dalam konteks pengenalan suara, seperti pada sistem deteksi lagu dari input humming, data mentah berupa sinyal audio diolah untuk memperoleh atribut-atribut relevan, seperti pola frekuensi, amplitudo, dan durasi. Proses ini bertujuan untuk mereduksi kompleksitas data asli, sekaligus mempertahankan informasi esensial yang diperlukan untuk pengenalan pola.



Gambar 2.6 Ilustrasi Ekstraksi Fitur

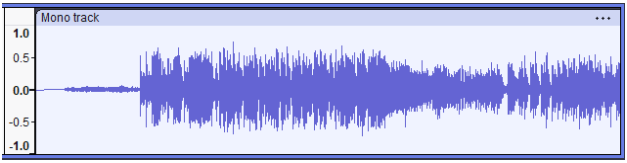
Ekstraksi fitur yang tepat sangat penting karena dapat mengurangi kebisingan dan informasi yang tidak relevan, sehingga meningkatkan akurasi dan efisiensi model dalam proses pengenalan suara atau musik.

2.2.6 Spektogram

Spektogram adalah representasi visual dari spektrum frekuensi sebuah sinyal suara yang bervariasi terhadap waktu. Spektogram menampilkan kepadatan spektral (density spectral) yang diwakili oleh intensitas warna, sehingga memudahkan analisis distribusi energi pada frekuensi tertentu selama periode waktu tertentu. Dalam pengolahan sinyal digital, khususnya pada sistem pendeteksi lagu berbasis *Convolutional Neural Network* (CNN), spektogram berperan penting sebagai fitur masukan. Pola frekuensi yang dihasilkan oleh humming manusia divisualisasikan dalam bentuk spektogram untuk menangkap karakteristik frekuensi dan energi yang relevan. Hal ini memungkinkan jaringan saraf tiruan untuk mempelajari pola-pola suara yang kemudian dapat digunakan untuk mengenali dan mengklasifikasikan lagu.

Spektogram juga digunakan untuk menganalisis perbedaan karakteristik antara sinyal suara asli dan sinyal yang telah melalui proses transformasi. Dalam penelitian ini, penggunaan metode transformasi

derivative gelombang glotal menyebabkan peningkatan ketajaman spektral pada sinyal hasil konversi dibandingkan dengan sinyal asli. Parameter seperti Open Quotient (OQ), Speed Quotient (SQ), serta pitch dan formant, mempengaruhi hasil konversi suara sehingga terjadi perubahan intensitas spektral meskipun bentuk dan durasi sinyal tidak berubah. Penggunaan spektogram dalam sistem deteksi lagu melalui humming akan memungkinkan analisis yang lebih mendalam terhadap pola frekuensi, yang krusial untuk meningkatkan akurasi model dalam mengenali lagu berdasarkan input humming.



Gambar 2.7 Representasi Spektogram

2.2.7 Mel Spectogram dB

Setelah proses ekstraksi fitur dan terdapar representasi visual dari frekuensi sinyal suara, maka data diproses menjadi bentuk satuan mel agar CNN dapat menganalisis pola unik atau khas dari data audio.

2.2.8 Augmentasi

Augmentasi data adalah proses yang dilakukan untuk meningkatkan kualitas dan variasi dataset dengan menambahkan data tambahan yang dihasilkan dari data awal. Dalam konteks sistem ini, augmentasi data melibatkan modifikasi pada input audio, seperti perubahan kecepatan, penambahan noise, atau pergeseran pitch, untuk menciptakan variasi suara yang berbeda namun tetap relevan.

2.2.9 Python

Bahasa pemrograman yang karakteristik utamanya adalah sintaks yang sederhana dan intuitif, sehingga memudahkan pengguna, terutama pemula, dalam mempelajarinya. Dukungan pustaka yang luas serta

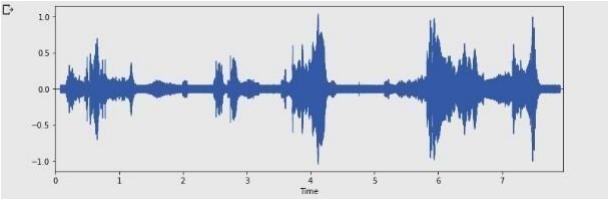
komunitas yang aktif membuat Python menjadi bahasa pilihan untuk banyak pengembang dan peneliti dalam berbagai bidang teknologi, termasuk machine learning dan deep learning.

