Jurnal Kreativitas Mahasiswa Informatika **E-ISSN: 3164-1989**
Penerbit: Fakultas Ilmu Komputer Universitas Pamulang **P-ISSN: 2797-6327**
Vol. 7, No. 2, Agustus 2026 (537-548) **https://doi.org/10.32493/jatimika.v7i1**

Penerapan *Computer Vision* Dengan *Gesture Recognition* Pada Aplikasi Penerjemah Bahasa Isyarat Indonesia (Bisindo) Berbasis *Mobile* Untuk Meningkatkan Komunikasi Efektif Pada Teman Tuli

Wulan Dwi Yanti[1*], Fadly Ariadi[2]

[1]Teknik Informatika, Universitas Pamulang, Jl. Surya Kencana No. 1, Pamulang Barat, Banten, Indonesia
e-mail: [1]wulanlanlanyanti@gmail.com, [2]dosen02389@unpam.ac.id
*Corresponding Author

Submitted Date: January 09, 2026 Reviewed Date: April 14, 2026
Revised Date: April 20, 2026 Accepted Date: May 24, 2026

Abstract

The existence of Communication barriers between the Deaf community and the general public, who often have limited or no mastery of Bahasa Isyarat Indonesia *(BISINDO), remain a significant challenge. This research aims to develop a real-time mobile translation application for BISINDO using computer vision and gesture recognition technology. The system implements the Convolutional Neural Network (CNN) algorithm with the YOLOv8 model to recognize hand gesture patterns forming the sign language alphabet. The methodology employed in the development of this system is the Agile method. The results demonstrate that the application successfully detects and translates single-alphabet sign gestures into text accurately with a stable response speed. Based on testing across 2,182 validation images, the model achieved a mean Average Precision (mAP50) of 0.995, a Precision rate of 0.989, and a Recall rate of 0.99. Furthermore, the system exhibits highly efficient computational performance with an inference speed of 3.1 ms per image. This proves that the model possesses high reliability in classifying the BISINDO alphabet, thereby enhancing communication effectiveness for the Deaf community in daily interactions*

Keywords: BISINDO*, Computer Vision, Gesture Recognition, Mobile App, CNN.*

Abstrak

Adanya hambatan komunikasi anatar teman Tuli dan masyarakat umum yang kurang atau bahkan tidak menguasai Bahasa Isyarat Indonesia (BISINDO) masih menjadi tantangan besar. Penelitian ini bertujuan untuk membuat aplikasi *mobile* penerjemah BISINDO secara *real-time* menggunakan teknologi *computer vision* dan *Gesture Recognition*. Sistem ini menggunakan algoritma *Convolutional Neural Network (CNN)* dengan model *YOLO V8* untuk mengenali pola gerakan tangan yang membentuk alfabet Bahasa isyarat. Metedologi yang digunakan dalam pengembangan sistem ini adalah metode *agile*. Hasil penelitian menunjukkan bahwa aplikasi berhasil mendeteksi dan menerjemahkan gerakan isyarat abjad tunggal menjadi teks secara akurat dengan kecepatan respon yang stabil. Berdasarkan pengujian pada 2182 citra validasi, model mencapai nilai *mean Average Precision* (mAP50) sebesar 0,995, tingkat *Precision* sebesar 0,989, dan *Recall* sebesar 0,99. Selain itu, sistem menunjukkan performa komputasi yang sangat efisien dengan kecepatan *inference* sebesar 3,1 ms per citra. Hal ini membuktikan bahwa model memiliki keandalan tinggi dalam mengklasifikasikan abjad BISINDO sehingga dapat meningkatkan efektivitas komunikasi bagi teman Tuli dalam interaksi sehari-hari.

Kata kunci : BISINDO, *Computer Vision*, *Gesture Recognition*, *Mobile App*, CNN.

1. Pendahuluan

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International (CC BY-NC 4.0) License
Copyright © 2026 Author1, Author2, Author3

Komunikasi merupakan kebutuhan mendasar setiap manusia untuk bertukar informasi dan membangun hubungan sosial. Namun, teman Tuli yang menggunakan Bahasa Isyarat Indonesia (BISINDO) seringkali menghadapi kendala saat berinteraksi dengan masyarakat umum yang memiliki keterbatasan pemahaman terhadap bahasa isyarat tersebut. Dampaknya, akses terhadap layanan publik dan partisipasi sosial menjadi terhambat.

Seiring perkembangan *Artificial Intelligence (AI)*, teknologi *Gesture Recognition* berbasis *Computer Vision* muncul sebagai solusi untuk membantu menjembatani kesenjangan ini dengan menerjemahkan gerakan tangan menjadi teks. Penelitian ini difokuskan pada pengembangan aplikasi *mobile* berbasis *Android* yang mampu melakukan penerjemahan secara *real-time.* Fokus utama aplikasi ini adalah pengenalan abjad dasar BISINDO yang dapat membantu komunitas seperti Majelis Ta'lim Tuli Indonesia (MTTI) dalam meningkatkan aksesibilitas komunikasi.

Meskipun saat ini sistem difokuskan pada pengenalan abjad tunggal, pengembangan ini merupakan langkah awal yang krusial sebelum melangkah ke pengenalan kata atau kalimat dinamis. Tantangan utama dalam pengembangan ke depan adalah kompleksitas gerakan BISINDO yang melibatkan aspek temporal (waktu), kecepatan gerak, dan perubahan orientasi tangan yang jauh lebih dinamis dibandingkan abjad statis. Oleh karena itu, rencana pengembangan selanjutnya akan diarahkan pada integrasi algoritma yang mampu memproses data *video-based* untuk menangkap esensi gerakan bahasa isyarat secara utuh dan berkelanjutan.

2. Metodologi Penelitian

2.1 Metode Pengumpulan Data

Teknik pengumpulan data yang penulis lakukan dalam mencari dan mengumpulkan data serta mengolah informasi yang diperlukan menggunakan beberapa metode sebagai berikut:

a. Metode Wawancara

Metode wawancara dilakukan melalui beberapa dialog dan tanya jawab secara mendalam terhadap salah satu narasumber, yaitu salah satu pengurus dan sekaligus penanggung jawab di Komunitas Majelis Ta'lim Tuli Indonesia. Proses ini tidak hanya bertujuan untuk mengumpulkan sebuah informasi dasar, tetapi juga untuk membantu identifikasi kebutuhan sistem langsung dari pengguna akhir *(end user).* Melalui metode ini,penulis dapat mengidentifikasi secara detail hambatan komunikasi nyata yang dihadapi oleh Teman Tuli di lapangan, memahami preferensi antarmuka aplikasi yang ramah pengguna, serta memvalidasi variasi gerakan tangan dalam BISINDO yang paling krusial untuk diprioritaskan dalam sistem penerjemahan.

b. Studi Pustaka

Metode ini dilakukan dengan cara melakukan penelusuran, pengumpulan, dan analisis terhadap berbagai jurnal ilmiah serta dokumen yang terpercaya. Penulis mengkaji berbagai sumber yang meliputi jurnal penelitian nasional maupun internasional mengenai kecerdasan buatan, teknologi, serta laporan penelitian terdahulu yang berkaitan erat dengan tema *Computer Vision, Gesture Recognition*, dan implementasi algoritma *Convolutional Neural Network (CNN).* Studi pustaka ini berfungsi sebagai landasan teoretis yang kokoh untuk memahami mekanisme kerja model YOLOv8 dalam deteksi objek, serta sebagai bahan rujukan komparatif untuk memastikan bahwa posisi penelitian ini memiliki nilai inovasi dan kebaruan dibandingkan dengan aplikasi penerjemah bahasa isyarat yang sudah ada sebelumnya.

2.2 Metode Pengembangan Sistem

Penelitian ini menggunakan pendekatan pengembangan perangkat lunak dengan metode *agile*. Pemilhan metode ini didasarkan pada karakteristiknya yang mudah untuk disesuaikan, fleksibel, dan iteratif. Sehingga, memungkinkan penulis untuk melakukan perbaikan berkelanjutan berdasarkan hasil evaluasi pada setiap tahapannya. Adapun tahapan penelitian meliputi:

a. Analisa Kebutuhan

Tahap awal ini difokuskan untuk mengidentifikasi secara spesifikasi teknis dan fungsional yang diperlukan agar sistem dapat beroperasi secara optimal. Analisis mencakup kebutuhan sistem dalam menangkap *input* visual gerakan tangan secara *real time* melalui modul kamera pada perangkat *smartphone*. Selain itu, dilakukan pemetaan terhadap kebutuhan komputasi algoritma *Convolutional Neural Network (CNN)* agar proses pemisahan fitur dan klasifikasi isyarat dapat berjalan lancar tanpa mengalami jeda yang tinggi pada perangkat *mobile*.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International (CC BY-NC 4.0) License
Copyright © 2026 Author1, Author2, Author3

b. Pengumpulan Data

Mengingat performa kecerdasan buatan sangat bergantung pada kualitas data, penulis melakukan pengumpulan dataset citra gerakan tangan yang merepresentasikan abjad BISINDO. Seluruh data yang terkumpul dianotasi secara detail melalui platform *Roboflow* untuk memberikan label yang presisi pada setiap objek tangan. Pada tahap ini juga dilakukan proses *preprocessing* dan augmentasi data seperti pengaturan kecerahan *(brightness),* rotasi, dan *flipping* guna memperkaya variasi data. Hal ini bertujuan agar model *AI* memiliki tingkat generalisasi yang baik saat menghadapi perbedaan kondisi pencahayaan atau sudut pandang kamera di lapangan.

c. Perancangan Sistem

Untuk membantu memberikan gambaran yang terstruktur, perancangan sistem menggunakan *Unified Modelling language (UML).* Perancangan ini mencakup:

1) *Use Case Diagram*

Digunakan untuk mengidentifikasi dan memetakan secara visual hubungan fungsional antara aktor, dalam hal ini adalah pengguna (Teman Tuli maupun masyarakat umum), terhadap berbagai fitur utama yang telah disediakan oleh aplikasi. Diagram ini memberikan gambaran mengenai batasan sistem serta bagaimana pengguna berinteraksi dengan modul deteksi isyarat, pengaturan aplikasi, dan fitur pendukung lainnya guna mencapai tujuan komunikasi.

2) *Activity Diagram*

Dirancang khusus untuk menjabarkan seluruh alur kerja sistem secara mendalam. Diagram ini memvisualisasikan langkah-langkah yang dimulai dari tahap inisialisasi modul kamera pada perangkat *mobile*, pengambilan citra tangan secara dinamis, proses pemrosesan data oleh algoritma, hingga berakhir pada penampilan hasil terjemahan teks pada layar aplikasi. Hal ini memastikan bahwa setiap transisi aktivitas dalam sistem dapat diapahami alurnya dengan jelas dan sistematis.

3) *Sequence Diagram*

Berfungsi untuk memodelkan urutan pertukaran pesan *(massage exchange)* serta logika operasional antar komponen perangkat lunak secara kronologis. Pemodelan ini sangat krusial untuk memastikan adanya integrase yang sinkron dan efisien antara antarmuka pengguna *(user interface)* dengan mesin inferensi kecerdasan buatan *(YOLOv8).* Dengan adanya diagram ini, interaksi antar objek dalam sistem dapat divalidasi sehingga data visual yang ditangkap kamera dapat diproses dan dikembalikan dalam bentuk teks secara tepat waktu *(real time)* dan akurat.

4) Implementasi

Pada fase ini, rancangan teknis direalisasikan ke dalam bentuk kode program. Logika pemrosesan data dan pelatihan model dibangun menggunakan bahasa pemrograman *Python* dengan bantuan alat *Jupyter Notebook* untuk keperluan eksperimen dan analisis data. Untuk antarmuka pengguna pada perangkat *mobile*, penulis menggunakan *framework Flutter*, yang menjamin performa aplikasi yang responsif dan estetika visual yang modern. Sebagai inti dari sistem deteksi, digunakan model *YOLOv8 (You Only Look Once version 8)*, yang dipilih karena keunggulannya dalam melakukan deteksi objek dengan kecepatan sangat tinggi namun tetap menjaga tingkat akurasi yang optimal untuk skala perangkat *mobile*.

5) Pengujian

Tahap akhir dilakukan untuk memastikan kualitas sistem melalui pengujian fungsionalitas menggunakan metode *Blackbox Testing*. Fokus pengujian ini adalah untuk memvalidasi setiap fitur aplikasi seperti fungsionalitas tombol, transisi halaman, dan akurasi keluaran teks apakah sudah sesuai dengan spesifikasi kebutuhan yang telah ditetapkan sebelumnya. Selain pengujian fitur, dilakukan juga validasi terhadap model deteksi guna mengukur sejauh mana aplikasi dapat mengenali gerakan isyarat secara tepat dalam berbagai skenario penggunaan nyata.

3. Hasil dan Diskusi

3.1 Perancangan Sistem

a. *Use Case Diagram*

Use Case pada dasarnya adalah representasi dari proses sistem secara menyeluruh yang melibatkan aktor, khususnya pengguna. *Use Case* digunakan untuk menunjukkan bagaimana *stakeholder* sistem akan

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International (CC BY-NC 4.0) License
Copyright © 2026 Author1, Author2, Author3

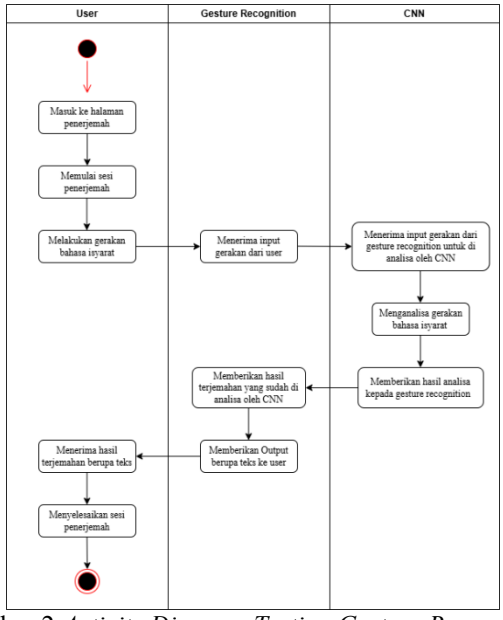
berinteraksi dengan sistem. Pengembangan *Use Case* membantu dalam memahami persyaratan sistem secara rinci. Berikut adalah use case yang ada pada sistem penerjemahan BISINDO :

Gambar 1 *Use Case Diagram*

b. Activity Diagram

Activity Diagram adalah representasi grafis dalam *Unified Modeling Language (UML)* yang melayani tujuan utama memodelkan alur kerja prosedural dari sebuah sistem. Diagram ini secara eksplisit menggambarkan urutan aksi atau langkah-langkah yang harus diselesaikan untuk mencapai tujuan tertentu. Dalam konteks terbaru, diagram ini tidak hanya menunjukkan aliran kontrol antar aktivitas tetapi juga aliran objek (data) yang diolah atau ditransfer selama proses tersebut, serta mengidentifikasi partisi (*swimlanes*) yang memisahkan tanggung jawab antara berbagai komponen sistem atau aktor.

1) Activity Diagram Testing Gesture Recognition



Gambar 2 *Activity Diagram Testing Gesture Recognition*

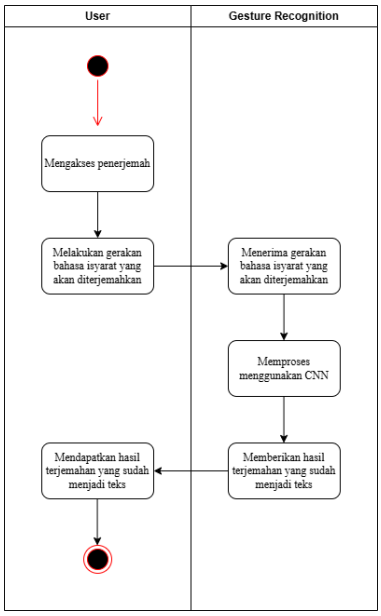
Pada gambar 2 menggambarkan alur aktivitas pengujian *Gesture Recognition*. Proses diawali ketika *user* membuka halaman penerjemah, lalu memulai sesi penerjemah dan melakukan gerakan Bahasa isyarat.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International (CC BY-NC 4.0) License
Copyright © 2026 Author1, Author2, Author3

Gesture Recognition menerima input gerakan dari *user*, setelahnya akan dianalisis ke modul *CNN* dan memberikan hasil analisanya kepada *gesture recognition*. Hasil yang sudah ada akan diberikan kepada *user* dalam bentuk *output* teks, lalu *user* akan menerima hasil terjemahannya dan setelahnya bisa menyelesaikan sesi penerjemahan.

2) *Activity Diagram* Interaksi Pengguna

Pada gambar 3 menunjukan alur interaksi pengguna antara *user* dengan *gesture recognition* dalam sistem penerjemahan. Proses dimulai saat *user* mengakses sistem penerjemahan, kemudian melakukan gerakan Bahasa isyarat yang ingin diterjemahkan. Gerakan Bahasa isyarat akan diterima oleh *gesture recognition*, lalu diproses menggunakan teknologi *convolutional neural networks* (CNN) untuk menganalisis gerakan Bahasa isyarat tersebut. Setelah itu, *gesture recognition* akan memberikan hasil terjemahan kepada *user*.



Gambar 3 *Activity Diagram* Interaksi Pengguna

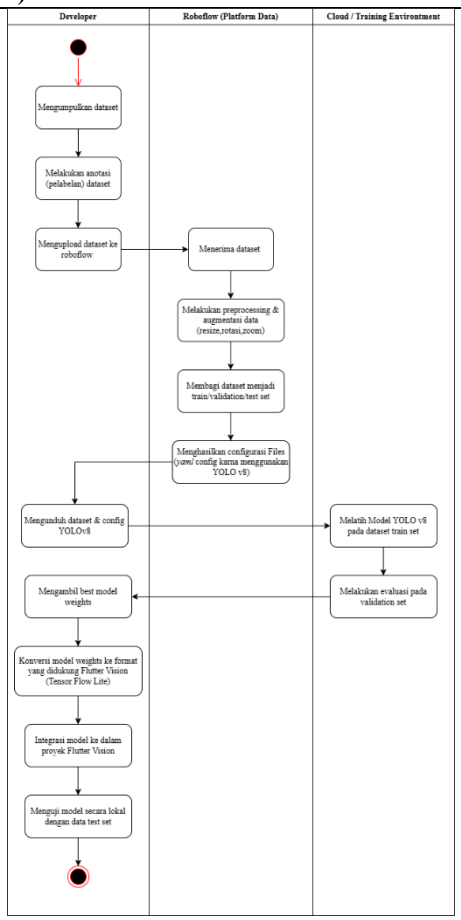
User selanjutnya akan menerima hasil terjemahan dari *gesture recognition* berupa teks, dan setelah menerima terjemahan *user* akan mengakhiri sesi terjemahan.

3) Activity Diagram Training Gesture

Pada gambar 4 *Activity Diagram Training Gesture Recognition* (Pembelajaran *Gesture Recognition*) menunjukan proses pengumpulan *dataset* untuk menjaga agar *gesture recognition* tetap akurat saat menerjemahkan gerakan bahasa isyarat dari *user*. Proses ini dimulai dari *developer* mengumpulkan dataset berupa gambar atau *video gesture* Bahasa isyarat, selanjutnya developer melakukan anotasi *(labelling)* dataset menggunakan format *YOLOv8.*

Setelah melabelling, dataset akan di upload ke *Roboflow*. *Roboflow* akan menerima dataset dan melakukan *preprocessing* dan augmentasi data seperti meresize, merotasi. Setelahnya dataset dijadi menjadi *train/validation/test set*, dan akan menghasilkan *configuration files* menggunakan *yaml config* untuk *YOLOv8).* Setelahnya *developer* mengunduh dataset dan *config YOLOv8* dari *Roboflow.* Lalu, *developer* melatih model *YOLOv8* pada *dataset train set* dan melakukan evaluasi pada *validation set.* Setelah melakukan evaluasi, *developer* akan mengambil *best model weight* nya dan menkonversinya ke format yang mendukung *Flutter Vision* dan yang digunakan disini adalah *Tensor Flow Lite.* Setelah berhasil menkonversinya, *developer* akan menitegrasi model ke dalam proyek *Flutter Vision* dan menguji model secara lokal dengan data test set, dan proses training selesai.

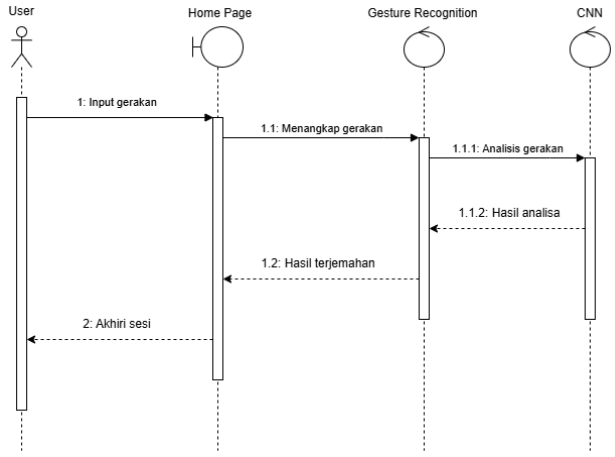
This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International (CC BY-NC 4.0) License
Copyright © 2026 Author1, Author2, Author3



Gambar 4 *Activity Diagram Training Gesture*

c. Sequence diagram

1) *Sequence Diagram Testing Gesture Recognition*



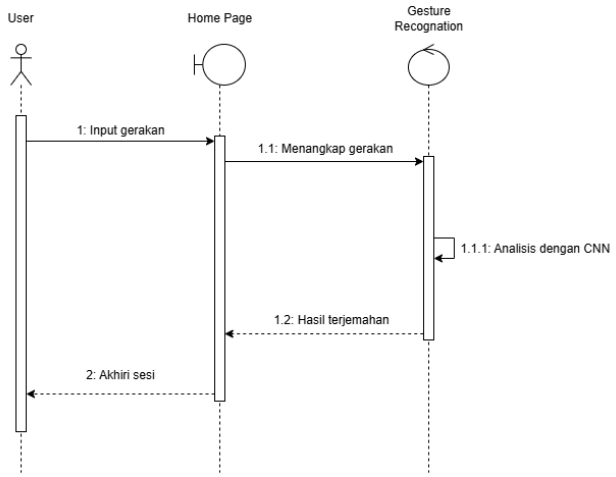
Gambar 5 *Sequence Diagram Testing Gesture Recognition*

Proses dimulai ketika *User* memberikan *Input* gerakan (Langkah 1) kepada *Home Page. Home Page* segera merespons dengan meminta modul *Gesture* Recognition untuk Menangkap gerakan tersebut (*Langkah* 1.1). Setelah menangkap gerakan, modul tersebut mendelegasikannya kepada model *CNN* untuk menjalani Analisis gerakan (Langkah 1.1.1). Model *CNN*, yang berperan sebagai inti pengenalan, akan memproses data tersebut dan mengembalikan Hasil analisa (Langkah 1.1.2) kepada modul *Gesture Recognition*. Berbekal hasil analisis ini, modul *Gesture Recognition* melakukan terjemahan akhir dan

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International (CC BY-NC 4.0) License
Copyright © 2026 Author1, Author2, Author3

mengirimkan Hasil terjemahan (Langkah 1.2) kembali ke *Home Page*. Seluruh interaksi ini menunjukkan kolaborasi sistematis, di mana *Home Page* bertindak sebagai antarmuka, modul *Gesture Recognition* mengelola data, dan *CNN* melakukan klasifikasi/identifikasi, hingga akhirnya sesi diakhiri dengan pesan Akhiri sesi (Langkah 2) kepada *User*.

2) *Sequence Diagram* Interaksi Pengguna



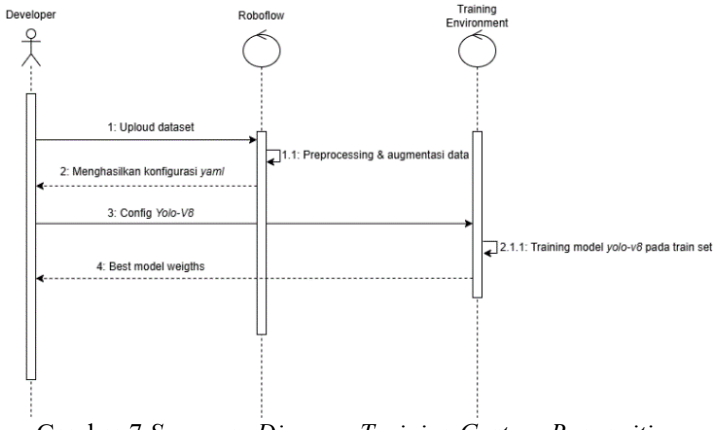
Gambar 6 *Sequence Diagram* Interaksi Pengguna

Proses interaksi dimulai ketika *User* memberikan *Input* gerakan (Langkah 1) kepada *Home Page*. *Home Page*, yang bertindak sebagai antarmuka utama, kemudian langsung memanggil modul *Gesture Recognition* untuk Menangkap gerakan tersebut (Langkah 1.1).

Modul *Gesture Recognition* memegang tanggung jawab inti dalam memproses input ini. Setelah menangkap gerakan, modul ini segera melakukan pemrosesan lebih lanjut dengan menjalankan Analisis dengan *CNN* (Langkah 1.1.1). Meskipun model *CNN* tidak digambarkan sebagai lifeline terpisah (seperti pada diagram sebelumnya), tindakan ini mengindikasikan bahwa proses analisis dan klasifikasi gerakan menggunakan model pembelajaran mendalam *(CNN)* terjadi di dalam konteks atau melalui pemanggilan oleh modul *Gesture Recognition* itu sendiri.

Setelah analisis selesai, modul *Gesture Recognition* menghasilkan Hasil terjemahan (Langkah 1.2), yang merupakan interpretasi atau makna dari gerakan yang dimasukkan pengguna. Hasil terjemahan ini dikirimkan kembali ke *Home Page. Home Page* kemudian menggunakan hasil ini untuk memberikan umpan balik atau memicu fungsi yang relevan. Seluruh proses interaksi diselesaikan ketika *Home Page* mengirimkan pesan akhiri sesi (Langkah 2) kembali kepada *User*. Diagram ini secara ringkas namun jelas memvisualisasikan seluruh siklus dari input pengguna hingga penyelesaian sesi interaksi pengenalan gerakan.

3) Sequence Diagram Training Gesture Recognition



Gambar 7 *Sequence Diagram Training Gesture Recognition*

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International (CC BY-NC 4.0) License
Copyright © 2026 Author1, Author2, Author3

Proses dimulai oleh *Developer* yang mengunggah dataset gerakan yang telah dianotasi (Langkah 1) ke platform *Roboflow*. *Roboflow*, yang dikenal sebagai alat manajemen dan peningkatan *dataset* untuk *computer vision*, kemudian secara otomatis melakukan *Preprocessing & augmentasi data* (Langkah 1.1) untuk mempersiapkan data agar optimal untuk pelatihan.

Dataset yang digunakan dalam penelitian ini memiliki total 24.448 gambar dengan 24.796 instansi objek yang mencakup 47 kelas isyarat BISINDO. Dataset tersebut dibagi secara sistematis menjadi 22.119 gambar untuk tahap *training* (90,4%), 2.182 gambar untuk tahap *validation* (8,9%), dan 147 gambar untuk tahap *testing* (0,7%). Untuk meningkatkan keandalan model terhadap penggunaan di dunia nyata, dilakukan proses augmentasi pada platform Roboflow yang mencakup variasi kondisi lingkungan seperti penyesuaian kecerahan (*brightness*), pemberian efek *blur*, serta penambahan *noise*. Hal ini dilakukan agar model tetap mampu mengenali abjad BISINDO secara akurat meskipun terdapat perbedaan intensitas cahaya maupun kompleksitas latar belakang pada saat pengambilan gambar secara langsung.

Setelah dataset siap, *Developer* mengambil konfigurasi yang diperlukan dengan Menghasilkan konfigurasi yaml (Langkah 2) dari *Roboflow*. Konfigurasi ini penting karena berisi parameter yang akan digunakan oleh model selama pelatihan. Selanjutnya, *Developer* mengirimkan konfigurasi dan dataset yang telah diproses ke *Training Environment* melalui langkah *Config Yolo-V8* (Langkah 3), yang secara spesifik menunjukkan bahwa model *object detection YOLO-V8* yang akan digunakan.

Langkah berikutnya adalah inti dari proses ini: Training Environment menjalankan Training model yolo-V8 pada train set (Langkah 3.1.1). Proses ini melibatkan iterasi pelatihan model menggunakan dataset yang disiapkan. Setelah pelatihan selesai, Developer mengambil hasil terbaik, yaitu Best model weights (Langkah 4), dari Training Environment. Weights ini adalah keluaran paling penting karena merepresentasikan model yang telah dilatih dan siap digunakan untuk inferensi pengenalan gerakan.

d. Flowchart

Flowchart atau diagram alir adalah representasi visual yang digunakan untuk menggambarkan langkah-langkah, urutan, dan keputusan dalam suatu proses atau algoritma secara sistematis. Diagram ini menggunakan simbol-simbol standar seperti oval untuk menandai awal dan akhir, persegi panjang untuk proses, serta belah ketupat untuk keputusan, yang kemudian dihubungkan dengan panah untuk menunjukkan arah alur kerja. Flowchart dapat membantu merancang proses secara rinci sehingga memudahkan analisis dan dokumentasi (Dewaweb 2024). Berikut adalah *flowchart* yang ada pada sistem penerjemahan BISINDO:

Gambar 8 *Flowchart*

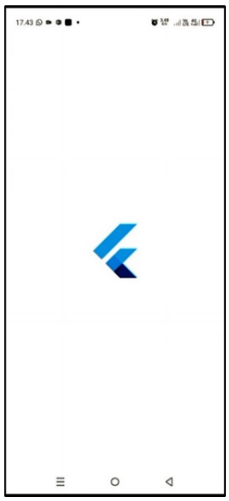
This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International (CC BY-NC 4.0) License
Copyright © 2026 Author1, Author2, Author3

3.2 Implementasi Sistem

Dalam konteks TI, implementasi mencakup instalasi, konfigurasi, kustomisasi, integrasi sistem, pelatihan pengguna serta pengiriman dan pengubahan sistem agar sesuai kebutuhan organisasional (Lutkevich, 2022). Berikut adalah implementasi sistem penerjemahan BISINDO:

a. Tampilan *Loading Page 1*

Halaman *Loading Page 1* (Gambar 3) merupakan tampilan pertama yang dimana menandakan aplikasi akan dibuka. Halaman ini berfungsi sebagai layar splash awal dan indikator yang menandakan bahwa aplikasi telah beroperasi.



Gambar 9 *Loading Page 1*

b. Tampilan *Loading Page 2*

Halaman *Loading Page 2* (Gambar 4) merupakan tampilan penting dalam alur inisiasi sistem yang berfungsi sebagai indikator visual untuk proses pemuatan data yang kompleks.

Gambar 10 *Loading page 2*

c. Tampilan Halaman Utama *(Home Page)*

Halaman Utama *(Home Page)* (Gambar 11) merupakan tampilan implementasi *user interface* yang berfungsi sebagai titik awal untuk proses penerjemahan.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International (CC BY-NC 4.0) License
Copyright © 2026 Author1, Author2, Author3

Gambar 11 *Home Page*

d. Tampilan Halaman Penerjemahan *(Translate Page)*

Halaman Penerjemahan *(Translate Page 1)* (Gambar 12 dan 13) merupakan tampilan inti sistem yang menampilkan fungsi utama penerjemahan BISINDO secara *real-time.* Antarmuka ini didominasi oleh area *viewfinder* kamera, yang seacara langsung menampilkan tangan penggunan saat melakukan gerakan isyarat. Di atas area tangan, terdapat kotak *output* yang diberi label *"LETTER"* yang berfungsi sebagai area hasil terjemahan.

Gambar 12 *Translate Page 1*

Gambar 13 *Translate Page 2*

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International (CC BY-NC 4.0) License
Copyright © 2026 Author1, Author2, Author3

4. Kesimpulan

Berdasarkan hasil pembahasan penelitian ini pada penerapan teknologi computer vision dan gesture recognition untuk aplikasi penerjemah bahasa isyarat Indonesia (BISINDO) dapat disimpulkan sebagai berikut:

a. Penerapan teknologi *Computer Vision dan Gesture Recognition* dalam ekosistem sistem ini diwujudkan secara komprehensif melalui pemanfaatan arsitektur *Deep Learning* berbasis *Convolutional Neural Network* (CNN). Model ini memegang peranan yang sangat krusial dalam melakukan ekstraksi fitur visual secara mendalam serta mengklasifikasikan pola-pola gerakan tangan yang dinamis dan kompleks menjadi representasi alfabet Bahasa Isyarat Indonesia (BISINDO). Proses transformasi ini memungkinkan sistem untuk mengubah masukan citra visual menjadi data teks digital yang informatif, sehingga memudahkan proses interpretasi makna bagi masyarakat luas yang tidak menguasai bahasa isyarat.
b. Kehadiran aplikasi penerjemah berbasis *mobile* ini secara signifikan mampu mengoptimalkan serta meningkatkan efektivitas komunikasi interpersonal bagi komunitas Teman Tuli dalam skala interaksi harian. Dengan menawarkan solusi digital yang jauh lebih praktis, portabel, dan responsif, aplikasi ini hadir sebagai alternatif mutakhir yang mampu menggantikan ketergantungan pada metode komunikasi konvensional yang bersifat manual seperti penggunaan buku catatan fisik atau pengetikan teks pada aplikasi memo yang selama ini dinilai kurang efisien, memakan waktu lama, serta sering kali menghambat spontanitas dalam pertukaran informasi secara langsung.
c. Berdasarkan serangkaian hasil pengujian fungsionalitas sistem yang telah dilakukan secara sistematis, aplikasi ini menunjukkan performa yang sangat solid dengan tingkat akurasi yang memadai dalam mengidentifikasi huruf-huruf alfabet tunggal BISINDO secara presisi. Selain aspek akurasi, sistem ini dirancang untuk mampu beroperasi pada kecepatan respon real-time yang stabil dengan tingkat latensi yang rendah. Pengujian dilakukan pada berbagai perangkat dengan spesifikasi berbeda, mulai dari tingkat *entry-level* hingga *mid-range*, yang meliputi Infinix Smart 9 (RAM 3GB), Oppo A31 (RAM 6GB), Infinix Hot 10s (RAM 6GB), hingga Samsung Galaxy A35 5G (RAM 8GB). Hasil pengujian menunjukkan bahwa sistem mampu menghasilkan kecepatan deteksi (*inference*) rata-rata sebesar 3,1 ms per citra. Dengan latensi yang sangat rendah tersebut, aplikasi mampu mempertahankan frame rate yang stabil untuk menjamin kelancaran alur interaksi dan meminimalisir hambatan teknis (jeda waktu) saat aplikasi digunakan dalam situasi percakapan yang sebenarnya.

Aspek kemudahan penggunaan atau *User Experience (UX)* pada aplikasi ini menjadi prioritas utama yang diimplementasikan melalui rancangan antarmuka yang intuitif namun tetap fungsional. Mekanisme operasionalnya dirancang sedemikian rupa agar pengguna dapat dengan mudah memposisikan gerakan isyarat tangan di depan kamera ponsel; secara simultan, sistem akan melakukan pemrosesan data melalui mesin inferensi kecerdasan buatan untuk kemudian menampilkan hasil terjemahan berupa teks secara instan pada layar ponsel. Integrasi yang seamless antara perangkat keras kamera dan algoritma pengolah data ini memastikan bahwa teknologi yang dikembangkan benar-benar ramah pengguna dan tepat guna.

a. Saran

Berdasarkan hasil penelitian ini, berikut beberapa saran yang dapat dijadikan acuan untuk pengembangan sistem di masa mendatang agar hasilnya lebih optimal, yaitu sebagai berikut:

1. Pengembangan kosakata dan penerjemahan kalimat utuh, pada saat ini sistem masih berfokus pada deteksi abjad atau Isyarat tinggal dan hanya beberapa kata saja karena adanya kekurangan *dataset* yang ada. Pengembangan model harus dilanjutkan dengan memperluas *dataset* untuk pengenalan tidak hanya abjad tunggal, tetapi juga kata-kata dasar dan frasa umum dalam BISINDO, sehingga sistem mampu menerjemahkan makna yang lebih kompleks.
2. Fungsionalitas sistem bisa lebih ditingkatkan lagi menjadi dua arah, yaitu dengan menambah fitur terjemahan dari teks atau suara lisan ke dalam bentuk Isyarat.
3. Model bisa lebih dilatih lagi dengan variasi data yang lebih luas, mencakup kondisi pencahayaan yang kurang ideal, latar belakang yang berbeda-beda, dan berbagai warna kulit tangan pengguna, untuk memastikan akurasi sistem di lingkungan penggunaan yang beragam.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International (CC BY-NC 4.0) License
Copyright © 2026 Author1, Author2, Author3

4. Model pada kecerdasan buatan *(AI)* perlu dioptimalkan agar memiliki ukuran yang lebih kecil dan waktu pemrosesan yang lebih cepat, sehingga dapat berjalan dengan lancar dan *real-time* pada perangkat *mobile* dengan spesifikasi standar.
5. Aplikasi dapat menambahkan fitur pembelajaran BISINDO yang lebih interaktif, termasuk modul latihan yang memberikan umpan balik *real-time* kepada pengguna mengenai keakuratan gerakan Isyarat yang mereka praktikkan.

Saran-saran di atas diharapkan bisa menjadi pedoman dalam membuat sistem penerjemahan yang lebih beragam, lebih efisien, dan sesuai dengan adanya kebutuhan komunikasi. Sehingga sistem tersebut dapat terus berkembang dan memberikan terjemahan yang lebih tepat serta bermanfaat bagi pengguna.

Referensi

Athitsos., T. &. (2020). Human-Computer Interaction: Gesture recognition from image sequences.

Cholil, S. R. (2021). Implementasi Algoritma Convolutional Neural Network Untuk Pengenalan Isyarat Tangan. *Jurnal Teknologi Informasi dan Terapan (JTIK)*.

Darmawan, A. (2021). Aplikasi Hand Gesture Recognition Sebagai Media Penerjemah Bahasa Isyarat Berbasis Android.

Farid, A. M. (2024). Spesifikasi Kebutuhan Perangkat Lunak Sistem Informasi E-Survey… Infotek: Jurnal Informatika dan Teknologi.

Ganjarrintana, D. e. (n.d.). Penerapan SDLC Agile dalam Pembuatan SIRUMA (Sistem Informasi Rumah Makan). *Ejournal TSB*.

Gao. (2023). Gesture recognition in image processing and artificial intelligence technology.

Halim, H. &. (2023). Aplikasi Pengidentifikasi Bahasa Isyarat Berdasarkan Gerak Tubuh Secara Real Time Menggunakan YOLO. . *Simtek: jurnal sistem informasi dan teknik komputer*.

Hanifah. (2025). Perancangan website UI/UX untuk pembelajaran Bahasa Isyarat Indonesia (BISINDO) dengan pendekatan User Centered Design (UCD).

Jahanger, A. K. (2022). Artificial Intelligence and Deep Learning: Concepts and Applications. Publisher Academic Press.

Jahanger, Q. K. (2022). Implementation Framework to Facilitate Digitalization of Construction-Phase Information Management by Project Owners. *Journal of Information Technology in Construction (ITcon)*.

Juliandy, C. &. (2022). Computer vision and machine perception.

Lutkevich, B. (2022). What is implementation? TechTarget.

Mohamed, N. a. (2021). Gesture recognition as an intuitive communication bridge.

Moorent, R. (n.d.). Application development and data processing commands.

Muhamad, F. N. (2023). Fault-Prone Software Requirements Specification Detection Using Ensemble Learning for Edge/Cloud Applications. *Applied Sciences*.

Permana, D. &. (2023). Aplikasi Pengenalan Abjad Sistem Isyarat Bahasa Indonesia (SIBI) Dengan Algoritma YOLOv5. *Jurnal Simantec*.

Rączkowska, K. &.-G. (2022). What Should a Good Software Requirements Specification Include? Results of a Survey.

Roboflow. (2023). Roboflow: Computer Vision Tools for Developers and Data Scientists.

Saputra, Y. &. (n.d.). Sistem baru dan peningkatan pelayanan informasi. 2025.

Sari, I. P. (2022). Penerapan Metode Agile Dalam Pengembangan Aplikasi Mobile. *Jurnal Informatika dan Teknologi (Jintek)*.

Seaborn., U. &. (2023). Non-verbal communication in human-robot interaction.

Singh., K. &. (2023). Deep context and language transfer in digital translation.

Wanto, A. e. (2020). Teknik Deep Learning: Teori dan Implementasi. *Medan: Yayasan Kita Menulis.*

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International (CC BY-NC 4.0) License
Copyright © 2026 Author1, Author2, Author3