

# Rancang Bangun Aplikasi Mobile Pengenalan Bahasa Isyarat Berbasis Machine Learning (Studi Kasus: SLB Putra Mandiri Surabaya)

1<sup>st</sup> Nafis Difaudin

Sistem Informasi

Telkom University

Surabaya, Indonesia

[nafisd@student.telkomuniversity.ac.id](mailto:nafisd@student.telkomuniversity.ac.id)

2<sup>nd</sup> Mochamad Nizar Palefi Ma'ady

Sistem Informasi

Telkom University

Surabaya, Indonesia

[mnizarpm@telkomuniversity.ac.id](mailto:mnizarpm@telkomuniversity.ac.id)

3<sup>rd</sup> Purnama Anaking

Sistem Informasi

Telkom University

Surabaya, Indonesia

[purnamaanaking@telkomuniversity.ac.id](mailto:purnamaanaking@telkomuniversity.ac.id)

*Komunikasi merupakan kebutuhan dasar manusia, termasuk bagi penyandang tuna rungu dan tuna wicara. Penelitian ini dilakukan di SLB Putra Mandiri Surabaya, yang menunjukkan bahwa masih banyak siswa belum menguasai bahasa isyarat SIBI dengan baik. Untuk menjawab permasalahan tersebut, dikembangkan aplikasi mobile pembelajaran dan penerjemah bahasa isyarat berbasis machine learning menggunakan metode prototype. Metode ini dipilih karena memungkinkan pengembangan yang fleksibel dan berbasis umpan balik pengguna. Fitur utama aplikasi ini adalah kamera penerjemah real-time yang menggunakan model Convolutional Neural Network (CNN) untuk mengenali gestur tangan. Ekstraksi fitur dilakukan menggunakan MediaPipe, sementara pelatihan model menggunakan dataset gesture SIBI dari Kaggle serta input manual. Model dilatih menggunakan TensorFlow dan diimplementasikan melalui API Flask yang terintegrasi ke aplikasi Flutter. Selain fitur penerjemah, aplikasi ini menyediakan modul pembelajaran berupa kamus alfabet isyarat, video pembelajaran, dan kuis edukatif. Hasil pengujian menunjukkan aplikasi mampu mengenali gesture secara akurat serta meningkatkan pemahaman siswa terhadap SIBI. Hasil yang diharapkan adalah terciptanya sebuah aplikasi edukatif yang dapat meningkatkan keterampilan komunikasi siswa tunarungu, memudahkan proses belajar SIBI, dan memperkaya interaksi sosial mereka dengan lingkungan sekitar secara lebih inklusif.*

**Kata kunci—** SIBI, Machine learning, Flutter, CNN, Metode Prototype

## I. PENDAHULUAN

Komunikasi menjadi hak dasar setiap individu, termasuk bagi penyandang tuna wicara dan tuna rungu. Di Indonesia, Sistem Isyarat Bahasa Indonesia (SIBI) digunakan secara luas sebagai media komunikasi formal di lingkungan pendidikan [1]. Namun demikian, di SLB Putra Mandiri Surabaya, masih ditemukan kendala dalam penguasaan bahasa isyarat oleh para siswa, sehingga menyebabkan keterbatasan dalam interaksi sosial.

Data dari Mutiah et. al.[2] menyebutkan bahwa hingga 85% masyarakat mengalami kesulitan dalam berinteraksi dengan penyandang disabilitas, terutama tuna rungu dan tuna grahita. Hal ini menunjukkan perlunya edukasi serta media pembelajaran yang ramah pengguna bagi siswa berkebutuhan

husus agar tercipta komunikasi yang lebih efektif dan inklusif.

Perkembangan teknologi informasi dan komunikasi telah membuka peluang baru dalam meningkatkan aksesibilitas. Aplikasi mobile yang dapat mengenali dan menerjemahkan bahasa isyarat secara real-time menjadi solusi potensial untuk menjembatani komunikasi antara penyandang disabilitas dan masyarakat luas. Teknologi seperti Convolutional Neural Network (CNN), MediaPipe, dan TensorFlow memungkinkan pengembangan sistem pengenalan gestur tangan dengan akurasi tinggi [3], [4].

Beberapa penelitian terdahulu telah mengembangkan sistem serupa dengan metode machine learning, namun masih terbatas pada ruang lingkup bahasa atau platform tertentu [5], [6]. Oleh karena itu, penelitian ini bertujuan untuk mengembangkan aplikasi mobile edukatif yang dapat mengenali bahasa isyarat SIBI dengan pendekatan prototyping.

Tujuan dari penelitian ini adalah untuk mengembangkan aplikasi mobile yang mampu mengenali gestur alfabet bahasa isyarat SIBI secara real-time dan menyediakan fitur pendukung seperti kamus interaktif, video pembelajaran, serta kuis edukatif. Harapannya, aplikasi ini dapat meningkatkan pemahaman dan keterampilan komunikasi siswa tunarungu serta mendorong pembelajaran yang lebih inklusif.

## II. KAJIAN TEORI

### A. Bahasa Isyarat SIBI

Sistem Bahasa Isyarat Indonesia (SIBI) merupakan salah satu bentuk komunikasi visual yang digunakan oleh penyandang tunarungu untuk memfasilitasi interaksi sosial. SIBI menggunakan gerakan tangan, ekspresi wajah, dan gerakan tubuh yang disusun secara sistematis untuk merepresentasikan kosa kata dalam bahasa Indonesia[1]. Kegiatan komunikasi ini dimaksudkan untuk membantu penyandang tuna rungu memahami dan menyampaikan pesan secara efektif, khususnya dalam lingkungan pendidikan. Dalam penerapannya, SIBI dirancang mengikuti tata bahasa Indonesia sehingga penggunaannya lebih formal

dibandingkan Bahasa Isyarat Indonesia (BISINDO) yang natural dan kontekstual[1]. SIBI merupakan metode penting untuk meningkatkan keterampilan berbahasa dan interaksi sosial penyandang tunarungu, khususnya penyandang gangguan pendengaran berat, karena mendorong keterampilan komunikasi secara keseluruhan, khususnya komunikasi antara sinyal visual, gerakan bibir, dan tulisan.

### B. Convolutional Neural Network (CNN)

Convolutional Neural Network (CNN) merupakan algoritma machine learning yang dirancang khusus untuk mengolah data visual seperti gambar dan video dengan menggunakan operasi konvolusi untuk mengekstraksi fitur secara otomatis. CNN terdiri dari dua komponen utama, yaitu feature extraction dan classification layer. Pada tahap ekstraksi fitur, lapisan konvolusi mendeteksi elemen penting dalam gambar seperti tepi dan tekstur, sementara pooling layer berfungsi mengurangi dimensi data untuk efisiensi dan mencegah overfitting. Fitur yang telah diekstraksi kemudian diproses oleh lapisan fully connected untuk menghasilkan klasifikasi akhir. CNN dikenal memiliki kemampuan dalam mengenali pola visual yang kompleks secara efisien [3].

### C. Mediapipe

Mediapipe merupakan library open-source yang dikembangkan oleh Google dan digunakan untuk pemrosesan multimedia berbasis machine learning. Dalam konteks pengenalan gestur tangan, Mediapipe Hand Landmark mampu mendeteksi 21 titik koordinat(landmark) dari tangan secara real-time dan akurat, yang dapat digunakan sebagai input ke dalam model klasifikasi [4].

### D. Flutter

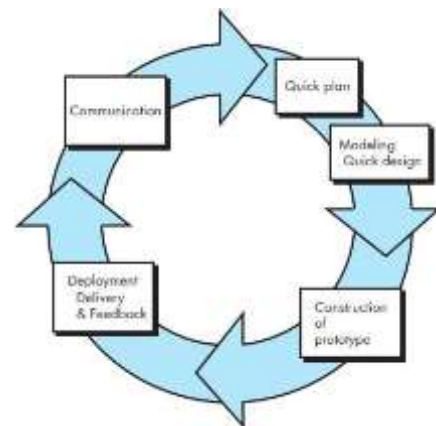
Flutter merupakan framework UI open-source dari Google yang digunakan untuk mengembangkan aplikasi mobile berbasis Android maupun iOS dengan satu basis kode. Flutter menawarkan fleksibilitas tinggi dalam pengembangan antar muka pengguna serta kompatibel dengan berbagai sistem operasi, sehingga cocok digunakan untuk proyek lintas platform[7].

### E. TensorFlow

TensorFlow merupakan platform open-source untuk komputasi numerik dan machine learning yang dikembangkan oleh Google. TensorFlow mendukung berbagai operasi matematis kompleks dan digunakan secara luas untuk pelatihan dan deployment model deep learning. Dalam Penelitian ini, TensorFlow digunakan untuk memabangun serta melatih model CNN yang mengelai gestur bahasa isyarat[8].

### F. Metode Prototype

Metode Prototype adalah pendekatan pengemabgan sistem yang berfokus pada pembuatan versi awal dari aplikasi yang dapat diuji oleh pengguna. Melalui proses iteratif, prototype dikembangkan berdasarkan umpan balik pengguna hingga memenuhi kebutuhan yang diinginkan[9].



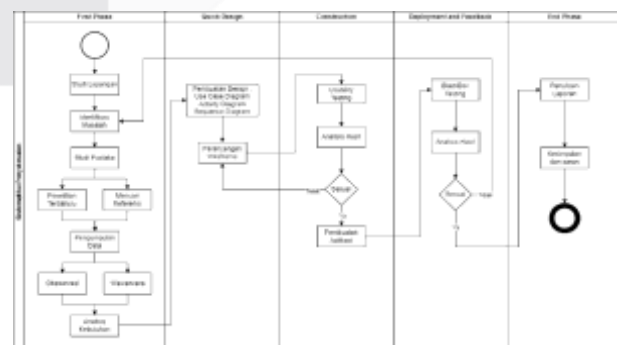
GAMBAR 1  
(METODE PROTOTYPE)

Metode Prototype memiliki beberapa tahapan dalam pengembangannya, berikut merupakan tahapan dalam metode prototype:

- Communication, memiliki tujuan untuk mengidentifikasi kebutuhan dari pengembangan aplikasi[9].
- Quick Plan, pada tahap ini perancang melakukan perencanaan cepat berdasarkan kebutuhan yang didapat pada tahap Communication[9].
- Modelling Quick Design, tim perancang akan membuat model desain, seperti UML atau pemodelan lain yang relevan[9].
- Construction of Prototype, perancang mulai memulai pembangunan perangkat lunak, yang bertujuan untuk mendapatkan feedback dari klien[9].
- Deployment and Feedback, pada tahap ini prototipe akan diserahkan kepada klien untuk mendapatkan feedback[9].

## III. METODE

Pada bab ini dijelaskan alur dari tahapan pengembangan aplikasi melalui Gambaran dari sistematika penyelesaian masalah. Pendekatan yang dilakukan dalam penelitian ini menggunakan metode prototype yang dijelaskan dalam rincian berikut.



GAMBAR 2  
(SISTEMATIKA PENYELESAIAN)

#### A. First Phase

Proses dimulai dengan tahapan untuk mengidentifikasi masalah yang dimiliki oleh pengguna ataupun masalah yang terlihat selama dilakukannya observasi secara langsung di SLB Putra Mandiri Surabaya. Setelah itu, dilakukan pengumpulan data yang relevan terkait masalah yang Tengah dialami oleh user selama observasi. Hasil analisis digunakan dalam tahap quick design untuk merumuskan rencana pengembangan sistem.

#### B. Quick Design

Setelah mendapatkan data yang diperlukan dari First Phase, selanjutnya dilakukan perancangan cepat yang berisikan perancangan data, interface serta model sistem. Model Sistem melibatkan pembuatan berbagai jenis diagram seperti use case diagram, activity diagram, serta sequence diagram. Use case diagram digunakan untuk menggambarkan interaksi antara pengguna dengan sistem, Activity diagram menggambarkan alur kerja sistem, dan Sequence diagram menunjukkan urutan proses antara objek dengan sistem. Tujuan dari adanya diagram ini untuk meningkatkan pemahaman perancang dengan adanya visualisasi hubungan sistem yang akan dibangun.

#### C. Construction of Prototype

Fase Construction of prototype memiliki alur kerja yang memiliki tahapan penting. Pada tahap pertama dilakukan feedback testing dari prototype yang telah dibuat, user akan melakukan evaluasi terhadap kegunaan, keberfungsian dan ketrampilan pengguna dalam menggunakan aplikasi. Hasil dari feedback testing kemudian dianalisa untuk mengidentifikasi kekurangan serta kesalahan dari prototype yang telah dibuat. Jika hasil pengujian menunjukkan bahwa prototype tidak sesuai dengan harapan atau membutuhkan perbaikan akan dilakukan perbaikan dari masalah yang telah ditemukan. Setelah perbaikan terselesaikan, tahapan berikutnya ialah pengembangan atau pembuatan aplikasi sesuai dengan prototype yang telah dibuat sebelumnya. Pada tahap ini, desain dan fungsionalitas telah ditentukan dalam prototype yang akan diimplementasikan kedalam aplikasi menggunakan framework Flutter.

#### D. Deployment

Fase deployment, delivery, dan feedback merupakan kelanjutan dari tahap construction of prototype, yang bertujuan untuk memastikan bahwa produk akhir telah siap digunakan oleh pengguna. Langkah awal pada fase ini adalah pelaksanaan usability testing, yaitu proses pengujian yang dirancang untuk mengevaluasi seberapa mudah dan efisien pengguna dapat berinteraksi dengan antarmuka aplikasi. Pengguna diminta menyelesaikan serangkaian tugas tertentu, dan dilakukan observasi terhadap perilaku serta respons mereka untuk mengidentifikasi hambatan yang mungkin muncul selama penggunaan. Hasil dari proses ini menjadi acuan untuk menyempurnakan aspek user experience (UX) dan antarmuka pengguna (UI) sebelum peluncuran aplikasi secara resmi.