Gambar 2.8 Logo Python

2.2.10 Librosa

Librosa adalah library Python yang dirancang untuk analisis musik dan audio, yang menyediakan berbagai fungsi untuk ekstraksi fitur audio. Library ini memungkinkan pengguna untuk melakukan pemrosesan sinyal audio dengan cara yang efisien dan intuitif, memudahkan analisis sinyal untuk keperluan penelitian di bidang musik dan pengenalan suara. Salah satu fitur utama dari Librosa adalah kemampuannya untuk mengekstrak berbagai jenis fitur dari sinyal audio, seperti Mel Frequency Cepstral Coefficients (MFCCs) dan melspectrogram. MFCCs, yang merupakan salah satu fitur paling umum digunakan dalam pengenalan suara, memungkinkan representasi suara yang lebih sesuai dengan persepsi manusia terhadap frekuensi, sehingga meningkatkan akurasi dalam proses klasifikasi dan pengenalan suara (Anggeli et al., 2021).

Dalam konteks penelitian yang menggunakan Librosa, library ini memungkinkan peneliti untuk melakukan ekstraksi fitur audio dengan mudah, seperti penghitungan MFCC dan melspectrogram. Dengan Librosa, pengguna dapat memvisualisasikan sinyal audio dalam bentuk grafik dan melakukan analisis yang mendalam terhadap karakteristik suara.

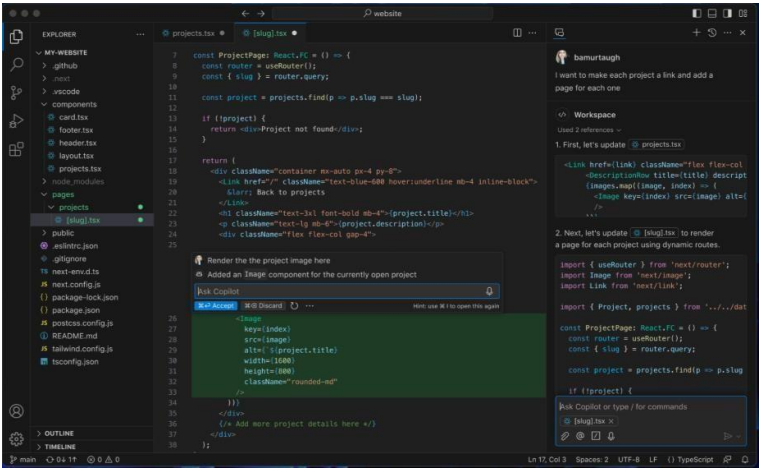


Gambar 2.9 Sinyal Audio Berbentuk Visual

2.2.11 Visual Studio Code

Aplikasi ini dirancang untuk menyediakan lingkungan pengembangan yang fleksibel dan cepat, sehingga dapat diandalkan oleh para pengembang dalam berbagai proyek perangkat lunak. Ketersediaannya yang luas membuatnya menjadi pilihan populer di kalangan programmer karena dapat digunakan pada berbagai perangkat dengan sistem operasi yang berbeda.

Kemampuan VS Code dapat diperluas dengan memasang berbagai plugin melalui marketplace yang disediakan, dengan fleksibilitas ini, VS Code mampu menjadi alat yang serba guna dalam berbagai kebutuhan pengembangan perangkat lunak, baik untuk pemrograman berbasis web maupun aplikasi desktop.



Gambar 2.10 Tampilan Visual Studio Code

2.2.12 Flask

Flask pada Python, yaitu *framework* ringan yang digunakan untuk membangun aplikasi web dengan cepat dan mudah. Dibuat dengan bahasa Python, Flask memungkinkan pengembang membuat situs web atau API dengan struktur yang sederhana dan fleksibel, tanpa memerlukan konfigurasi rumit seperti framework lain. Di dalam flask ini kita bisa memanggil file seperti html atau css yang dapat digunakan sebagai tampilan website.

2.2.13 Tensorflow

TensorFlow adalah sebuah open-source library yang digunakan untuk komputasi numerik dan pembelajaran mesin. Library ini menyediakan antarmuka untuk membangun dan melatih model machine learning, seperti Convolutional Neural Networks (CNN), yang banyak digunakan untuk berbagai aplikasi pengenalan pola, termasuk pengenalan pembicara. (Hourri et al., 2021) TensorFlow digunakan bersama dengan Keras untuk merancang dan mengevaluasi model CNN yang mampu mengekstraksi fitur suara secara langsung dari sinyal audio.

Gambar 2.11 Logo Tensorflow

2.2.14 Numpy

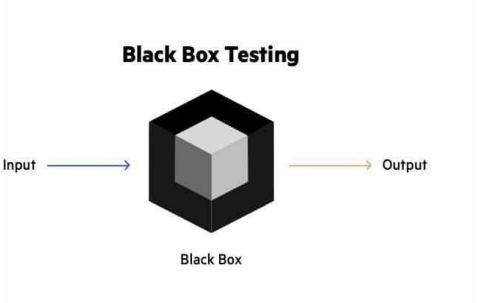
NumPy adalah pustaka (library) Python yang digunakan untuk melakukan perhitungan numerik secara efisien. Dengan NumPy, kita bisa mengolah data dalam bentuk array multidimensi, serta menjalankan operasi matematika, statistik, dan aljabar linear dengan lebih cepat dibandingkan menggunakan struktur data standar Python seperti list.

Gambar 2.12 Logo Numpy

2.2.14 Black Box Testing

Pengujian perangkat dengan tujuan memeriksa fungsionalitas sistem tanpa memeriksa bagian dalam atau kodenya. Tujuannya untuk menemukan kesalahan pada fungsi, antarmuka, struktur data, dan performa sistem hanya berdasarkan masukan dan keluaran.

Teknik yang umum digunakan dalam black box testing adalah equivalence partitions, yang membagi masukan ke dalam kelompok-kelompok tertentu. Ini memungkinkan pengujian untuk lebih efisien dengan memeriksa validitas data input, sehingga dapat mengidentifikasi dan memperbaiki kesalahan sebelum sistem digunakan secara luas.



Gambar 2.13 Ilustrasi Black Box

2.2.15 Confusion Matrix

Confusion Matrix digunakan sebagai alat evaluasi yang menampilkan data dalam bentuk tabel matriks, sebagaimana terlihat pada gambar berikut.

	Prediksi	
Aktual	TRUE	FALSE
TRUE	TP	FP
FALSE	FN	TN

Gambar 2.14 Tabel Matrix Untuk Pengujian

Melihat seberapa baik sistem dalam mengenali data dengan benar adalah tujuan dari metode ini. Tabel ini berisi empat jenis hasil. Pertama, saat sistem bisa mengenali data yang memang benar. Kedua, jika sistem juga benar dalam menyatakan data itu salah. Namun terkadang sistem menyangka data salah padahal seharusnya benar. Sebaliknya, jika sistem menyangka data itu benar padahal seharusnya salah. Empat jenis hasil ini membantu mengevaluasi kinerja sistem secara menyeluruh.

Terdapat 4 hasil pengujian confusion matix yaitu, *accuration, precision, recall dan F1-Score.*

2.2.15.1 Accuracy

Akurasi mengukur seberapa banyak prediksi yang benar dari keseluruhan prediksi yang dihasilkan oleh model. Nilai akurasi dihitung menggunakan rumus berikut:

$$\text{Accuracy} = \frac{TP_{\text{total}}}{\text{Jumlah Data}}$$

TP atau True Positive menunjukkan banyaknya data yang diklasifikasikan secara akurat ke dalam kelas yang benar.

2.2.15.2 *Precision*

Untuk mengetahui sejauh mana model melakukan prediksi yang tepat terhadap suatu kelas, digunakan ukuran *precision*. Berikut adalah rumusnya:

$$\text{Precision} = \frac{TP}{TP + FP}$$

False Positive (FP) merujuk pada jumlah data dari kelas lain yang keliru diprediksi sebagai bagian dari kelas ini.

2.2.15.3 *Recall*

Recall menunjukkan kemampuan model dalam mengenali seluruh data yang benar-benar berasal dari satu kelas. Berikut adalah rumus untuk menghitung recall:

$$\text{Recall} = \frac{TP}{TP + FN}$$

FN atau False Negative merupakan kasus di mana data dari kelas tertentu salah diklasifikasikan ke dalam kelas lain.

2.2.15.4 *F1-Score*

Sebagai ukuran evaluasi, *F1-score* mengombinasikan *precision* dan *recall* dalam bentuk rata-rata harmonik untuk memberikan gambaran kinerja model secara seimbang. Rumusnya adalah:

$$\text{F1-score} = \frac{2 \cdot (\text{Precision} \cdot \text{Recall})}{\text{Precision} + \text{Recall}}$$

Jika *Precision* dan *Recall* tingi, maka F1-Scorenya juga akan otomatis tinggi.