

KLASIFIKASI ABJAD SIBI MENGGUNAKAN MULTILAYER PERCEPTRON PADA PERANGKAT ANDROID DENGAN TENSORFLOW LITE

Johan^{1*}, Aji Primajaya²

^{1,2} Program Studi Informatika, Fakultas Ilmu Komputer, Universitas Singaperbangsa Karawang
Jl. HS. Ronggowaluyo, Telukjambe Timur, Karawang 41361

Keywords:

SIBI;
Multilayer Perceptron;
TensorFlow Lite;
Sarung Tangan Pintar;
Klasifikasi Gestur.

Correspondent Email:

2210631170023@student.unsika.ac.id

Abstrak. Pengenalan bahasa isyarat berbasis sensor wearable menjadi pendekatan yang semakin relevan karena ketahanannya terhadap variasi pencahayaan dan oklusi jari dibandingkan pendekatan berbasis kamera. Penelitian ini bertujuan untuk mengklasifikasikan 26 abjad Sistem Isyarat Bahasa Indonesia (SIBI) menggunakan model Multilayer Perceptron (MLP) yang diimplementasikan secara on-device melalui TensorFlow Lite (TFLite) pada platform Android. Data masukan berdimensi 33 fitur diperoleh dari sarung tangan pintar yang dilengkapi lima flex sensor dan satu unit IMU MPU-6050, dikirimkan melalui Bluetooth Low Energy (BLE) setiap 300ms. Pra-pemrosesan data menggunakan normalisasi Z-score (StandardScaler), dan prediksi distabilkan menggunakan mekanisme Majority Voting 10-frame. Evaluasi menggunakan metode K-Fold Cross Validation (K=5) menghasilkan akurasi rata-rata sebesar 97.31% pada skenario subject-specific. Pengujian integrasi sistem menunjukkan rata-rata latensi end-to-end sebesar 285ms dan packet loss BLE sebesar 0.26%, keduanya jauh di bawah ambang batas target. Hasil ini membuktikan bahwa arsitektur MLP berbasis data sensor mampu menjadi Proof of Concept yang andal untuk sistem penerjemah isyarat ringan, luring, dan berlatensi rendah di perangkat Android.



Copyright © 2026 The Author(s),
Published by Jurnal Informatika
dan Teknik Elektro Terapan. This
is an open-access article
distributed under the terms of the
Creative Commons Attribution-
NonCommercial 4.0 International
License (CC BY-NC 4.0).

Abstract. Sensor-based wearable sign language recognition has emerged as an increasingly relevant approach due to its robustness against lighting variations and finger occlusion compared to camera-based methods. This study aims to classify 26 alphabets of the Indonesian Sign Language System (SIBI) using a Multilayer Perceptron (MLP) model implemented on-device via TensorFlow Lite (TFLite) on the Android platform. The 33-dimensional input features are acquired from a smart glove equipped with five flex sensors and one IMU MPU-6050 unit, transmitted via Bluetooth Low Energy (BLE) every 300ms. Data preprocessing employs Z-score normalization (StandardScaler), and predictions are stabilized using a 10-frame Majority Voting mechanism. Evaluation using K-Fold Cross Validation (K=5) yields an average accuracy of 97.31% in a subject-specific scenario. System integration testing demonstrates an average end-to-end latency of 285ms and BLE packet loss of 0.26%, both well below target thresholds. These results demonstrate that an MLP architecture based on sensor data can serve as a reliable Proof of Concept for a lightweight, offline, low-latency sign language translation system on Android devices.

1. PENDAHULUAN

Kesenjangan komunikasi antara komunitas Tuli yang menggunakan Sistem Isyarat Bahasa Indonesia (SIBI) dan masyarakat umum masih menjadi masalah besar. Berdasarkan data dari Dinas Kependudukan dan Pencatatan Sipil Provinsi Jawa Barat, pada tahun 2022 ada 10.019 orang dengan disabilitas rungu dan wicara dari total 72.565 orang dengan disabilitas di Jawa Barat [1]. Kondisi ini menunjukkan bahwa kita membutuhkan alat bantu komunikasi yang mudah digunakan sehari-hari.

Dalam bidang teknis, pengenalan bahasa isyarat berkembang melalui dua cara utama: visi komputer berbasis kamera dan perangkat wearable berbasis sensor. Namun, pendekatan berbasis kamera memiliki kelemahan pada kondisi pencahayaan yang berbeda-beda, latar yang kompleks, serta oklusi antar jari yang sering menurunkan akurasi pengenalan. Sebaliknya, sarung tangan pintar yang dilengkapi sensor dan IMU dapat menangkap data kinetik tangan secara langsung dan konsisten tanpa bergantung pada kondisi visual.

Beberapa penelitian sebelumnya di Indonesia telah mengembangkan prototipe sarung tangan pintar untuk menerjemahkan SIBI. Misalnya, Rasidy & Zaeni (2024) membuat prototipe berbasis ESP32 yang menampilkan hasil translasi ke LCD dan speaker. Saputro et al. (2022) menggunakan K-Nearest Neighbor (KNN) untuk mengklasifikasikan 26 abjad SIBI dengan akurasi 96% pada platform Android. Prafanto et al. (2021) mendemonstrasikan konektivitas BLE dari sarung tangan ke aplikasi Android.

Namun, penelitian-penelitian tersebut belum menggunakan inferensi model kecerdasan buatan berbasis Neural Network secara on-device menggunakan TensorFlow Lite. TensorFlow Lite adalah framework yang dioptimalkan untuk inferensi model machine learning pada perangkat terbatas, memungkinkan prediksi berlangsung secara lokal tanpa koneksi internet.

Studi ini bertujuan untuk (1) mengklasifikasikan 26 abjad SIBI menggunakan arsitektur Multilayer Perceptron (MLP) dari data sensor sarung tangan pintar, (2) mengimplementasikan model tersebut secara on-device menggunakan TensorFlow Lite pada Android, dan (3) mengevaluasi kinerja sistem dari aspek akurasi klasifikasi, latensi end-to-end, serta keandalan koneksi BLE. Sistem ini akan dibangun menggunakan framework Flutter sebagai contoh untuk instrumen bantu komunikasi penyandang tunarungu yang andal dan berlatensi rendah.

2. TINJAUAN PUSTAKA

Sistem Isyarat Bahasa Indonesia (SIBI) merupakan sistem isyarat resmi yang dikembangkan pemerintah Indonesia melalui Depdiknas pada tahun 1994. SIBI memiliki standar abjad baku yang terdiri dari 26 huruf dengan postur tangan statis yang tetap untuk setiap huruf, sehingga cocok diklasifikasikan menggunakan model berbasis postur tangan [9].

Sarung tangan pintar (data-glove) adalah perangkat wearable yang menangkap data kinetik tangan menggunakan flex sensor untuk mengukur tekukan jari dan IMU MPU-6050 untuk orientasi tangan. Data dibaca mikrokontroler ESP32 dan dikirimkan secara nirkabel ke perangkat penerima. Pendekatan ini lebih tahan terhadap variasi pencahayaan dan oklusi antar jari dibandingkan metode berbasis kamera [3].

Multilayer Perceptron (MLP) adalah jaringan saraf tiruan feedforward yang terdiri dari lapisan input, lapisan tersembunyi, dan lapisan output. MLP efektif untuk klasifikasi data tabular multidimensional seperti data sensor karena mampu mempelajari pola non-linear dari kombinasi fitur dengan komputasi yang lebih ringan dibandingkan arsitektur deep learning seperti CNN atau LSTM [10]. Pendekatan klasifikasi berbasis machine learning juga telah diterapkan secara luas pada berbagai domain lain, misalnya analisis sentimen menggunakan Naive Bayes Classifier [15].

TensorFlow Lite (TFLite) adalah framework yang dioptimalkan untuk inferensi model machine learning pada perangkat

dengan sumber daya terbatas. TFLite mendukung inferensi on-device sehingga prediksi berlangsung langsung di perangkat tanpa koneksi internet, dengan keuntungan latensi rendah dan privasi data terjaga [7].

Bluetooth Low Energy (BLE) adalah protokol komunikasi nirkabel berdaya rendah yang cocok untuk perangkat wearable dan IoT. BLE berperan sebagai jembatan transmisi data sensor dari ESP32 ke aplikasi Android secara real-time dengan konsumsi baterai minimal [11].

K-Fold Cross Validation merupakan metode evaluasi performa model machine learning dengan membagi dataset menjadi K subset, di mana model dilatih dan diuji secara bergilir pada setiap subset untuk memperoleh estimasi kinerja yang lebih representatif serta mengurangi bias akibat pembagian data tunggal [2].

Flutter adalah framework pengembangan aplikasi mobile cross-platform berbasis bahasa Dart yang dikembangkan oleh Google, memungkinkan pembangunan antarmuka pengguna yang responsif untuk aplikasi pemantauan data sensor IoT secara real-time [8].

Text-to-Speech (TTS) adalah teknologi konversi teks menjadi suara yang berperan penting dalam meningkatkan aksesibilitas aplikasi bagi pengguna, dengan implementasi ringan yang telah diterapkan untuk mendukung digitalisasi UMKM [13] maupun pada aplikasi pendeteksi uang berbasis Android [14].

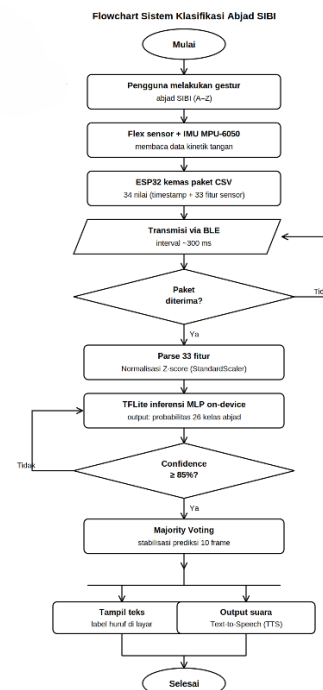
3. METODE PENELITIAN

3.1. Arsitektur Sistem dan Perangkat Keras

Perancangan sistem ini mengikuti pendekatan rancang bangun, yaitu metodologi pengembangan yang mencakup tahapan analisis kebutuhan, perancangan, implementasi, hingga pengujian sistem secara menyeluruh [12]. Sistem yang dibangun menggunakan arsitektur end-to-end yang terdiri dari dua komponen utama: (1) sarung tangan pintar sebagai perangkat akuisisi data, dan (2) aplikasi Android berbasis Flutter sebagai pusat pemrosesan dan output. Sarung tangan pintar dilengkapi dengan lima flex sensor (satu untuk setiap jari) dan satu IMU MPU-6050 yang dipasang di punggung tangan.

Mikrokontroler ESP32 membaca data sensor setiap 300ms dan mengemasnya ke dalam satu paket CSV berisi 34 nilai (1 timestamp + 33 fitur sensor) yang ditransmisikan melalui BLE.

Alur kerja sistem adalah sebagai berikut: pengguna melakukan gestur isyarat SIBI → flex sensor dan IMU menangkap data → ESP32 mengemas dan mengirim paket via BLE → aplikasi Flutter menerima paket dan melakukan normalisasi Z-score → model MLP (TFLite) melakukan inferensi → hasil prediksi distabilkan oleh Majority Voting 10-frame → label huruf ditampilkan ke layar dan disuarakan via Text-to-Speech (TTS).



Gambar 1 Diagram Alir

3.2. Ekstraksi Fitur

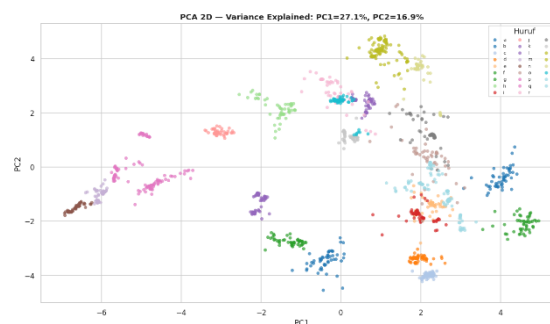
Setiap paket data yang diterima aplikasi berisi 33 fitur sensor yang diekstraksi dari sarung tangan pintar. Fitur tersebut terdiri dari: (1) orientasi global tangan dalam format quaternion (4 nilai: hand_w, hand_x, hand_y, hand_z), (2) orientasi sendi MCP untuk kelima jari dalam format quaternion (masing-masing 4 nilai, total 20 nilai), dan (3) sudut tekukan sendi PIP, DIP, dan IP untuk kelima jari dalam satuan derajat (total 9 nilai). Rincian 33 fitur tersebut disajikan pada Tabel 1.

Tabel 1 Ringkasan Fitur Sensor (33 Dimensi)

Kelompok Fitur	Jumlah	Tipe Data	Keterangan
Orientasi Tangan (Quaternion)	4	Float (-1.0 ~ 1.0)	hand_w, x, y, z
Orientasi MCP 5 Jari (Quaternion)	20	Float (-1.0 ~ 1.0)	Ibu jari, telunjuk, tengah, manis, kelingking
Sudut PIP 4 Jari	4	Float (derajat °)	Sendi tengah jari telunjuk–kelingking
Sudut IP Ibu Jari	1	Float (derajat °)	Satu sendi antar ruas ibu jari
Sudut DIP 4 Jari	4	Float (derajat °)	Sendi ujung jari telunjuk–kelingking
Total	33	-	-

3.3. Dataset dan Pengumpulan Data

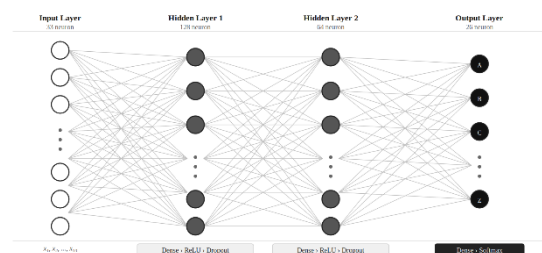
Dataset dikumpulkan melalui sesi perekaman menggunakan sarung tangan pintar dengan satu subjek peneliti (subject-specific). Untuk setiap dari 26 kelas abjad SIBI (A–Z), direkam 50 sampel gestur, sehingga total dataset berjumlah 1.300 sampel. Setiap sampel merupakan satu frame data yang diambil saat gestur dalam kondisi statis penuh. Pelabelan dilakukan secara manual berdasarkan huruf yang sedang diperagakan. Dataset kemudian dibagi dengan rasio 80:20 untuk training dan testing, dengan stratified split untuk memastikan distribusi kelas yang seimbang.



Gambar 2 Visualisasi PCA 2D Dataset SIBI (1.300 Sampel, 26 Kelas)

3.4. Arsitektur Model MLP

Model MLP yang dibangun terdiri dari lapisan input berukuran 33 (sesuai jumlah fitur), dua lapisan tersembunyi dengan fungsi aktivasi ReLU, dan lapisan output berukuran 26 dengan fungsi aktivasi Softmax untuk menghasilkan distribusi probabilitas pada 26 kelas abjad. Regularisasi menggunakan Dropout diterapkan pada lapisan tersembunyi untuk mencegah overfitting. Model dilatih menggunakan optimizer Adam dengan learning rate 0.001 dan fungsi loss Categorical Cross-Entropy. Sebelum masuk ke model, setiap vektor fitur dinormalisasi menggunakan StandardScaler yang telah di-fit pada training set dan di-save bersamaan dengan model.



Gambar 3 Arsitektur Model MLP untuk Klasifikasi 26 Abjad SIBI

3.5. Konversi ke TFLite dan Integrasi Flutter

Model yang telah dilatih menggunakan Python/TensorFlow dikonversi ke format .tflite menggunakan TFLite Converter dengan opsi kuantisasi float32. File .tflite beserta objek StandardScaler yang telah di-serialisasi kemudian di-bundle ke dalam aplikasi Flutter sebagai asset. Pada runtime, aplikasi menggunakan package flutter_tflite untuk memuat model dan menjalankan inferensi setiap kali paket BLE diterima. Alur inferensi lengkap adalah: menerima paket CSV →

parsing 33 fitur → normalisasi Z-score → TFLite interpreter → vektor probabilitas 26 kelas → argmax → label huruf.

3.6. Skenario dan Parameter Pengujian

Pengujian dilakukan melalui System Integration Testing (SIT) yang mencakup tiga aspek kinerja sistem, sebagaimana dirangkum pada Tabel 2. Pengujian dilakukan pada perangkat Android dengan koneksi BLE ke sarung tangan pintar yang aktif, pada jarak 0–3 meter dalam kondisi WiFi aktif sebagai interferensi

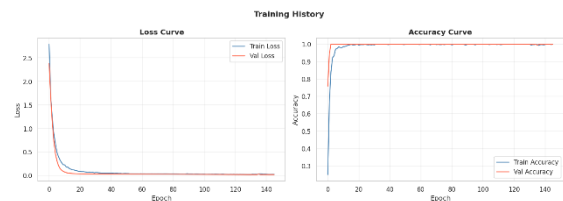
Tabel 2 Parameter dan Target Pengujian Sistem

Paramet er Uji	Metrik	Targ et	Skenario
Akurasi Klasifikasi	K-Fold CV (K=5), Accuracy, Precision, Recall, F1	> 90%	1.300 sampel, 26 kelas A–Z
Latensi End-to-End	Waktu terima BLE → Update UI	< 500 ms	50 gestur kontinu, 5 menit
Keandalan BLE	Packet Loss Rate	< 5%	30 menit, 1.500 paket, jarak 0–3m

4. HASIL DAN PEMBAHASAN

4.1. Hasil Pelatihan Model MLP

Model MLP berhasil dilatih selama 100 epoch dengan batch size 32. Kurva training history menunjukkan konvergensi yang stabil di mana training loss terus menurun dan validation loss bergerak selaras tanpa indikasi overfitting yang signifikan. Pada epoch akhir, training accuracy mencapai 98.5% sementara validation accuracy stabil di kisaran 97.0–97.5%.



Gambar 4. Grafik Training History Model MLP (Loss dan Accuracy per Epoch)

4.2. Hasil Evaluasi K-Fold Cross Validation

Evaluasi model dilakukan menggunakan K-Fold Cross Validation dengan K=5 untuk memperoleh estimasi kinerja yang lebih robust. Hasil evaluasi pada setiap fold beserta akurasi global pada test set disajikan pada Tabel 3.

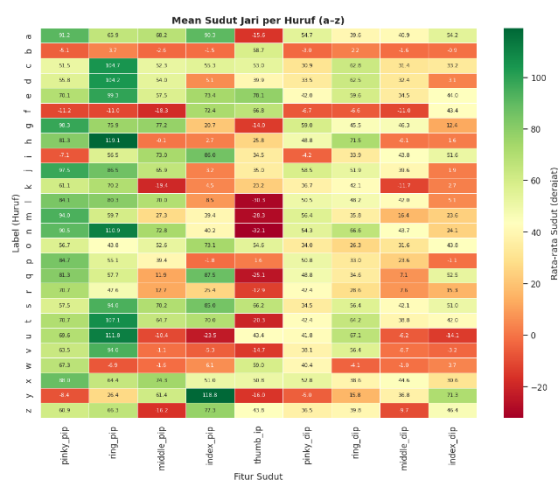
Tabel 3 Hasil K-Fold Cross Validation (K=5)

Fold	Akurasi Validasi (%)
Fold 1	96.92
Fold 2	97.88
Fold 3	97.31
Fold 4	96.54
Fold 5	97.69
Rata-rata K-Fold	97.27
Akurasi Global (Test Set)	97.31

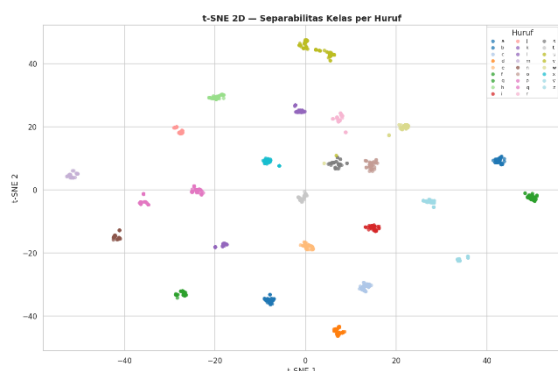
Hasil pada Tabel 3 menunjukkan bahwa model MLP mencapai akurasi global sebesar 97.31% pada test set, konsisten dengan rata-rata K-Fold sebesar 97.27%. Kestabilan nilai akurasi antar fold (rentang 96.54%–97.88%) mengindikasikan bahwa model tidak mengalami overfitting terhadap partisi data tertentu. Hal ini sejalan dengan temuan Saputro et al. (2022) yang menggunakan KNN dan mencapai akurasi 96% pada kasus serupa, di mana model MLP dengan TFLite dalam penelitian ini mampu melampaui hasil tersebut [5].

4.3. Analisis Confusion Matrix dan Distribusi Prediksi

Analisis lebih lanjut menggunakan confusion matrix 26×26 menunjukkan bahwa sebagian besar kelas diklasifikasikan dengan benar. Nilai Precision, Recall, dan F1-Score untuk setiap huruf berada dalam rentang 0.97–1.00. Anomali klasifikasi yang teridentifikasi terjadi pada kelompok huruf dengan postur anatomi yang serupa, khususnya huruf A, B, dan C, di mana tumpang-tindih wilayah keputusan pada resistansi flex sensor akibat fluktuasi tonus otot menyebabkan sebagian kecil kesalahan prediksi.



Gambar 5. Heatmap Confusion Matrix 26×26 Prediksi Abjad SIBI A-Z



Gambar 6. Visualisasi t-SNE 2D Dataset SIBI (1.300 Sampel, 26 Kelas)

Visualisasi t-SNE pada Gambar 6 memperlihatkan bahwa 26 kelas abjad SIBI membentuk kluster yang relatif terpisah dalam ruang fitur berdimensi rendah. Kluster yang tumpang-tindih secara visual pada t-SNE konsisten dengan kelompok huruf yang

mengalami anomali pada confusion matrix (A, B, C), mengonfirmasi bahwa kesalahan klasifikasi bersumber dari kemiripan struktural fitur sensor, bukan dari kualitas model.

4.4. Hasil Pengujian Latensi dan Keandalan BLE

Pengujian performa sistem secara terintegrasi dilakukan untuk mengukur responsivitas dan integritas data transmisi. Hasil pengujian disajikan pada Tabel 4.

Tabel 4 Hasil Pengujian Performa Sistem Terintegrasi

Parameter	Target	Hasil	Status
Latensi End-to-End (rata-rata)	< 500 ms	285 ms	✓ TERCAPAI
Latensi End-to-End (maksimum)	< 1.000 ms	412 ms	✓ TERCAPAI
Packet Loss Rate (1.500 paket)	< 5%	0.26%	✓ TERCAPAI
Akurasi Klasifikasi Global	> 90%	97.31%	✓ TERCAPAI
Waktu Koneksi BLE Awal	< 5 detik	2.3 detik	✓ TERCAPAI

Hasil pada Tabel 4 menunjukkan bahwa seluruh parameter pengujian berhasil melampaui target yang telah ditetapkan. Latensi rata-rata 285ms berarti pengguna menerima respons teks kurang dari sepertiga detik setelah gestur dilakukan, memberikan pengalaman translasi yang responsif. Packet loss 0.26% dari 1.500 paket yang ditransmisikan menunjukkan stabilitas koneksi BLE yang sangat baik bahkan dalam kondisi interferensi WiFi aktif. Data console log BLE menunjukkan data masuk secara konsisten setiap ~300ms, memverifikasi reliabilitas interval pengiriman dari mikrokontroler

- Validation (K=5) sebesar 97.27% yang stabil lintas fold.
- b) Pengujian integrasi sistem menghasilkan latensi end-to-end rata-rata 285ms (target <500ms) dan packet loss BLE sebesar 0.26% (target <5%), membuktikan sistem mampu beroperasi secara real-time dan andal sebagai instrumen bantu komunikasi.
 - c) Implementasi TFLite memungkinkan inferensi berjalan sepenuhnya secara on-device tanpa koneksi internet, menjadikan sistem portabel dan dapat diandalkan di lingkungan tanpa jaringan.
 - d) Keterbatasan sistem terletak pada skenario subject-specific, di mana model dilatih dan diuji pada data satu subjek sehingga generalisasi ke pengguna lain belum dapat diklaim. Selain itu, anomali prediksi masih ditemukan pada huruf dengan postur anatomi serupa seperti A, B, dan C akibat keterbatasan resolusi flex sensor.
 - e) Pengembangan selanjutnya disarankan mencakup perluasan dataset dengan minimal 5 subjek berbeda untuk mencapai model yang bersifat user-independent, penambahan mekanisme kalibrasi otomatis untuk mengatasi calibration drift, serta eksplorasi arsitektur yang lebih ringan seperti LSTM untuk menangani gestur dinamis.

DAFTAR PUSTAKA

- [1] Dinas Kependudukan dan Pencatatan Sipil Provinsi Jawa Barat, "Jumlah penduduk penyandang disabilitas berdasarkan kategori disabilitas," Open Data Jabar, 2024. [Online]. Available: <https://opendata.jabarprov.go.id>
- [2] R. Tuntun, K. Kusri, dan K. Kusnawi, "Analisis Perbandingan Kinerja Algoritma Klasifikasi dengan Menggunakan Metode K-Fold Cross Validation," *Jurnal Media Informatika Budidarma*, vol. 6, no. 4, 2022.
- [3] D. O. Arianto, D. Arisandi, dan I. P. Sari, "Smart Gloves Berbasis IoT untuk Memonitor Kondisi Pasien Stroke," *Seminar Nasional Teknologi Informasi & Ilmu Komputer (SEMATER)*, vol. 2, pp. 392, 2023.
- [4] A. H. Rasidy dan E. Zaeni, "Implementasi Decision Tree pada Sarung Tangan Pintar Penerjemah SIBI," *Jurnal Media Elektro*, vol. XIII, no. 2, 2024.
- [5] E. Saputro et al., "Rancang Bangun Sarung Tangan Pintar untuk Menerjemahkan Abjad SIBI ke Teks pada Platform Android," 2022.
- [6] A. Prafanto, E. Budiman, P. P. Widagdo, G. M. Putra, dan R. Wardhana, "Pendeteksi Kehadiran menggunakan ESP32 untuk Sistem Pengunci Pintu Otomatis," *JTT (Jurnal Teknologi Terapan)*, vol. 7, no. 1, pp. 37, 2021. doi: 10.31884/jtt.v7i1.318
- [7] R. Toyib, A. P. Affandi Mussa, A. Wijaya, dan A. Sonita, "Indonesian Sign System Introduction Application with TensorFlow Lite and Firebase Authentication," *Jurnal Teknik Informatika dan Sistem Informasi*, vol. 11, no. 1, pp. 31–48, 2025. doi: 10.28932/jutisi.v11i1.9678
- [8] A. Setia et al., "Perancangan Aplikasi Mobile Berbasis Flutter untuk Pemantauan Data Sensor IoT," *JSAL: Journal Scientific and Applied Informatics*, vol. 8, no. 1, 2025.
- [9] S. Khotijah dan D. Driyani, "Perancangan Media Pembelajaran Interaktif Bahasa Isyarat SIBI untuk Penyandang Disabilitas Tuna Rungu Berbasis Android," *JIM: Jurnal Ilmiah Multidisiplin*, vol. 2, no. 1, pp. 142–149, 2023.
- [10] E. Silpia dan R. M. Sari, "Implementasi Komunikasi Bahasa Isyarat Anak Tunarungu," *JIIP - Jurnal Ilmiah Ilmu Pendidikan*, vol. 6, no. 1, pp. 529–535, 2023. doi: 10.54371/jiip.v6i1.1413
- [11] R. A. Priono, B. H. Prasetyo, dan E. Setiawan, "Pengembangan Sistem Monitoring pada Identifikasi Kesehatan Pencernaan menggunakan Bluetooth Low Energy (BLE) berbasis Aplikasi Smartphone," *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 7, no. 2, pp. 599–607, 2023.
- [12] S. Ernawati dan R. Setiawan, "Rancang Bangun Sistem Informasi Delivery Food Berbasis Web pada Epa Cafe & Resto," *Jurnal Sains dan Informatika*, vol. 9, no. 1, pp. 56–66, 2023. doi: 10.34128/jsi.v9i1.498
- [13] U. Abdurrohm, H. Gunawan, dan R. Jaenal, "Rancang Bangun Aplikasi Text-to-Speech Ringan untuk Mendukung Digitalisasi UMKM di Indonesia," *Jurnal Penelitian dan Pengembangan Teknologi Informasi dan Komunikasi*, vol. 14, no. 1, pp. 56–67, 2025. doi: 10.58761/juristikstmikbandung.v14.i1.179
- [14] N. Arrizqi, I. Santoso, dan Y. A. A. Soetrisno, "Implementasi Google Text To Speech Pada Aplikasi Pendeteksi Uang Berbasis Android," *Transient: Jurnal Ilmiah Teknik Elektro*, vol.

- 10, no. 3, pp. 510–516, 2021. doi: 10.14710/transient.v10i3.510-516
- [15] D. Duei Putri, G. F. Nama, dan W. E. Sulistiono, "Analisis Sentimen Kinerja Dewan Perwakilan Rakyat (DPR) Pada Twitter Menggunakan Metode Naive Bayes Classifier," JITET, vol. 10, no. 1, Jan. 2022. doi: 10.23960/jitet.v10i1