Setelah proses usability testing, dilakukan unit testing untuk memastikan bahwa setiap modul atau komponen dalam sistem bekerja secara individual sesuai dengan fungsinya. Pengujian kemudian dilanjutkan dengan BlackBox testing, yang berfokus pada verifikasi fungsionalitas aplikasi berdasarkan masukan dan keluaran, tanpa mempertimbangkan struktur internal kode. Skenario

pengujian disusun untuk mencakup berbagai kemungkinan penggunaan, termasuk validasi input dan output, guna menjamin bahwa seluruh fitur telah berjalan sesuai dengan spesifikasi yang ditetapkan. Hasil dari pengujian ini menjadi dasar dalam proses evaluasi akhir dan penyempurnaan sebelum aplikasi diimplementasikan secara luas kepada pengguna akhir.

### IV. HASIL DAN PEMBAHASAN

Bab ini menjelaskan secara rinci mengenai proses pengembangan aplikasi mobile pengenalan bahasa isyarat. Proses mengacu pada pendekatan metode yang telah diterapkan yakni *prototype* yang berorientasi pada *feedback* dari pengguna, hal ini digunakan untuk memastikan sistem yang dikembangkan sesuai dengan kebutuhan dari pengguna akhir di SLB Putra Mandiri Surabaya.

#### A. Pengumpulan Data

Pada fase pertama dalam penelitian ini ialah dilakukannya pengumpulan data. Pada tahap pengumpulan data ini, data dikumpulkan dengan dua cara yaitu observasi dan wawancara. Observasi dilakukan di kelas khusus tuna wicara dan tuna rungu di SLB Putra Mandiri Surabaya, sedangkan wawancara dilakukan dengan salah satu guru yang mengajar kelas khusus tuna wicara dan tuna rungu untuk memperoleh informasi lebih lanjut mengenai kebutuhan dari aplikasi yang akan dikembangkan. Hasil dalam pengumpulan data didapatkan kebutuhan *user* sesuai dengan tabel berikut:

TABEL 1  
(KEBUTUHAN USER)

Fitur	Deskripsi
Kamus bahasa isyarat	Fitur ini berfungsi sebagai media referensi bagi pengguna untuk mempelajari berbagai kosakata dalam bahasa isyarat. Kamus dilengkapi dengan gambar dan video yang memperlihatkan gerakan isyarat secara jelas.
Video Pembelajaran	Fitur ini menyediakan konten pembelajaran dalam bentuk video pembelajaran yang dirancang untuk membantu anak-anak untuk memahami materi bahasa isyarat secara menyenangkan.
Tebak bahasa isyarat	Fitur interaktif berupa permainan sederhana menyenangkan yang bertujuan untuk meningkatkan pemahaman dan kemampuan pengguna dalam mengenali isyarat melalui metode kuis atau tebak-tebakan.
Kamera penerjemah bahasa isyarat	Fitur utama dalam aplikasi yang memungkinkan pengguna untuk menerjemahkan gerakan bahasa isyarat secara <i>real-time</i> menggunakan kamera pada perangkat seluler. Teknologi ini memanfaatkan pemrosesan menggunakan python untuk pengolahan data gambar yang di latih lalu di gunakan pada media seluler untuk mengenali gerakan tangan dan menampilkan hasil terjemahan dalam bentuk teks.

#### B. Perancangan Sistem

Perancangan sistem menggambarkan hasil dari tahapan pengembangan aplikasi, yang mencakup pemodelan sistem melalui Use Case Diagram, Activity Diagram, Serta Sequence Diagram. Selain itu tahap ini juga mencakup pengembangan dari machine learning juga perancangan antarmuka pengguna dan pembuatan prototipe sebagai penampilan awal sistem yang akan dikembangkan.





GAMBAR 3  
(USE CASE DIAGRAM)

*Use Case Diagram* disajikan pada gambar diatas menggambarkan berbagai fungsi yang dapat dilakukan oleh pengguna. Pengguna memiliki akses terhadap fitur fitur yang dikembangkan oleh aplikasi seluler yang dikembangkan seperti, membuka kamera penerjemah Bahasa isyarat, melihat kamus Bahasa isyarat, melihat video pembelajaran. Diagram ini memberikan Gambaran menyeluruh mengenai alur interaksi pengguna dengan sistem secara detail dan terstruktur.

#### C. Perancangan Machine learning

Model machine learning yang dikembangkan merupakan model klasifikasi berbasis Convolutional Neural Network (CNN) yang dilatih untuk mengenali gestur alfabet A-Z. Model ini menerima input berupa koordinat 21 titik tangan (landmark) hasil ekstraksi dari MediaPipe. Setiap titik terdiri dari koordinat x, y, dan z sehingga total terdapat 63 fitur masukan.

TABEL 2  
(DATA LANDMARK LABEL)

62	61	60	59	...	0	Label
0.00648	0.63481	0.31804	-0.0332	...	-0.0091	A
-0.0936	0.30509	0.35812	-0.0809	...	-0.0365	B
-0.0307	0.52403	0.72816	-0.0285	...	-0.0064	C
-0.0929	0.593756	0.67896	-0.0919	...	-0.0685	D
-0.1181	0.445238	0.4775	-0.1062	...	-0.0398	E
.						
.						
.						
-0.0193	0.752096	0.22065	-0.0197	...	-0.0135	Z

Dataset yang digunakan terdiri dari lebih dari 2.000 data gestur tangan statis yang terbagi dalam 26 kelas huruf. Data dibagi menjadi 80% untuk pelatihan dan 20% untuk pengujian menggunakan fungsi `train_test_split`. Model dibangun dengan beberapa lapisan konvolusi, max pooling, dan diakhiri dengan lapisan dense.

Pelatihan dilakukan dengan 100–150 epoch menggunakan optimasi Adam dan fungsi kehilangan categorical crossentropy. Selama pelatihan, akurasi model meningkat secara konsisten dan menunjukkan stabilitas. Evaluasi model dilakukan menggunakan confusion matrix dan classification report, dengan hasil akurasi lebih dari 90%.

Implementasi sistem deteksi bahasa isyarat ini dilakukan melalui beberapa tahapan, mulai dari pengumpulan data citra tangan, ekstraksi menggunakan MediaPipe, pelatihan model CNN, hingga integrasi ke dalam aplikasi Flutter berbasis kamera secara real-time. Data gesture tangan diperoleh dari Kaggle, kemudian diproses untuk mengekstrak 21 titik koordinat tangan (x, y, z) menggunakan MediaPipe Hands. Hasil ekstraksi disimpan dalam format CSV, lalu dipisah menjadi fitur dan label sebelum dilatih menggunakan CNN. Arsitektur model terdiri dari 63 neuron input, dua hidden layer (128 dan 64 neuron), dropout, dan output layer softmax.

Model dikompilasi menggunakan optimizer Nadam dan dilatih selama 30 epoch. Setelah pelatihan selesai, model disimpan dalam format .h5. Integrasi ke Flutter dilakukan dengan memanfaatkan Flask sebagai API untuk menerima input gambar dari kamera, melakukan ekstraksi landmark, dan memprediksi gesture menggunakan model. Hasil prediksi dikirim kembali dan ditampilkan ke pengguna. Proses ini memungkinkan deteksi gesture secara langsung dan interaktif tanpa perangkat tambahan.

#### D. Tahap Prototype

Setelah tahap perancangan telah dilaksanakan, pneliti akan melanjutkan ke pembuatan prototype dari aplikasi yang dikembangkan, prototype memiliki 2 iterasi. Pada iterasi pertama peneliti membuaat desain high fidelity dan pengembangan prototype interaktif. Pada tahap ini juga dilakukan evalusai pengguna untuk melihat kesesuaian rancangan awal dengan kebutuhan pengguna.



GAMBAR 4  
(DESAIN HIGH FIDELITY)

Berikut merupakan contoh desain high fidelity yang dikembangkan, berikutnya akan dilakukan evaluasai pengguna untuk memastikan kesesuaian.

TABEL 3  
(EVALUASI PENGGUNA)

Desain	Feedback Pengguna	Kesimpulan
Home Screen	Tampilan sudah cukup bagus, namun perwarnaan tolong diperhatikan	Perbaikan
Kamus Screen	Tampilan sudah bagus, namun diperhatikan posisi dari Bahasa, serta tambahkan detail untuk tiap abjad	Perbaikan
Video Screen	Tampilan sudah bagus, simpel mudah dimengerti	Diterima
Camera Screen	Tampilan kurang menarik, agak lompat dari tema tema fitur sebelumnya tolong perbaiki tema dari tampilannya	Perbaikan

Setelah evaluasi dilakukan, proses dilanjutkan pada iterasi kedua yang berfokus pada penyesuaian dan revisi terhadap desain sesuai dari keinginan pengguna pada tahap evaluasi sebelumnya. Perbaikan ini bertujuan untuk menyempurnakan tampilan dan alur sistem agar mampu diterima oleh pengguna serta meningkatkan kenyamanan pengguna ketika penggunaan aplikasi nantinya.

E. Pengembangan Aplikasi

Tahapan Pembuatan aplikasi ini dilakukan dengan mengimplementasikan kode sesuai rancangan yang telah diterima oleh pihak terkait sebelumnya. Proses ini menggunakan Flutter untuk pengembangan tampilan depan dari aplikasi, Python *Flask* yang digunakan untuk fitur kamera penerjemah serta menggunakan public database yakni Supabase sebagai tempat penyimpanan data gambar dan data video yang diperlukan oleh aplikasi.



GAMBAR 5  
(HOME SCREEN)

Tampilan awal aplikasi dirancang agar ramah anak dan intuitif. Pengguna dapat memilih fitur utama seperti kamus, video, kuis, dan kamera penerjemah. Halaman ini telah dioptimalkan untuk berbagai ukuran layar perangkat Android.



GAMBAR 6  
(KAMUS SCREEN)

Menampilkan daftar alfabet bahasa isyarat SIBI lengkap dengan gambar animasi yang menarik. Fitur pencarian dan filter memungkinkan pengguna menemukan huruf tertentu dengan cepat. Setiap huruf dapat ditekan untuk melihat detail lebih lanjut.



GAMBAR 7  
(VIDEO SCREEN)

Memuat video edukatif dari YouTube yang dikurasi dengan hashtag khusus pembelajaran bahasa isyarat. Video dapat di-pause dan dimainkan kembali secara interaktif, mirip dengan pengalaman pengguna di platform YouTube.



GAMBAR 8  
(CAMERA SCREEN)

Fitur utama aplikasi ini memungkinkan deteksi gesture secara real-time menggunakan kamera depan maupun belakang. Siswa dapat mengubah arah kamera atau menyalakan flash sesuai kebutuhan. Gambar ditangkap dan dikirim ke server Flask, di mana sistem akan mengekstrak

landmark dan memprediksi gesture huruf. Output berupa huruf dan nilai confidence ditampilkan langsung di bawah kamera. Jika confidence > 0.9, huruf tersebut ditambahkan ke kolom kata. Fitur hapus juga tersedia jika pengguna ingin mengulang.

F. Black-Box Testing

Pada tahap ini dilakukan pengujian terhadap aplikasi yang telah dikembangkan menggunakan metode BlackBox testing. Pengujian dilakukan berdasarkan fitur yang tersedia dan dilaksanakan di SLB bersama murid dan guru sebagai end-user.

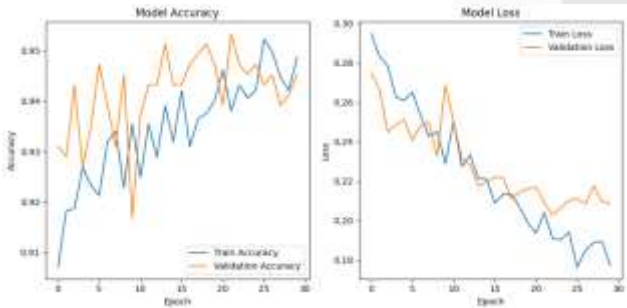
TABEL 4  
(BLACK-BOX TESTING)

No	Fitur	Test Case	Hasil Diharapkan	Hasil	Keterangan
1	Home Page	Mengakses halaman Home Page	Sistem menampilkan halaman yang berisi tampilan utama dari aplikasi	Valid	Pengujian dilaksanakan 2 kali dan hasilnya valid
2	Kamus Page	Mengakses Halaman Kamus	Sistem Menampilkan list dari data kamus	Valid	Pengujian dilaksanakan 2 kali dan hasilnya valid
3	Video Pembelajaran	Mengakses halaman Video Pembelajaran	Sistem menampilkan video dari youtube serta detail judul video	Valid	Pengujian dilaksanakan 3 kali dan hasilnya valid
4	Kamera Penerjemah	Menguji prediction dari kamera penerjemah	Sistem memberikan feedback prediction serta confidence dari hasil uji	Valid	Pengujian dilaksanakan 4 kali

G. Machine Learning Testing

1) Evaluasi Model

Model diuji dengan data training untuk melihat akurasi dan loss selama proses pelatihan. Hasil menunjukkan performa yang baik dan stabil pada sebagian besar kelas gesture.

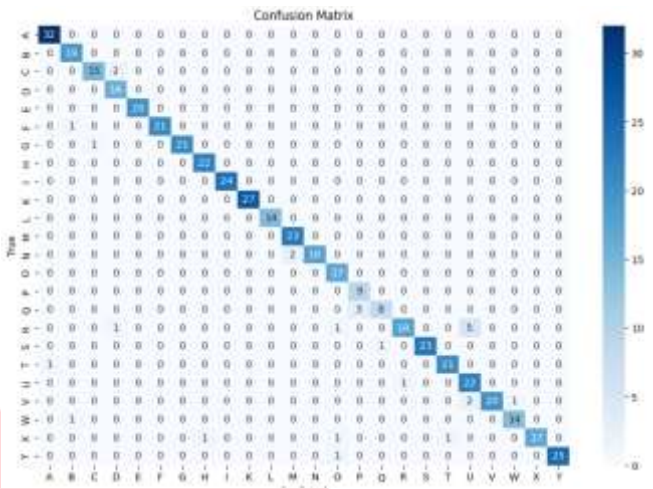


GAMBAR 9  
(MODEL LOSS AND ACCURACY)

2) Testing Model Pengujian Data Eksternal

Pengujian eksternal dilakukan dengan dataset SIBIKU untuk memverifikasi keakuratan model. Model membaca

gambar, memprediksi label, dan hasil divisualisasikan dalam bentuk confusion matrix serta classification report. Akurasi total mencapai 95%.



GAMBAR 10  
(CONFUSION MATRIX)

TABEL 5  
(CLASSIFICATION TABLE)

	Precision	Recall	Fi-score	Support
A	0.97	1.00	0.98	32
B	0.90	1.00	0.95	19
C	0.94	0.88	0.91	17
D	0.84	1.00	0.91	16
E	1.00	1.00	1.00	20
F	1.00	0.95	0.98	22
G	1.00	0.95	0.98	22
H	0.96	1.00	0.98	22
I	1.00	1.00	1.00	24
J	0.91	0.91	0.91	14
K	1.00	1.00	1.00	27
L	1.00	1.00	1.00	14
M	0.96	1.00	0.96	23
N	1.00	0.60	0.95	20
O	0.85	1.00	0.92	17
P	0.75	1.00	0.86	9
Q	0.89	0.73	0.80	11
R	0.95	0.72	0.82	25
S	1.00	0.96	0.98	24
T	0.95	0.95	0.95	22
U	0.76	0.96	0.85	23
V	1.00	0.87	0.93	23
W	0.93	0.93	0.93	15
X	1.00	0.85	0.92	19
Y	1.00	0.96	0.98	26
Z	0.93	0.93	0.93	16
Accuracy	-	-	0.95	493
Macro avg	0.94	0.94	0.94	493
Weighted avg	0.95	0.95	0.95	493

Beberapa gesture seperti E, I, K, dan L memiliki fl-score sempurna (1.00), sedangkan gesture P, Q, R, dan U menunjukkan penurunan performa akibat kemiripan visual dan kesalahan klasifikasi.

3) Pengujian Real Time

Uji dilakukan kepada 10 orang penguji, masing-masing 5 kali gesture A-Z, menghasilkan total 1.300 percobaan. Hasil menunjukkan akurasi rata-rata 95%. Gesture seperti I, K, B, dan A sangat akurat (>98%), sementara C, G, O, dan J kurang akurat (<70%).

Faktor yang memengaruhi: variasi posisi tangan, pencahayaan, gesture tidak konsisten, serta kemiripan visual antar gesture.

#### 4) Analisis Perbandingan Pengujian

Terdapat penurunan akurasi dari data statis ke real-time karena:

##### A. Variasi sudut pandang pengguna

Dalam pengujian *real-time*, pengguna tidak selalu menunjukkan *gesture* dengan posisi dan sudut yang konsisten seperti dalam gambar *dataset*.

##### B. Kondisi pencahayaan

Kondisi pencahayaan dalam pengujian *real-time* bervariasi tergantung pada lokasi pengguna.

##### C. Gesture tangan tidak konsisten

Karena *gesture* dilakukan langsung oleh pengguna, ada kemungkinan variasi dalam cara setiap orang melakukan bentuk huruf

##### D. Kemiripan bentuk gesture

Beberapa *gesture* memiliki bentuk yang secara visual sangat mirip, seperti *gesture* huruf C, G, O, Q, R, S, J, I yang semuanya melibatkan posisi jari melingkar atau saling bersinggungan

Meski demikian, sistem terbukti cukup andal dalam mengenali *gesture* secara langsung dan konsisten dengan akurasi rata-rata tinggi.

## V. KESIMPULAN

Penelitian ini berhasil menghasilkan aplikasi Teman Isyarat, sebuah aplikasi mobile berbasis Android yang ditujukan untuk membantu siswa tuna rungu di SLB Putra Mandiri Surabaya dalam mempelajari dan menggunakan Bahasa Isyarat Indonesia (SIBI). Aplikasi ini dikembangkan menggunakan metode prototyping dengan memanfaatkan teknologi Flutter dan model machine learning berbasis MediaPipe untuk mendeteksi *gesture* secara real-time. Hasil pengujian menunjukkan bahwa aplikasi Teman Isyarat dapat berfungsi sebagai media pembelajaran interaktif yang efektif dalam membantu siswa memahami dan mempraktikkan bahasa isyarat SIBI. Aplikasi ini dilengkapi dengan berbagai fitur utama seperti kamus isyarat, video pembelajaran, kuis interaktif, serta kamera penerjemah berbasis kecerdasan buatan. Komponen machine learning dalam aplikasi memainkan peran penting dengan memanfaatkan model CNN yang dilatih menggunakan dataset SIBI dari Kaggle dan data input manual, serta diintegrasikan dengan kamera Flutter melalui server Flask. Model pengenalan *gesture* menunjukkan performa yang cukup akurat dalam kondisi

pengujian, meskipun terdapat sedikit penurunan akurasi dalam implementasi nyata akibat variasi sudut pandang, pencahayaan, dan ketidakkonsistenan *gesture* pengguna. Secara keseluruhan, aplikasi ini tidak hanya mampu memenuhi kebutuhan sebagai alat bantu belajar bahasa isyarat, tetapi juga memberikan pengalaman belajar yang menyenangkan dan edukatif bagi anak-anak penyandang disabilitas.

## REFERENSI

- [1] A. Bintang. "SIBI Sign Language Dataset." Internet: <https://www.kaggle.com/datasets/alvations/sibi-sign-language>, 2021 [Jul. 15, 2025].
- [2] F. Chollet. Deep Learning with Python. New York: Manning Publications, 2017.
- [3] TensorFlow. "TensorFlow: An end-to-end open source machine learning platform." Internet: <https://www.tensorflow.org>, 2023 [Jul. 15, 2025].
- [4] Google. "MediaPipe Hands: Real-time hand and finger tracking." Internet: <https://google.github.io/mediapipe/solutions/hands.html>, 2023 [Jul. 15, 2025].
- [5] M. Abadi, et al. "TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems." Internet: <https://www.tensorflow.org>, 2015 [Jul. 15, 2025].
- [6] J. Brownlee. Imbalanced Classification with Python. [Online]. Machine Learning Mastery, 2020. Available: <https://machinelearningmastery.com/imbalanced-classification-with-python/> [Jul. 15, 2025].
- [7] A. Krizhevsky, I. Sutskever, and G. E. Hinton. "ImageNet Classification with Deep Convolutional Neural Networks." in Advances in Neural Information Processing Systems, vol. 25, 2012, pp. 1097-1105.
- [8] A. B. J. Teixeira, F. D. O. Teixeira, and J. M. de Souza. "Hand gesture recognition using deep learning: A review." ACM Computing Surveys, vol. 55, no. 2, pp. 1-36, 2023.
- [9] B. Kitchenham and S. Charters. "Guidelines for performing Systematic Literature Reviews in Software Engineering." Keele University & University of Durham, Technical Report EBSE-2007-01, 2007.
- [10] R. Szeliski. Computer Vision: Algorithms and Applications. New York: Springer, 2010.
- [11] Supabase. "Supabase: The open source Firebase alternative." Internet: <https://supabase.com>, 2024 [Jul. 15, 2025].
- [12] H. Zhang, H. Dai, Y. Sun, and C. Sun. "An Improved Real-Time Hand Gesture Recognition System Based on CNN." Journal of Physics: Conference Series, vol. 1820, no. 1, 2021.
- [13] N. Ketaren. "Implementasi Metode CNN untuk Pengenalan Bahasa Isyarat Indonesia." Jurnal Teknologi Informasi dan Komputer, vol. 9, no. 2, 2022.
- [14] Maze. "Remote Usability Testing Software." Internet: <https://maze.co>, 2024 [Jul. 15, 2025].
- [15] M. Miran, et al. "Real-Time Hand Gesture Recognition Using CNN and Transfer Learning." Sensors, vol. 21, no. 14, p. 4745, Jul. 2021.
- [16] S. Raschka and V. Mirjalili. Python Machine Learning. Birmingham, UK: Packt Publishing, 2019.