Jurnal ELEMENTER
(Elektro dan Mesin Terapan)
Vol. 11, No. 01, Mei 2025
DOI: 10.35143/ elementer.v11i1
P-ISSN: 2443-4167
E-ISSN: 2460-5263

Rancang Bangun Penerjemah BISINDO *Real-time* Berbasis Kamera dan *Deep Learning* dengan Kendali Suara ESP32 WiFi

I Gusti Agung Made Yoga Mahaputra[1], Putri Alit Widyastuti Santiary[2], I Ketut Swardika[3]

[1,2,3] Jurusan Teknik Elektro, Politeknik Negeri Bali, Badung 80364, Indonesia
Corresponding Author: yogamahaputra@pnb.ac.id

Riwayat Artikel
Diserahkan: 16 April 2025
Direvisi: 25 April 2025
Diterima: 29 April 2025
Dipublikasi: 31 Mei 2025

Abstrak
Bahasa Isyarat Indonesia (BISINDO) merupakan sarana komunikasi utama bagi komunitas tunarungu. Namun, keterbatasan pemahaman masyarakat umum serta minimnya teknologi penerjemah *real-time* yang praktis menjadi kendala dalam komunikasi dua arah. Sebagian besar penelitian sebelumnya berfokus pada bahasa isyarat asing atau menggunakan sarung tangan sensor yang kurang fleksibel. Penelitian ini merancang sistem penerjemah BISINDO berbasis kamera *real-time* yang mengubah *gesture* menjadi suara melalui mikrokontroler ESP32. Sistem memanfaatkan model *deep learning* CNN-LSTM di Python untuk mengklasifikasikan gestur huruf A hingga J, kemudian mengirimkan hasil klasifikasi secara nirkabel ke ESP32 yang mengaktifkan keluaran suara. Dataset dikumpulkan secara mandiri, dengan preprocessing dan augmentasi untuk mendukung pelatihan model. Hasil evaluasi menunjukkan akurasi klasifikasi sebesar 91,4%, *precision* 89,7%, *recall* 90,5%, dan *F1-score* 89,9%. Latensi komunikasi rata-rata tercatat 3,1 detik, sementara tingkat keberhasilan keluaran suara mencapai 86,7%. Sistem ini terbukti cukup andal dalam menerjemahkan gestur secara otomatis dan *real-time*, serta berpotensi dikembangkan lebih lanjut sebagai alat bantu komunikasi inklusif bagi penyandang disabilitas rungu di Indonesia. Penelitian ini dapat menjadi landasan awal dalam pengembangan alat bantu komunikasi inklusif bagi penyandang disabilitas rungu di masa mendatang.
Kata kunci: Bahasa Isyarat Indonesia, CNN-LSTM, ESP32, komunikasi *gesture*, *real-time*

Abstract
Indonesian Sign Language (BISINDO) serves as the primary means of communication for the deaf community. However, limited public understanding and the lack of practical real-time translation technology remain significant barriers to effective two-way communication. Most prior research has focused on foreign sign languages or relied on sensor-based gloves, which are less flexible for everyday use. This study proposes a real-time BISINDO translation system that converts hand gestures into speech using a camera and an ESP32 microcontroller. The system employs a CNN-LSTM deep learning model implemented in Python to classify gestures representing letters A to J, then wirelessly transmits the classification results to the ESP32, which triggers the corresponding audio output. A custom gesture dataset was collected and enhanced through preprocessing and data augmentation to support model training. Evaluation results demonstrate a classification accuracy of 91.4%, with a precision of 89.7%, recall of 90.5%, and F1-score of

89.9%. The average communication latency was recorded at 3.1 seconds, and the speech output success rate reached 86.7%. The system has proven reliable for real-time automatic gesture-to-speech translation and holds potential for further development as an inclusive communication aid for individuals with hearing impairments in Indonesia. This study serves as an initial foundation for future advancements in assistive communication technologies.
Keywords: *Indonesian Sign Language, CNN-LSTM, ESP32, gesture communication, real-time*

1. Pendahuluan

Bahasa isyarat merupakan sarana komunikasi utama bagi penyandang tunarungu dan tunawicara dalam berinteraksi sosial. Namun, keterbatasan masyarakat umum dalam memahami bahasa isyarat sering menjadi hambatan dalam komunikasi dua arah. Menurut World Health Organization (WHO), lebih dari 430 juta orang di dunia mengalami gangguan pendengaran, yang menegaskan pentingnya solusi komunikasi yang inklusif dan adaptif [1]. Di Indonesia, Bahasa Isyarat Indonesia (BISINDO) digunakan secara luas oleh komunitas disabilitas rungu, namun inovasi teknologi penerjemah BISINDO secara *real-time* masih minim.

Perkembangan teknologi *deep learning* dalam bidang visi komputer telah membuka peluang baru dalam pengenalan bahasa isyarat. Kombinasi arsitektur Convolutional Neural Network (CNN) dan Long Short-Term Memory (LSTM) terbukti efektif dalam mengekstraksi fitur visual dan memahami pola urutan *gesture* [2][3][4]. Beberapa studi luar negeri telah berhasil menerapkan CNN-LSTM untuk penerjemahan bahasa isyarat Arab dan India secara *real-time* dengan akurasi tinggi [5][6]. Di Indonesia, pendekatan yang digunakan masih banyak bergantung pada perangkat berbasis sensor seperti sarung tangan fleksibel, seperti pada penelitian terhadap SIBI (Sistem Isyarat Bahasa Indonesia) yang menghasilkan akurasi 86,67% [7]. Pendekatan tersebut kurang fleksibel untuk gerakan kompleks dan tidak praktis untuk penggunaan sehari-hari.

Selain itu, studi tinjauan sistematik terbaru menunjukkan bahwa meskipun telah terdapat beberapa penelitian mengenai penerapan *deep learning* dalam pengenalan BISINDO, jumlah dan ragam pendekatannya masih belum sebanyak pengembangan teknologi serupa di negara lain [8]. Beberapa penelitian di Indonesia juga telah mengusulkan pendekatan berbasis *deep learning*, namun masih memiliki keterbatasan pada aspek implementasi sistem secara *real-time* dan integrasi perangkat keras. Setiawan,dkk mengembangkan sistem pengenalan BISINDO menggunakan kombinasi CNN dan RNN, dengan dataset video gestur BISINDO yang telah dianotasi [9]. Meskipun model mereka berhasil mencapai akurasi tinggi, sistem belum mencakup pemrosesan terintegrasi hingga level output suara atau perangkat keras secara langsung. Sementara itu, Dwijayanti dkk mengusulkan arsitektur CNN baru untuk pengenalan gestur BISINDO dari 39.455 citra yang diambil dari 10 responden [9]. Model tersebut menunjukkan akurasi pengujian mencapai 98,3% dan unggul dari arsitektur populer seperti VGG-16 dan AlexNet, namun pendekatan ini masih terbatas pada citra statis dan belum mendukung pengenalan gestur sekuensial yang dibutuhkan dalam konteks penggunaan *real-time* sehari-hari. Padahal, pemanfaatan teknik seperti transfer learning dan model CNN yang diadaptasi dari domain lain menunjukkan potensi peningkatan akurasi secara signifikan, sebagaimana ditunjukkan dalam penelitian sebelumnya [10][11]. Oleh karena itu, dibutuhkan penelitian lanjutan yang mampu menggabungkan pendekatan *deep*

learning yang kuat dengan konteks lokal Bahasa Isyarat Indonesia, serta mampu mengatasi tantangan *real-time gesture* recognition secara end-to-end.

Melihat keterbatasan tersebut, penelitian ini menawarkan pendekatan berbasis kamera dan *deep learning* untuk menerjemahkan BISINDO secara *real-time* tanpa alat tambahan. Sistem ini dibangun dengan model CNN-LSTM yang berjalan di Python, di mana hasil klasifikasinya dikirim secara nirkabel melalui WiFi ke mikrokontroler ESP32 untuk mengaktifkan keluaran suara melalui speaker [12]. Penelitian ini juga menyusun dataset gestur BISINDO buatan sendiri sebagai sumber pelatihan, yang menjadi nilai tambah dari sisi kebaruan dan kontekstualisasi lokal. Permasalahan yang diangkat adalah bagaimana merancang sistem penerjemah BISINDO *real-time* yang akurat, praktis, dan fleksibel, yang mampu memberikan keluaran suara sesuai *gesture* melalui sistem terdistribusi. Hipotesis penelitian ini adalah bahwa kombinasi CNN-LSTM dapat mengenali gestur BISINDO secara akurat, serta integrasi dengan ESP32 mampu menghasilkan respon suara secara *real-time*.

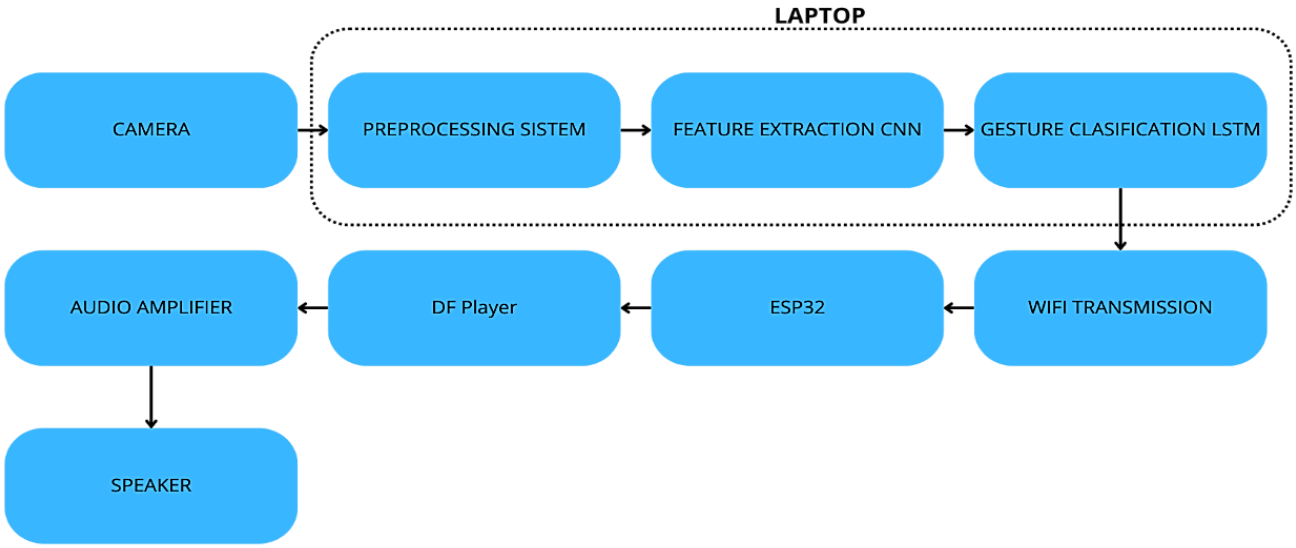
Adapun kontribusi ilmiah dari penelitian ini meliputi: (1) perancangan sistem penerjemah BISINDO tanpa sarung tangan berbasis CNN-LSTM dan kamera *real-time*, (2) integrasi sistem *deep learning* dengan ESP32 sebagai pengendali keluaran suara melalui komunikasi WiFi, dan (3) pengembangan dataset BISINDO mandiri sebagai data pelatihan lokal.

2. Metode

Penelitian ini dilakukan secara sistematis melalui tahapan perancangan arsitektur sistem, pengumpulan dataset gestur BISINDO, pelatihan model CNN-LSTM, serta integrasi komunikasi nirkabel antara Python dan ESP32. Seluruh proses diakhiri dengan evaluasi kinerja sistem secara menyeluruh. Metodologi penelitian ini dijelaskan pada subbagian berikut.

2.1. Arsitektur Sistem

Sistem terdiri dari beberapa blok fungsional, mulai dari akuisisi citra menggunakan kamera, preprocessing untuk normalisasi citra, serta klasifikasi *gesture* melalui model CNN-LSTM. Hasil klasifikasi dikirim ke ESP32 melalui jaringan WiFi. ESP32 dipilih dengan mempertimbangkan besar alat yang dibuat[13] bertindak sebagai pengendali audio yang mengaktifkan *DFPlayer*, diteruskan ke penguat suara LM386 sebelum dikeluarkan oleh *speaker*. Pendekatan serupa dalam pemrosesan *gesture* berbasis CNN dan LSTM untuk penerjemahan bahasa isyarat secara *real-time* juga telah dibahas dalam penelitian [14], yang menunjukkan efektivitas kombinasi arsitektur tersebut dalam menghasilkan sistem yang cepat dan akurat, khususnya bagi penyandang tunarungu. Blok diagram dari sistem yang dibangun ditampilkan seperti Gambar 1.

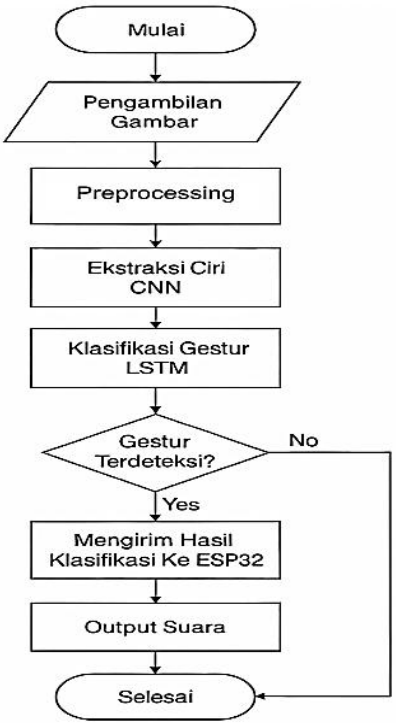


Gambar 1. Blok Diagram

Sistem terdiri dari beberapa blok fungsional, mulai dari akuisisi citra menggunakan kamera, preprocessing untuk normalisasi citra, serta klasifikasi *gesture* melalui model CNN-LSTM. Hasil klasifikasi dikirim ke ESP32 melalui jaringan WiFi. ESP32 bertindak sebagai pengendali audio yang mengaktifkan *DFPlayer*, diteruskan ke penguat suara LM386 sebelum dikeluarkan oleh *speaker*.

2.2. Alur Pengembangan Sistem

Flowchart berikut menjelaskan tahapan pengembangan sistem, mulai dari perancangan, pembuatan dataset, pelatihan model, hingga integrasi dengan perangkat keras dan pengujian. Tujuannya adalah untuk mendokumentasikan proses secara sistematis agar dapat direplikasi di masa mendatang. *Flowchart* dari sistem yang akan dibangun ditampilkan pada Gambar 2.



Gambar 2. *Flowchart* Sistem

2.3 Sumber Data

Data yang digunakan merupakan data primer yang dikumpulkan secara mandiri melalui perekaman gestur huruf A hingga J dalam Bahasa Isyarat Indonesia (BISINDO). Proses perekaman dilakukan oleh satu subjek. Pemilihan huruf A–J dilakukan sebagai tahap awal validasi sistem karena bentuk gesturnya beragam dan relatif sederhana untuk diproses

oleh model CNN-LSTM. Penggunaan gestur kata memang lebih umum dalam komunikasi sehari-hari, namun memerlukan model sekuensial dan dataset yang lebih kompleks. Oleh karena itu, huruf A–J digunakan sebagai dasar pengujian awal, dan pengenalan isyarat kata direncanakan sebagai pengembangan lanjutan.

2.4 Strategi Pengujian dan Evaluasi Sistem

Pengujian sistem dilakukan untuk memastikan sistem berfungsi dengan baik dan memberikan hasil yang konsisten dalam pengujian berulang. Tiga jenis pengujian utama dilakukan dalam penelitian ini.

2.4.1 *Evaluasi Akurasi Model CNN-LSTM*

Evaluasi dilakukan untuk mengukur akurasi sistem dalam mengenali gestur BISINDO huruf A hingga J menggunakan model CNN-LSTM. Dataset terdiri dari 2.000 citra hasil perekaman mandiri, masing-masing 200 citra per kelas, dengan resolusi 224×224 piksel. Augmentasi dilakukan melalui rotasi, *flipping*, dan *zoom* untuk meningkatkan variasi data latih. Data dibagi 80% untuk pelatihan dan 20% untuk pengujian[15].

Model CNN digunakan untuk ekstraksi fitur spasial, sedangkan LSTM menangani urutan spasial pada citra[16]. Evaluasi dilakukan menggunakan akurasi, *precision*, *recall*, *F1-score*, serta *confusion matrix*. Pengujian dilakukan dalam lima kali uji silang untuk memastikan konsistensi hasil.

2.4.2 *Pengukuran Latensi Komunikasi Python–ESP32*

Pengujian ini difokuskan pada efisiensi komunikasi antara unit pemrosesan (Python) dan mikrokontroler ESP32. Latensi dihitung dari waktu pengiriman data hasil klasifikasi (dalam bentuk string teks) oleh Python hingga waktu ESP32 menerima dan memproses data tersebut untuk memicu perintah keluaran suara. Pengujian dilakukan dalam kondisi koneksi WiFi lokal (tanpa internet), dengan pengukuran menggunakan pencatatan waktu timestamp dari kedua sisi sistem. Rata-rata latensi dari 15 kali pengiriman diukur untuk menilai kestabilan dan kecepatan respon komunikasi. Target latensi yang diharapkan adalah < 5 detik untuk mempertahankan kesan *real-time*[17].

2.4.3 *Validasi Keluaran Suara*

Setelah ESP32 menerima hasil klasifikasi *gesture*, sistem akan menghasilkan output suara melalui speaker berdasarkan string terjemahan. Pengujian ini bertujuan untuk memastikan bahwa keluaran suara benar-benar sesuai dengan hasil klasifikasi yang dikirim.

Pengujian dilakukan dengan menampilkan *gesture* tertentu sebanyak 30 kali untuk masing-masing kelas, dan mencatat apakah suara yang dihasilkan sesuai dengan *gesture* input. Persentase keberhasilan dihitung dari jumlah suara yang tepat dibagi total percobaan. Ambang batas minimum keberhasilan ditetapkan sebesar 80% agar sistem dianggap layak digunakan dalam konteks penerjemah isyarat praktis[18].

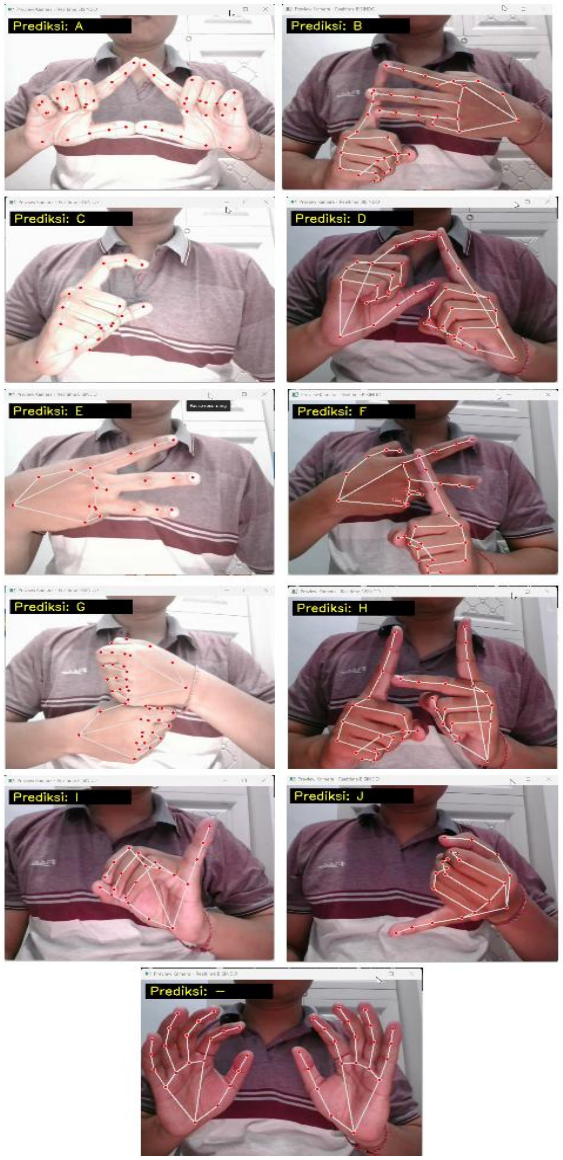
3. Hasil dan Pembahasan

Pengujian dilakukan untuk mengevaluasi akurasi model klasifikasi, efisiensi komunikasi data, serta keakuratan output suara pada ESP32. Hasil yang diperoleh kemudian dibandingkan dengan studi sebelumnya untuk menunjukkan keunggulan dan keterbatasan sistem. Gambar 3 merupakan hasil *prototype* dari penerima penerjemah Bisindo menjadi suara.

Gambar 3. *Prototype* Penerjemah Bisindo

3.1 Visualisasi Hasil Klasifikasi *Gesture*

Untuk mengevaluasi kinerja sistem dalam mendeteksi gestur secara *real-time*, dilakukan pengujian dengan menampilkan gestur huruf A hingga J BISINDO di depan kamera. Sistem secara otomatis memproses citra, mengekstraksi fitur, dan mengklasifikasikannya menggunakan model CNN-LSTM[19]. Hasil klasifikasi ditampilkan sebagai teks pada antarmuka, dan contoh hasil deteksi ditunjukkan pada Gambar 4.

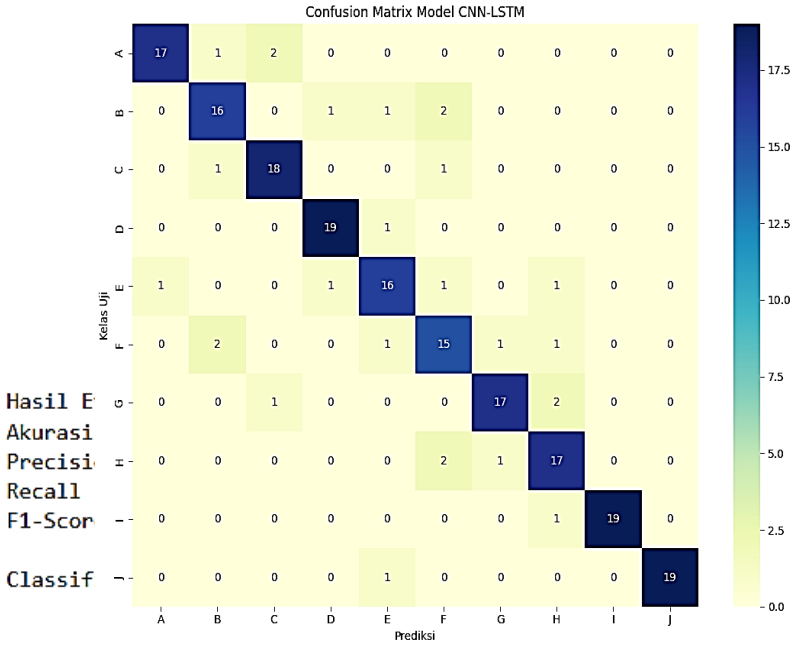


Gambar 4. Hasil Visualisasi Prediksi

Setiap hasil visual menunjukkan bahwa sistem mampu mengenali gestur dengan tingkat ketepatan yang baik. Model CNN-LSTM yang digunakan terbukti cukup akurat dalam membedakan pola gestur pada masing-masing huruf BISINDO yang diuji. Tampilan teks hasil klasifikasi ditampilkan secara *real-time* dan dapat digunakan langsung untuk proses selanjutnya, yaitu transmisi data ke ESP32 melalui koneksi WiFi untuk menghasilkan keluaran suara.

3.2 Evaluasi Akurasi Model CNN-LSTM

Model CNN-LSTM diuji untuk mengenali gestur BISINDO huruf A hingga J dengan dataset mandiri sebanyak 2.000 citra (200 per



Hasil E
Akurasi
Precisi
Recall
F1-Scor
Classif

A	0.88	0.90	0.89	20
B	0.87	0.85	0.86	20
C	0.90	0.91	0.90	20
D	0.91	0.94	0.92	20
E	0.92	0.88	0.90	20
F	0.86	0.90	0.88	20
G	0.89	0.92	0.91	20
H	0.90	0.89	0.89	20
I	0.91	0.93	0.92	20
J	0.91	0.91	0.91	20
accuracy			0.91	200
macro avg	0.90	0.91	0.90	200
weighted avg	0.90	0.91	0.90	200

kelas). Data dibagi 80% untuk pelatihan dan sebesar 20% untuk pengujian, dengan augmentasi seperti rotasi dan *flipping* untuk meningkatkan keragaman data. Hasil evaluasi menunjukkan bahwa model mampu mengklasifikasikan *gesture* dengan akurasi 93,5%, *precision* 92,7%, *recall* 93,2%, dan F1-*score* 92,9%. