Buletin Sistem Informasi dan Teknologi Islam
Vol 7, No 3 (2026), pp 217-229
ISSN: 2721-0901
DOI: https://dx.doi.org/10.33096/busiti.v7i3.3506

Analisis Kontribusi Deteksi Tangan dan Ekstraksi Landmark untuk Pengenalan Alfabet BISINDO

Contribution Analysis of Hand Detection and Landmark Extraction for BISINDO Alphabet Recognition

Wa Nanda Sulystrian[a,1], Muhammad Faisal[a,2,*], Fahrim Irhamna Rachman[a,3], Abd Rakhim Nanda[b,4], Rizki Yusliana Bakti[a,5], Muhammad Syafaat S. Kuba[b,6], Andi Makbul Syamsuri[b,7]

[a] *Informatika, Universitas Muhammadiyah Makassar, Makassar, Indonesia*
[b] *Teknik Pengairan, Universitas Muhammadiyah Makassar, Makassar, Indonesia*
[1]*105841110622@student.unismuh.ac.id;* [2]*muhfaisal@unismuh.ac.id;* [3]*fachrim141020@unismuh.ac.id;*
[4]*abd.rakhimnanda@unismuh.ac.id;* [5]*rizkyyusliana@unismuh.ac.id;* [6]*syafaat_skuba@unismuh.ac.id,*
[7]*amakbulsyamsuri@unismuh.ac.id*
**corresponding author*

Informasi Artikel

Diserahkan : 9 Mei 2026
Diterima : 31 juli 2026
Direvisi : 4 Agustus 2026
Diterbitkan : 18 Agustus 2026

Kata Kunci:
BISINDO-Huruf
Deteksi Tangan Berbasis-YOLOv8
MediaPipe Hands
Normalisasi Landmark
Multi-Layer Perceptron

ABSTRAK

Komunitas tunarungu menghadapi hambatan dalam aspek bahasa dan komunikasi sehingga membutuhkan dukungan teknologi yang sesuai dengan karakteristik visual mereka. Kondisi tersebut menunjukkan pentingnya pengembangan teknologi untuk mendukung akses komunikasi yang lebih inklusif melalui pengenalan Bahasa Isyarat Indonesia (BISINDO). Penelitian ini menganalisis kontribusi deteksi tangan dan ekstraksi landmark terhadap performa pengenalan alfabet BISINDO menggunakan pendekatan berbasis *computer vision*. Sistem yang diusulkan mengintegrasikan YOLOv8 untuk deteksi tangan, MediaPipe Hands untuk ekstraksi dua puluh satu landmark, normalisasi landmark, serta Multi-Layer Perceptron sebagai model klasifikasi. Evaluasi dilakukan menggunakan dataset berisi 1066 citra alfabet BISINDO dari 26 kelas melalui tiga skenario eksperimen. Hasil penelitian menunjukkan bahwa kombinasi deteksi tangan dan normalisasi landmark menghasilkan performa terbaik dengan nilai accuracy 0.915, precision 0.913, recall 0.915, dan F1-score 0.896, serta meningkatkan accuracy 14.8% dibandingkan pendekatan tanpa deteksi tangan. Temuan ini menunjukkan bahwa deteksi tangan dan representasi landmark berkontribusi penting terhadap peningkatan akurasi sistem. Pendekatan yang diusulkan berpotensi diterapkan pada aplikasi penerjemah alfabet BISINDO berbasis kamera secara *real-time* untuk mendukung komunikasi yang lebih inklusif bagi komunitas tunarungu di Indonesia.

***Keywords*:**
BISINDO-Alphabet
YOLOv8-Based Hand Detection
MediaPipe Hands
Landmark Normalization
Multi-Layer Perceptron

ABSTRACT

Deaf communities face barriers in language and communication, creating a need for technological support that aligns with their visual communication characteristics. This condition highlights the importance of developing technology to support more inclusive communication access through Indonesian Sign Language (BISINDO) recognition. This study analyzes the contribution of hand detection and landmark extraction to BISINDO alphabet recognition performance using a computer vision-based approach. The proposed system integrates YOLOv8 for hand detection, MediaPipe Hands for extracting twenty-one landmarks, landmark normalization, and a Multi-Layer Perceptron as the classification model. Evaluation was conducted using a dataset consisting of 1066 BISINDO alphabet images from 26 classes through three experimental scenarios. The results show that the combination of hand detection and landmark normalization achieved the best performance, with an accuracy of 0.915, precision of 0.913, recall of 0.915, and F1-score of 0.896, while improving accuracy by 14.8% compared with the approach without hand detection. These findings demonstrate that hand detection and landmark representation contribute significantly to improving system accuracy. The proposed approach has the potential to be implemented in a camera-based real-time BISINDO alphabet translation application to support more inclusive communication for deaf communities in Indonesia.

I. Pendahuluan

Komunitas tunarungu menghadapi hambatan dalam aspek bahasa dan komunikasi yang dapat memengaruhi proses pembelajaran dan interaksi. Penelitian sebelumnya menunjukkan bahwa siswa tunarungu masih mengalami keterbatasan dalam penguasaan kosakata serta membutuhkan media pembelajaran yang sesuai dengan karakteristik mereka [1]. Kondisi tersebut menunjukkan pentingnya pengembangan teknologi yang dapat mendukung komunikasi dan akses informasi bagi komunitas tunarungu secara lebih inklusif. Dalam konteks tersebut, pengenalan Bahasa Isyarat Indonesia (BISINDO) berbasis computer vision menjadi salah satu pendekatan yang berpotensi membantu menjembatani komunikasi antara komunitas tunarungu dan masyarakat umum.

Bahasa isyarat merupakan salah satu media komunikasi penting bagi penyandang disabilitas pendengaran. Namun, keterbatasan pemahaman masyarakat umum terhadap bahasa isyarat masih dapat menimbulkan kesenjangan komunikasi. Kondisi tersebut mendorong pengembangan sistem pengenalan bahasa isyarat berbasis teknologi sebagai salah satu pendekatan untuk mendukung interaksi antara komunitas tunarungu dan masyarakat luas. Perkembangan *computer vision* dan *deep learning* dalam beberapa tahun terakhir telah memungkinkan sistem pengenalan gestur dilakukan secara otomatis dengan tingkat akurasi yang tinggi [2]. Pendekatan berbasis visi *(vision-based)* menjadi pilihan utama karena mampu menangkap informasi gerakan secara alami tanpa memerlukan perangkat tambahan seperti sensor atau sarung tangan khusus, sehingga lebih fleksibel dan mudah diimplementasikan [3].

Dalam sistem pengenalan bahasa isyarat, khususnya pada pengenalan alfabet, terdapat beberapa tahapan penting yang membentuk pipeline sistem, yaitu deteksi tangan dan ekstraksi landmark. Deteksi tangan bertujuan untuk mengidentifikasi lokasi dan area tangan dalam citra atau video, sedangkan ekstraksi landmark digunakan untuk merepresentasikan bentuk dan posisi tangan dalam bentuk data numerik yang dapat diproses oleh model klasifikasi. Salah satu metode yang banyak digunakan adalah ekstraksi landmark menggunakan MediaPipe, yang mampu menghasilkan titik-titik koordinat penting *(keypoints)* secara real-time [4]. Representasi ini dinilai efektif karena dapat menyederhanakan kompleksitas citra menjadi data terstruktur yang relevan untuk proses klasifikasi gestur [5].

Meskipun demikian, performa sistem pengenalan gestur sangat dipengaruhi oleh kualitas hasil pada setiap tahapan tersebut. Proses deteksi tangan masih menghadapi berbagai tantangan, seperti *occlusion*, variasi pencahayaan, dan latar belakang yang kompleks, yang dapat menurunkan akurasi identifikasi area tangan [6]. Di sisi lain, ekstraksi landmark berbasis dua dimensi memiliki keterbatasan dalam merepresentasikan struktur tangan secara menyeluruh, terutama pada variasi sudut pandang dan pose tertentu [4]. Permasalahan ini menunjukkan bahwa kedua komponen tersebut memiliki peran penting dan saling mempengaruhi dalam menentukan keberhasilan sistem pengenalan alfabet bahasa isyarat.

Penelitian mengenai pengenalan bahasa isyarat telah berkembang melalui berbagai pendekatan berbasis *computer vision* dan *deep learning* dengan tujuan utama meningkatkan akurasi maupun efisiensi sistem klasifikasi. Pada pengenalan BISINDO, pendekatan Spatially Aware Body Gesture Recognition mengombinasikan Convolutional Neural Network (CNN), Graph Convolutional Network (GCN), dan Bidirectional Long Short-Term Memory (BiLSTM) untuk meningkatkan akurasi pengenalan gestur [2]. Penelitian lain menerapkan knowledge distillation pada beberapa arsitektur lightweight deep learning, seperti EfficientNet, MobileNetV3, dan ShuffleNetV2, guna memperoleh model yang lebih efisien tanpa penurunan performa yang signifikan [7]. Pada bahasa isyarat lain, penelitian memanfaatkan MediaPipe yang dipadukan dengan Feedforward Neural Network, PointNet, Vision Mamba, maupun CNN-BiLSTM. Ringkasan perbandingan penelitian terdahulu disajikan pada Tabel 1.

Tabel 1. Perbandingan Penelitian Terdahulu

Peneliti	Dataset	Metode	Fokus Penelitian	Keterbatasan
Sebastian et al., 2025 [2]	BISINDO	CNN + GCN + BiLSTM	Meningkatkan akurasi pengenalan gestur	Belum menganalisis kontribusi setiap tahapan pipeline.
Zhang et al., 2025 [4]	ASL (American Sign Language)	MediaPipe + PointNet + MLP	Meningkatkan pengenalan gesture (ASL)	Kompleksitas model relatif tinggi
Lucian et al., 2025 [7]	BISINDO	EfficientNet, MobileNetV3, ShuffleNetV2 + Knowledge Distillation	Meningkatkan efisiensi komputasi	Fokus pada optimasi model klasifikasi.
Altaher et al., 2025 [8]	ASL (American Sign Language)	Vision Mamba	Perbandingan arsitektur deep learning	Evaluasi berfokus pada akurasi model.
Ranee et al., 2025 [9]	KSL (Khasi Sign Language)	MediaPipe + Feedforward Neural Network	Klasifikasi berbasis *landmark*	Tidak mengevaluasi pengaruh deteksi tangan terhadap performa sistem.
Penelitian ini	BISINDO	YOLOv8 + MediaPipe + MLP	Analisis kontribusi deteksi tangan dan normalisasi landmark	Mengevaluasi kontribusi setiap komponen pipeline.

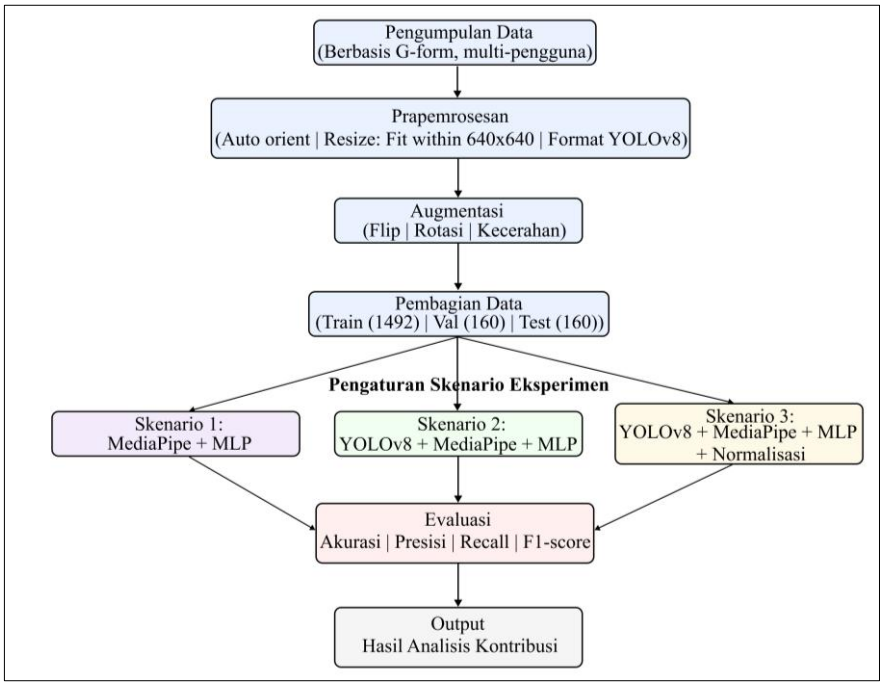
Berdasarkan penelitian-penelitian tersebut, sebagian besar penelitian mengevaluasi keberhasilan sistem berdasarkan peningkatan akurasi model klasifikasi atau efisiensi komputasi melalui pengembangan arsitektur *deep learning*. Belum banyak penelitian yang menganalisis kontribusi setiap tahapan dalam pipeline, khususnya deteksi tangan dan normalisasi landmark, terhadap performa pengenalan alfabet BISINDO. Oleh karena itu, penelitian ini menganalisis kontribusi deteksi tangan, ekstraksi landmark, dan normalisasi landmark terhadap performa pengenalan alfabet BISINDO melalui tiga skenario eksperimen. Fokus penelitian bukan pada pengembangan arsitektur baru, melainkan pada identifikasi pengaruh setiap komponen terhadap kualitas representasi fitur dan performa klasifikasi.

II. Metode

Penelitian ini menggunakan pendekatan kuantitatif dengan tujuan untuk menganalisis kontribusi masing-masing komponen dalam pipeline pengenalan alfabet Bahasa Isyarat Indonesia (BISINDO). Sistem yang dikembangkan terdiri dari tiga tahapan utama, yaitu deteksi tangan, ekstraksi landmark, dan klasifikasi gestur.

A. Prosedur Penelitian

Penelitian ini dilakukan melalui beberapa tahapan utama yang meliputi pengumpulan data, prapemrosesan dan augmentasi data, ekstraksi fitur, pengembangan model, pengujian melalui skenario eksperimen, evaluasi kinerja, serta analisis hasil. Alur penelitian yang dilakukan ditunjukkan pada Gambar 1.



Gambar 1. Diagram Penelitian

Tahap awal dimulai dengan proses pengumpulan data berupa citra gestur alfabet BISINDO yang diperoleh secara mandiri. Selanjutnya, data yang telah dikumpulkan melalui tahap prapemrosesan dan augmentasi untuk meningkatkan kualitas serta variasi data. Dataset kemudian dibagi menjadi data latih, validasi, dan uji sebelum digunakan dalam proses eksperimen.

Pada tahap eksperimen, dilakukan perbandingan tiga skenario yang dirancang untuk mengevaluasi kontribusi masing-masing komponen dalam pipeline, yaitu tanpa deteksi tangan, dengan deteksi tangan, serta dengan tambahan normalisasi fitur. Setiap skenario diuji menggunakan dataset yang sama untuk memastikan perbandingan yang adil.

Tahap akhir penelitian adalah evaluasi kinerja menggunakan metrik klasifikasi serta analisis hasil untuk mengidentifikasi kontribusi setiap komponen terhadap performa sistem secara keseluruhan.

B. Deskripsi Dataset

Dataset yang digunakan dalam penelitian ini merupakan dataset yang dikumpulkan secara mandiri dengan total 1066 citra gestur alfabet BISINDO yang mencakup 26 kelas (A–Z). Distribusi jumlah citra pada setiap kelas seimbang, di mana setiap alfabet memiliki 41 citra, sehingga potensi bias akibat ketidakseimbangan kelas dapat diminimalkan. Distribusi dataset untuk setiap kelas disajikan pada Tabel 2.

Tabel 2. Distribusi Dataset Alfabet BISINDO

Huruf	Train	Valid	Test	Total
A	30	4	7	41
B	29	6	6	41
C	27	4	10	41
D	28	4	9	41
E	24	6	11	41
F	29	8	4	41
G	32	5	4	41
H	31	5	5	41
I	26	8	7	41
J	26	7	8	41
K	29	5	7	41
L	31	5	5	41
M	28	8	5	41
N	33	2	6	41
O	30	9	2	41
P	31	5	5	41
Q	31	5	5	41
R	29	9	3	41
S	27	6	8	41
T	33	3	5	41
U	28	9	4	41
V	24	8	9	41
W	25	9	7	41
X	24	11	6	41
Y	31	4	6	41
Z	30	5	6	41
Total	**746**	**160**	**160**	**1066**

Data dikumpulkan dalam kondisi yang bervariasi, meliputi perbedaan pencahayaan, latar belakang, serta sudut pengambilan gambar. Selain itu, sebagian besar gestur dalam dataset menggunakan dua tangan, meskipun terdapat beberapa kelas yang menggunakan satu tangan. Variasi ini bertujuan untuk meningkatkan kemampuan generalisasi model terhadap kondisi nyata. Dataset kemudian dibagi menjadi data latih sebanyak 746 citra, data validasi sebanyak 160 citra, dan data uji sebanyak 160 citra.

C. Prapemrosesan dan Augmentasi Data

Sebelum digunakan dalam proses pelatihan, seluruh citra melalui tahap prapemrosesan untuk memastikan konsistensi data. Tahap prapemrosesan meliputi *auto-orient* untuk menyesuaikan orientasi gambar secara otomatis serta *resizing* citra ke ukuran 640 × 640 piksel agar sesuai dengan kebutuhan model berbasis YOLOv8.

Selain prapemrosesan, dilakukan augmentasi data menggunakan platform Roboflow untuk meningkatkan variasi dataset dan mengurangi risiko overfitting. Augmentasi data dilakukan menggunakan Roboflow dengan *horizontal flipping*, rotasi ±10°, dan penyesuaian *brightness* ±15% untuk meningkatkan variasi orientasi, sudut pengambilan gambar, dan pencahayaan tanpa mengubah karakteristik utama gestur. Pendekatan serupa telah banyak digunakan dalam penelitian *deep learning* berbasis citra untuk meningkatkan kemampuan generalisasi model tanpa merusak karakteristik utama data [10].

Augmentasi hanya diterapkan pada data latih, sedangkan data validasi dan data uji tetap digunakan dalam kondisi asli untuk memastikan evaluasi yang objektif dan tidak bias. Pendekatan ini umum digunakan dalam pengembangan model *machine learning* untuk memastikan bahwa performa yang dihasilkan mencerminkan kemampuan model dalam menghadapi data nyata [11].

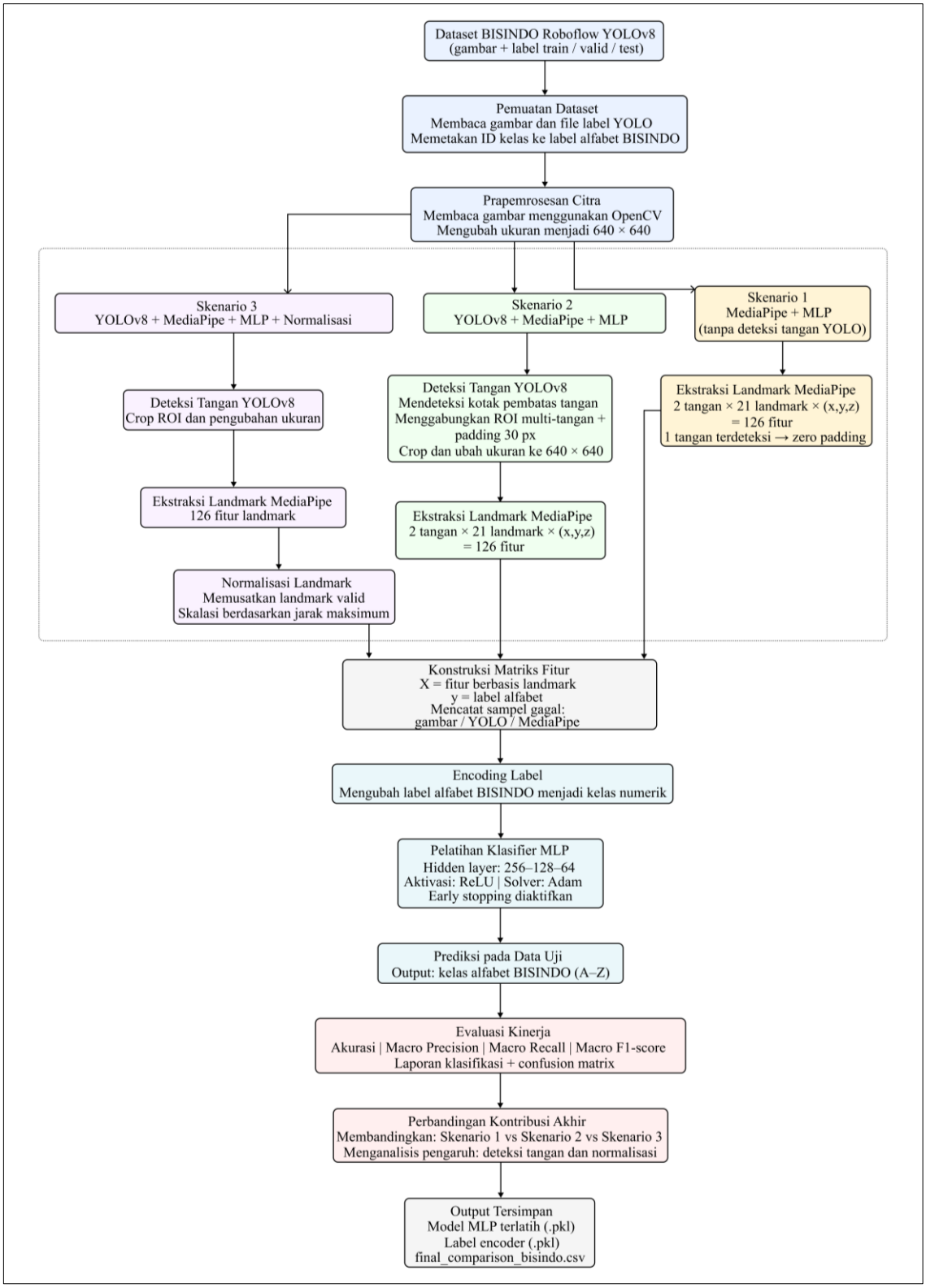
D. Pembagian Dataset

Dataset dibagi dengan rasio awal 70% untuk pelatihan, 15% untuk validasi, dan 15% untuk pengujian. Pembagian tersebut dipilih karena merupakan proporsi yang umum digunakan dalam pengembangan model *machine learning*, sehingga mampu memberikan keseimbangan antara jumlah data pelatihan yang memadai untuk proses pembelajaran model dan data evaluasi yang cukup untuk mengukur kemampuan generalisasi model.

Setelah proses augmentasi, jumlah data latih meningkat dua kali lipat dari 746 menjadi 1492 citra. Proporsi dataset berubah menjadi sekitar 82% untuk data latih (1492 citra), 9% untuk data validasi (160 citra), dan 9% untuk data uji (160 citra). Pembagian ini tetap mempertahankan data validasi dan uji dalam kondisi asli untuk memastikan evaluasi yang adil dan tidak bias.

E. Metode yang Diusulkan

Metode yang diusulkan terdiri dari tiga tahap utama, yaitu deteksi tangan, ekstraksi landmark, dan klasifikasi gestur. Alur lengkap metode yang diusulkan ditunjukkan pada Gambar 2.



Gambar 2. Diagram Metode yang Diusulkan

Pendekatan ini merupakan standar dalam sistem pengenalan gestur modern karena memisahkan proses lokalisasi, representasi fitur, dan klasifikasi. Integrasi beberapa teknik dalam satu sistem juga terbukti meningkatkan akurasi secara signifikan [12].

1) Deteksi Tangan

Deteksi tangan dilakukan menggunakan model YOLOv8 yang mampu melokalisasi area tangan dalam citra input. *Output* dari tahap ini berupa bounding box yang digunakan untuk melakukan pemotongan area tangan. Kinerja deteksi diukur menggunakan *Intersection over Union* (IoU):

$$IoU = \frac{Area(B_{Pred} \cap B_{gt}}{Area(B_{Pred} \cup B_{gt}} \quad (1)$$

Pada Persamaan (1), B_{pred} merupakan bounding box hasil prediksi dan B_{gt} merupakan ground truth. IoU merupakan metrik standar dalam evaluasi deteksi objek dan menjadi dasar perhitungan mAP (*mean Average Precision*) [3]. IoU digunakan secara internal oleh YOLO selama perhitungan mAP dan tidak dilaporkan sebagai metrik terpisah. Metrik ini mengukur tingkat kesesuaian antara *bounding box* prediksi dan *ground truth*, yang menjadi indikator utama performa deteksi objek pada model YOLO [13].

2) Ekstraksi Landmark

Setelah proses deteksi, area tangan diproses menggunakan MediaPipe untuk mengekstraksi 21 titik landmark tangan dalam koordinat tiga dimensi,sebagaimana ditunjukkan pada Persamaan (2):

$$pi = (xi, yi, zi) \quad (2)$$

Landmark yang diperoleh kemudian dinormalisasi menggunakan koordinat pergelangan tangan sebagai titik acuan sehingga posisi landmark menjadi relatif terhadap tangan. Pendekatan serupa telah digunakan pada penelitian sebelumnya melalui *localization* dan *zoom normalization* pada landmark MediaPipe untuk meningkatkan performa pengenalan pose tangan [14]. Normalisasi digunakan untuk mengurangi variasi akibat posisi dan ukuran tangan, sebagaimana dirumuskan pada Persamaan (3):

$$p'i = \frac{pi - p0}{max(||pi - p0||)} \quad (3)$$

Proses normalisasi ini penting untuk menghasilkan representasi fitur yang stabil dan telah digunakan dalam berbagai penelitian berbasis landmark untuk meningkatkan performa klasifikasi gestur. Penggunaan landmark 3D terbukti memberikan representasi yang lebih informatif dibandingkan 2D, terutama dalam menangkap orientasi dan struktur spasial tangan [15].

3) Klasifikasi Gestur

Fitur landmark digunakan sebagai input untuk model *Multilayer Perceptron* (MLP). MLP merupakan model jaringan saraf yang efektif dalam memodelkan hubungan non-linear antar fitur [16]. Pada penelitian ini, model diimplementasikan menggunakan MLPClassifier dari pustaka Scikit-learn dengan konfigurasi hyperparameter sebagaimana ditunjukkan pada Tabel 3.

Tabel 3. Konfigurasi Hyperparameter Model MLP

Parameter	Nilai
Model	MLPClassifier (Scikit-learn)
Hidden Layer	(256, 128, 64)
Activation Function	ReLU
Optimizer	Adam
Learning Rate	0,001
Maximum Iteration	700
Early Stopping	True
Validation Fraction	0,15
n_iter_no_change	30
Batch Size	Auto (default)
Random State	42

Arsitektur MLP menggunakan tiga hidden layer berukuran 256, 128, dan 64 yang disusun secara bertahap. Konfigurasi ini dipilih untuk memberikan keseimbangan antara kapasitas model dan kompleksitas komputasi pada dataset yang digunakan. Pada layer *output* digunakan fungsi *Softmax* untuk menghasilkan probabilitas kelas, sebagaimana ditunjukkan pada Persamaan (4):

$$P(y = i|x) = \frac{e^{Zi}}{\sum_j e^{Zj}} \quad (4)$$

Selanjutnya, proses pelatihan model menggunakan fungsi *cross-entropy loss* untuk mengukur perbedaan antara distribusi probabilitas hasil prediksi dan label sebenarnya, sebagaimana dirumuskan pada Persamaan (5):

$$L = -\sum y \log(\hat{y}) \quad (5)$$

Fungsi *loss* ini digunakan untuk mengukur perbedaan antara distribusi prediksi dan label sebenarnya. Pendekatan ini merupakan metode standar dalam klasifikasi berbasis jaringan saraf dan telah banyak digunakan dalam sistem pengenalan gestur berbasis landmark [9].

F. Skenario Eksperimen

Dalam menganalisis kontribusi masing-masing komponen dalam pipeline yang diusulkan, dilakukan serangkaian eksperimen komparatif yang dirancang secara sistematis. Eksperimen ini bertujuan untuk mengevaluasi pengaruh setiap tahap, yaitu deteksi tangan, ekstraksi landmark, dan klasifikasi, terhadap performa sistem secara kuantitatif.

Skenario pertama menggunakan pendekatan MediaPipe + MLP tanpa deteksi tangan, di mana landmark diekstraksi langsung dari citra input tanpa proses isolasi area tangan. Skenario ini digunakan sebagai skenario dasar untuk mengevaluasi performa sistem tanpa tahap deteksi. Skenario kedua menggunakan pipeline YOLOv8 + MediaPipe + MLP, di mana deteksi tangan digunakan untuk mengisolasi region of interest sebelum ekstraksi landmark. Pendekatan ini bertujuan untuk mengevaluasi kontribusi deteksi tangan terhadap kualitas fitur. Skenario ketiga merupakan pengembangan dari skenario kedua dengan menambahkan proses normalisasi fitur landmark, yaitu pipeline YOLOv8 + MediaPipe + MLP dengan normalisasi fitur. Tahap ini bertujuan untuk mengurangi variasi akibat perbedaan posisi dan ukuran tangan.

Seluruh skenario diuji menggunakan dataset dan pembagian data yang sama untuk memastikan perbandingan yang adil. Perbandingan antar skenario digunakan untuk mengevaluasi kontribusi tahap deteksi tangan dan normalisasi terhadap performa sistem.

G. Metrik Evaluasi

Metrik evaluasi seperti accuracy, precision, recall, dan F1-score merupakan standar dalam evaluasi model klasifikasi dan telah digunakan secara luas dalam berbagai penelitian *machine learning* [9]. Performa sistem dievaluasi menggunakan metrik berikut:

1) Accuracy

Accuracy digunakan untuk mengukur proporsi prediksi yang benar terhadap seluruh data. Pada kasus klasifikasi *multi-class* alfabet BISINDO, accuracy secara praktis dihitung sebagai rasio antara jumlah prediksi yang benar terhadap total sampel valid yang berhasil diproses oleh sistem:

$$Accuracy = \frac{Number\ of\ Correct\ Predictions}{Total\ Number\ of\ Predictions} \quad (6)$$

Pada Persamaan (6), *Number of Correct Predictions* menunjukkan jumlah sampel yang diklasifikasikan dengan benar, sedangkan *Total Number of Predictions* menunjukkan jumlah seluruh sampelvalid yang diproses oleh sistem. Metrik ini memberikan gambaran umum terhadap performa model dalam mengklasifikasikan gestur secara keseluruhan dan banyak digunakan dalam evaluasi sistem klasifikasi berbasis *machine learning*.

2) Precision

Precision digunakan untuk mengukur tingkat ketepatan prediksi positif yang dihasilkan oleh model, sebagaimana dirumuskan pada Persamaan (7):

$$Precision = \frac{TP}{TP+FP} \quad (7)$$

Pada persamaan (7), TP *(true positive)* menunjukkan jumlah prediksi positif yang benar, sedangkan FP *(false positive)* menunjukkan jumlah sampel yang diprediksi positif tetapi sebenarnya termasuk kelas lain. Metrik ini penting untuk memastikan bahwa prediksi yang dihasilkan benar-benar relevan, terutama pada sistem klasifikasi multi-kelas.

3) Recall

Recall digunakan untuk mengukur kemampuan model dalam mendeteksi seluruh sampel positif pada setiap kelas, sebagaimana dirumuskan pada Persamaan (8):

$$Recall = \frac{TP}{TP+FN} \quad (8)$$

Pada Persamaan (8), TP *(false negative)* menunjukkan jumlah sampel positif yang sebenarnya tetapi diprediksi sebagai kelas lain. Nilai recall yang tinggi menunjukkan bahwa model mampu mengenali sebagian besar gestur dengan benar,sehingga penting dalam sistem pengenalan bahasa isyarat.

4) F1-score

F1-score merupakan *harmonic mean* dari precision dan recall yang digunakan untuk memperoleh ukuran keseimbangan antara kedua metrik tersebut, sebagaimana dirumuskan pada Persamaan (9):

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (9)$$

Metrik-metrik tersebut digunakan untuk memberikan evaluasi yang objektif terhadap performa model pada tugas klasifikasi multi-kelas. Selain itu, digunakan confusion matrix untuk menganalisis kesalahan klasifikasi antar kelas. Analisis ini membantu dalam mengidentifikasi kelas yang sering mengalami kesalahan klasifikasi.

5) Mean Average Precision (mAP)

Tahap deteksi tangan, digunakan metrik *mean Average Precision* (mAP), yang merupakan standar dalam evaluasi model deteksi objek seperti YOLO:

$$mAP = \frac{1}{N} \sum_{i=1}^{N} APi \quad (10)$$

Pada Persamaan (10), N menunjukkan jumlah kelas yang dievaluasi, sedangkan APi menunjukkan nilai *Average Precision* pada kelas ke-(i). mAP menggabungkan precision dan recall dalam berbagai threshold, sehingga memberikan evaluasi yang lebih komprehensif terhadap performa model deteksi [13].

III. Hasil dan Pembahasan

A. Kinerja Deteksi Tangan

Tahap awal penelitian difokuskan pada deteksi tangan menggunakan YOLOv8 untuk mengisolasi area tangan sebagai *region of interest* (ROI) sebelum proses ekstraksi landmark menggunakan MediaPipe. Tahap ini bertujuan agar sistem hanya memproses area gestur yang relevan sehingga gangguan dari latar belakang dapat diminimalkan.

Berdasarkan hasil pelatihan, model YOLOv8 menunjukkan performa yang baik dalam mendeteksi tangan pada dataset alfabet Bahasa Isyarat Indonesia (BISINDO) A–Z. Model terbaik diperoleh pada *epoch* ke-42 dan disimpan sebagai *best.pt*. Selain performa akurasi, kecepatan inferensi juga menjadi faktor penting dalam sistem pengenalan gestur real-time. YOLOv8 menghasilkan *inference speed* sebesar 255.2 ms/image sehingga masih cukup efisien untuk diterapkan pada sistem pengenalan gestur berbasis kamera secara real-time. Hasil evaluasi model YOLOv8 dapat dilihat pada Tabel 4.

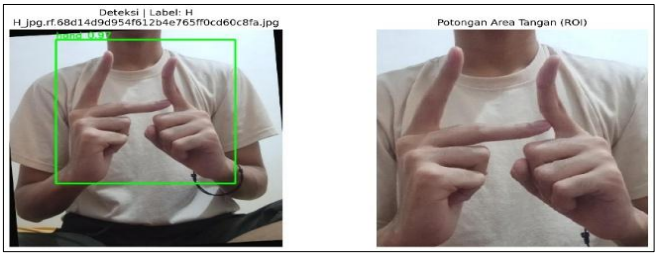
Tabel 4. Kinerja Deteksi Tangan menggunakan YOLOv8

Metrik	Nilai
Precision	0.935
Recall	0.923
mAP@50	0.978
mAP@50-95	0.861
Inference Speed	255.2 ms/image
Best Epoch	42

Berdasarkan hasil pengujian, penggunaan YOLOv8 mampu memfokuskan proses ekstraksi fitur pada area tangan (region of interest) sehingga gangguan akibat latar belakang dapat diminimalkan. Nilai precision sebesar 0.935 dan mAP@50 sebesar 0.978 menunjukkan bahwa model memiliki kemampuan deteksi yang tinggi, sedangkan nilai mAP@50–95 sebesar 0.861 menunjukkan performa yang tetap konsisten pada berbagai tingkat IoU. Hasil ini menunjukkan bahwa kualitas deteksi tangan berkontribusi terhadap peningkatan kualitas landmark yang diekstraksi pada tahap berikutnya sehingga mendukung proses klasifikasi alfabet BISINDO secara lebih akurat.

Gambar 3 menunjukkan contoh hasil deteksi tangan menggunakan YOLOv8. Pada kondisi deteksi berhasil, model mampu melokalisasi area tangan dengan tepat dan menghasilkan *crop* ROI yang jelas untuk proses ekstraksi landmark. Namun, pada beberapa kasus masih ditemukan kegagalan deteksi, terutama pada gestur dengan posisi tangan simetris, *overlap* antar jari, atau bentuk gestur yang kompleks. Kondisi tersebut menyebabkan proses ROI *extraction* gagal dilakukan dan berkontribusi terhadap berkurangnya jumlah sampel valid pada tahap klasifikasi.

(a) deteksi berhasil



(b) deteksi berhasil

(c) deteksi gagal

Gambar 3. Contoh Hasil Deteksi Tangan Berhasil dan Gagal Menggunakan YOLOv8

B. Perbandingan Skenario Eksperimen

Untuk mengevaluasi kontribusi masing-masing komponen dalam pipeline pengenalan alfabet BISINDO, penelitian ini membandingkan tiga skenario eksperimen, yaitu: (1) MediaPipe + MLP tanpa deteksi tangan, (2) YOLOv8 + MediaPipe + MLP tanpa normalisasi, dan (3) YOLOv8 + MediaPipe + MLP dengan normalisasi landmark. Evaluasi dilakukan menggunakan metrik accuracy, precision, recall, dan F1-score. Hasil perbandingan ketiga skenario ditunjukkan pada Tabel 5.

Tabel 5. Perbandingan Antar Skenario Eksperimen

Metode	Accuracy	Precision	Recall	F1-score
MediaPipe + MLP	0.797	0.818	0.763	0.757
YOLOv8 + MediaPipe + MLP	0.862	0.847	0.858	0.833
YOLOv8 + MediaPipe + MLP + Normalisasi	0.915	0.913	0.915	0.896

Berdasarkan hasil eksperimen, skenario ketiga menghasilkan performa terbaik dengan accuracy sebesar 0.915, precision 0.913, recall 0.915, dan F1-score 0.896. Dibandingkan skenario dasar, penambahan deteksi tangan menggunakan YOLOv8 meningkatkan accuracy dari 0.797 menjadi 0.862 atau sekitar 8.2%. Selanjutnya, penambahan proses normalisasi landmark kembali meningkatkan accuracy menjadi 0.915 atau sekitar 6.1% dibandingkan skenario kedua. Secara keseluruhan, kombinasi deteksi tangan dan normalisasi landmark menghasilkan peningkatan accuracy sekitar 14.8% dibandingkan pendekatan dasar.

Peningkatan performa pada skenario kedua menunjukkan bahwa proses deteksi tangan menggunakan YOLOv8 mampu menghasilkan region of interest (ROI) yang lebih representatif sebelum ekstraksi landmark dilakukan. Penambahan normalisasi landmark pada skenario ketiga semakin meningkatkan konsistensi representasi fitur sehingga model MLP mampu membedakan setiap kelas alfabet BISINDO dengan lebih baik.

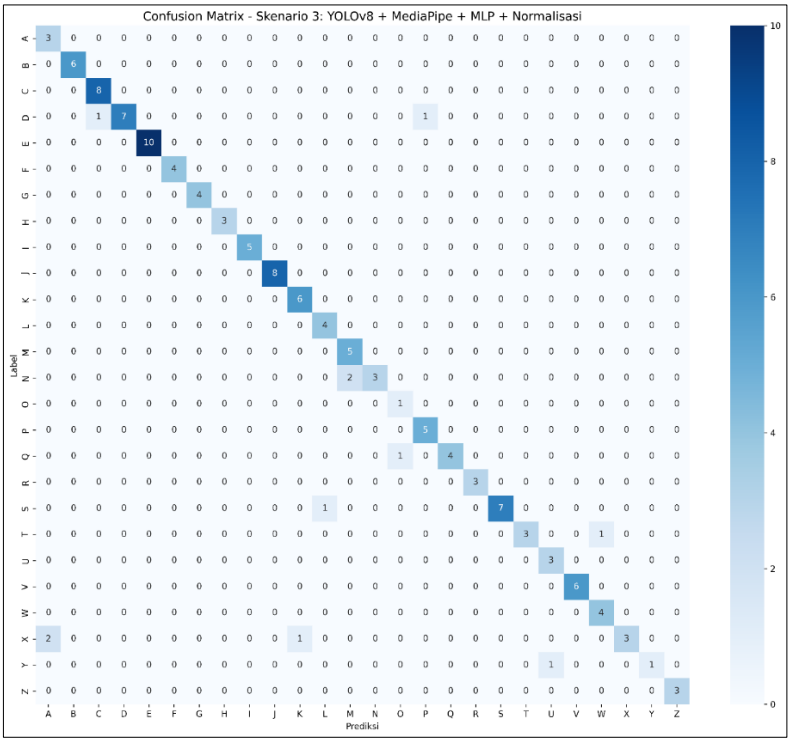
Hasil penelitian ini konsisten dengan penelitian Sebastian et al. [2] yang mengembangkan sistem pengenalan BISINDO menggunakan kombinasi CNN, GCN, dan BiLSTM berbasis MediaPipe dengan accuracy terbaik sebesar 90% pada pengenalan kata (word-level). Sementara itu, penelitian ini memperoleh accuracy sebesar 0.915 (91.5%) pada pengenalan alfabet BISINDO. Perbedaan tersebut kemungkinan dipengaruhi oleh perbedaan karakteristik dataset, tingkat kompleksitas gestur, jumlah kelas, dan arsitektur yang digunakan. Penelitian Sebastian et al. berfokus pada pengenalan gestur dinamis berbasis spasial-temporal, sedangkan penelitian ini menganalisis kontribusi setiap tahapan pipeline. Temuan ini mendukung tujuan

penelitian, yaitu menganalisis kontribusi setiap komponen pipeline dalam meningkatkan performa pengenalan alfabet BISINDO.

C. Analisis Confusion Matrix dan Kesalahan Klasifikasi

Confusion matrix digunakan untuk melihat distribusi prediksi benar dan kesalahan klasifikasi pada masing-masing kelas alfabet BISINDO. Analisis dilakukan pada skenario terbaik, yaitu YOLOv8 + MediaPipe + MLP + Normalisasi. Hasil confusion matrix pada Gambar 4 menunjukkan bahwa sebagian besar prediksi terkonsentrasi pada diagonal utama, yang mengindikasikan bahwa model mampu mengenali mayoritas alfabet BISINDO secara konsisten. Beberapa kelas, seperti A, B, C, E, F, G, H, I, J, K, L, M, O, P, R, S, T, U, V, W, dan Z, berhasil diklasifikasikan dengan benar pada seluruh sampel uji sehingga tidak menunjukkan kesalahan klasifikasi. Hasil tersebut menunjukkan bahwa representasi landmark pada kelas-kelas tersebut memiliki karakteristik yang cukup berbeda sehingga dapat dipelajari dengan baik oleh model MLP.

Meskipun demikian, beberapa kesalahan klasifikasi masih ditemukan pada kelas yang memiliki konfigurasi tangan serupa. Kesalahan terbesar terjadi pada huruf N, yang sebanyak dua sampel diprediksi sebagai M, sedangkan tiga sampel lainnya berhasil dikenali dengan benar. Selain itu, huruf X diprediksi sebagai A sebanyak dua sampel dan sebagai K sebanyak satu sampel, sehingga hanya tiga dari enam sampel yang berhasil diklasifikasikan dengan benar. Kesalahan lain terjadi pada huruf D yang masing-masing satu sampel diprediksi sebagai C dan P, huruf Q yang satu sampel diprediksi sebagai O, serta huruf Y yang satu sampel diprediksi sebagai U. Pola tersebut menunjukkan bahwa kesalahan klasifikasi terutama terjadi pada pasangan huruf yang memiliki konfigurasi jari serupa.



Gambar 4. Confusion Matrix pada Skenario dengan Performa Terbaik

Selain kesalahan klasifikasi, kegagalan juga ditemukan pada tahap deteksi tangan dan ekstraksi landmark. Pada beberapa sampel, YOLOv8 berhasil mendeteksi area tangan namun MediaPipe gagal menghasilkan landmark secara optimal sehingga proses klasifikasi tidak dapat dilanjutkan. Pada kasus lain, YOLOv8 tidak berhasil melakukan lokalisasi area tangan sehingga proses ekstraksi landmark gagal dilakukan. Kegagalan tersebut umumnya dipengaruhi oleh overlap antar jari, ukuran gestur yang terlalu kecil, atau posisi tangan yang kurang jelas pada citra input.

Gambar 5 menunjukkan beberapa contoh kesalahan klasifikasi dan kegagalan sistem yang ditemukan selama pengujian. Kesalahan pada tahap deteksi tangan akan menyebabkan proses ekstraksi landmark tidak berjalan dengan baik, sedangkan landmark yang kurang representatif dapat meningkatkan kemungkinan terjadinya kesalahan klasifikasi. Dengan demikian, performa sistem tidak hanya ditentukan oleh model klasifikasi, tetapi juga oleh kualitas keluaran pada setiap tahapan sebelumnya.

(a) Label: N → Prediksi: M

(Klasifikasi salah)

(b) Label: C

(Landmark gagal)

(c) Label: X

(Deteksi YOLO gagal)

Gambar 5. Contoh Kesalahan Klasifikasi pada Sistem yang Diusulkan: (a) Klasifikasi salah, (b) Landmark gagal, (c) Deteksi YOLO gagal

D. Diskusi

Hasil penelitian menunjukkan bahwa kualitas representasi fitur pada tahap prapemrosesan berperan dalam meningkatkan performa pengenalan alfabet BISINDO. Deteksi tangan menggunakan YOLOv8 menghasilkan region of interest yang lebih representatif, sedangkan normalisasi landmark mengurangi variasi koordinat akibat perbedaan posisi dan ukuran tangan. Kondisi tersebut menghasilkan representasi fitur yang lebih konsisten sehingga MLP dapat mempelajari karakteristik setiap kelas dengan lebih baik.

Hasil penelitian ini sejalan dengan Sebastian et al. [2] yang memperoleh accuracy sebesar 90% pada pengenalan kata BISINDO menggunakan kombinasi CNN, GCN, dan BiLSTM. Penelitian ini memperoleh accuracy sebesar 91.5% pada pengenalan alfabet BISINDO. Perbedaan tersebut tidak dapat dibandingkan secara langsung karena terdapat perbedaan pada jenis tugas, dataset, jumlah kelas, dan arsitektur model. Meskipun demikian, hasil eksperimen penelitian ini memberikan bukti kuantitatif bahwa deteksi tangan dan normalisasi landmark memberikan kontribusi terhadap peningkatan performa.

Kesalahan klasifikasi terutama terjadi pada pasangan huruf dengan konfigurasi tangan yang serupa, seperti N–M, X–A, X–K, D–C, D–P, Q–O, dan Y–U. Hal ini menunjukkan bahwa kemiripan morfologi gestur dapat menghasilkan representasi landmark yang berdekatan sehingga meningkatkan kemungkinan kesalahan klasifikasi. Temuan tersebut mendukung prinsip pattern recognition bahwa kualitas dan daya diskriminatif representasi fitur berpengaruh terhadap kemampuan model dalam membedakan pola antar kelas. Dalam konteks computer vision, hasil ini menunjukkan bahwa tahap lokalisasi dan normalisasi dapat memengaruhi kualitas fitur sebelum proses klasifikasi.

Dari sisi implementasi, YOLOv8 menghasilkan inference speed sebesar 255.2 ms/image. Penambahan deteksi tangan dan normalisasi meningkatkan kompleksitas komputasi, tetapi memberikan peningkatan accuracy sebesar 14.8% dibandingkan skenario dasar. Sistem masih memiliki keterbatasan berupa kegagalan deteksi tangan dan ekstraksi landmark, serta dataset yang terbatas pada alfabet BISINDO statis A–Z.

Generalisasi terhadap pengguna baru, perangkat kamera, dan kondisi pencahayaan yang berbeda juga belum dievaluasi secara khusus. Penelitian selanjutnya dapat memperluas dataset lintas pengguna dan perangkat, meningkatkan robustness deteksi, serta mengembangkan pengenalan gestur dinamis berbasis LSTM atau Transformer.

IV. Kesimpulan dan saran

Penelitian ini berhasil menganalisis kontribusi deteksi tangan dan ekstraksi landmark terhadap performa sistem pengenalan alfabet Bahasa Isyarat Indonesia (BISINDO) melalui tiga skenario eksperimen yang menggunakan kombinasi YOLOv8, MediaPipe, MLP, dan normalisasi landmark. Hasil penelitian menunjukkan bahwa deteksi tangan menggunakan YOLOv8 dan normalisasi landmark memberikan kontribusi yang saling melengkapi dalam meningkatkan performa sistem, dengan accuracy terbaik sebesar 0.915 pada skenario YOLOv8 + MediaPipe + MLP + normalisasi landmark. Temuan ini menunjukkan bahwa peningkatan performa sistem tidak hanya ditentukan oleh model klasifikasi, tetapi juga oleh kualitas representasi fitur yang dihasilkan pada tahap deteksi tangan dan prapemrosesan landmark. Secara teoretis, temuan ini mendukung konsep dalam pattern recognition dan computer vision bahwa kualitas representasi fitur merupakan faktor penting dalam keberhasilan proses klasifikasi. Oleh karena itu, penelitian ini tidak hanya meningkatkan performa pengenalan alfabet BISINDO, tetapi juga menunjukkan bahwa optimasi setiap tahapan pipeline merupakan aspek penting dalam pengembangan sistem pengenalan bahasa isyarat berbasis *computer vision*. Meskipun memberikan hasil yang baik, penelitian ini masih memiliki keterbatasan pada kegagalan deteksi tangan, kegagalan ekstraksi landmark akibat overlap antar jari, serta penggunaan dataset yang masih terbatas pada alfabet BISINDO statis A–Z. Selain itu, kemampuan generalisasi model terhadap pengguna baru, perangkat kamera yang berbeda, maupun kondisi pencahayaan yang lebih beragam belum dievaluasi secara khusus. Oleh karena itu, penelitian selanjutnya dapat difokuskan pada peningkatan *robustness* deteksi tangan pada kondisi visual yang lebih kompleks, pengembangan metode ekstraksi fitur yang lebih diskriminatif, perluasan dataset lintas pengguna dan lintas perangkat, serta penerapan model berbasis temporal sequence, seperti LSTM atau Transformer, untuk mendukung pengenalan gestur BISINDO dinamis secara lebih natural dan real-time.

Ucapan Terima Kasih (Opsional)

Penulis mengucapkan terima kasih kepada Program Studi Teknik Informatika, Fakultas Teknik, UNISMUH Makassar yang telah mendukung pelaksanaan penelitian ini.

Daftar Pustaka

[1] I. Yuniarsih, A. Huda, and U. N. Malang, “Pengembangan Multimedia Interaktif Berbasis Android untuk Meningkatkan Penguasaan Kosakata Siswa Tunarungu,” vol. 8, no. November, pp. 92–97, 2022, doi: 10.17977/um031v8i22022p92-97.

[2] C. Sebastian *et al.*, “Indonesian Sign Language (BISINDO) Recognition Using Spatially Aware Body Gesture Recognition,” *Procedia Comput. Sci.*, vol. 269, pp. 1002–1011, 2025, doi: 10.1016/j.procs.2025.09.042.

[3] R. Jalayer, “Robotics and Computer-Integrated Manufacturing A review on deep learning for vision-based hand detection , hand segmentation and hand gesture recognition in human – robot interaction,” *Robot. Comput. Integr. Manuf.*, vol. 97, no. May 2025, p. 103110, 2026, doi: 10.1016/j.rcim.2025.103110.

[4] M. Perceptron, A. S. Language, A. S. Language, and N. A. Deaf, “Measurement : Sensors 3D geometric features based real-time American sign language recognition using PointNet and MLP with MediaPipe hand skeleton detection,” vol. 38, pp. 2–7, 2025, doi: 10.1016/j.measen.2024.101697.

[5] J. H. Meghana, H. K. Chethan, K. S. S. Kumar, S. P. S. Prakash, and S. P. S. Prakash, “Comprehensive Analysis of Pose Estimation and Machine Learning Classifiers for Precise Yoga Pose Detection and Classification,” *Procedia Comput. Sci.*, vol. 258, pp. 3345–3356, 2025, doi: 10.1016/j.procs.2025.04.592.

[6] X. Chai, M. Zhao, J. Li, and J. Li, “Image small target detection in complex traffic scenes based on Yolov8 multiscale feature fusion,” *Alexandria Eng. J.*, vol. 126, no. May, pp. 578–590, 2025, doi: 10.1016/j.aej.2025.04.105.

[7] D. Lucian, C. Alexander, Y. Raffel, C. Alexander, and Y. Raffel, “Improving Indonesian sign language recognition using lightweight deep learning architectures with knowledge distillation method,” *Procedia Comput. Sci.*, vol. 269, pp. 618–629, 2025, doi: 10.1016/j.procs.2025.09.005.

[8] A. Salem, C. Bang, B. Alsharif, and A. Altaher, “Franklin Open Mamba vision models : Automated American sign language recognition,” *Franklin Open*, vol. 10, no. March 2024, p. 100224, 2025, doi: 10.1016/j.fraope.2025.100224.

[9] L. A. Ranee, F. L. Marshillong, S. A. Lyndoh, L. Amory, and F. Lyngdoh, “Khasi Sign Language Recognition using Google ’ s Mediapipe and and Deep Learning Feedforward Neural Network Approach,” *Procedia Comput. Sci.*, vol. 258, pp. 3619–3629, 2025, doi: 10.1016/j.procs.2025.04.617.

[10] V. Pradip and P. S. Chakurkar, “MethodsX Towards safer environments : A YOLO and MediaPipe-based

human fall detection system with alert automation," *MethodsX*, vol. 15, no. September, p. 103623, 2025, doi: 10.1016/j.mex.2025.103623.

[11] J. Badzoka, C. Kappacher, J. Lau, and C. W. Huck, "Accelerating microplastic analysis with machine learning : Utilizing multi-layer perceptron (MLP) for classification of hyperspectral data," vol. 218, no. September, 2025, doi: 10.1016/j.microc.2025.115401.

[12] R. Sreemathy, M. P. Turuk, S. Chaudhary, K. Lavate, A. Ushire, and S. Khurana, "International Journal of Cognitive Computing in Engineering Continuous word level sign language recognition using an expert system based on machine learning," *Int. J. Cogn. Comput. Eng.*, vol. 4, no. April, pp. 170–178, 2023, doi: 10.1016/j.ijcce.2023.04.002.

[13] R. Raj, R. Sreemathy, M. Turuk, J. Jagdale, and M. Anish, "Performance Comparison of Different Versions of YOLO for Indian Sign Language Captioning in Real Time of Multiple Signers," *Procedia Comput. Sci.*, vol. 259, pp. 991–1000, 2025, doi: 10.1016/j.procs.2025.04.053.

[14] M. Á. Remiro, M. Gil-Martín, and R. San-Segundo, "Improving Hand Pose Recognition Using Localization and Zoom Normalizations over MediaPipe Landmarks †," *Eng. Proc.*, vol. 58, no. 1, 2023, doi: 10.3390/ecsa-10-16215.

[15] B. Lagomarsino, A. Massone, F. Odone, M. Casadio, and M. Moro, "Video-based markerless assessment of bilateral upper limb motor activity following cervical spinal cord injury," *Comput. Biol. Med.*, vol. 196, no. PC, p. 110908, 2025, doi: 10.1016/j.compbiomed.2025.110908.

[16] T. Ahmed, J. Tulsi, M. Alam, K. Kadir, K. Mansoor, and M. S. Mazliham, "Analysis and visualization of fraud detection patterns through data mining and classification using MLP and hybrid deep learning model," *Egypt. Informatics J.*, vol. 32, no. May, p. 100829, 2025, doi: 10.1016/j.eij.2025.100829.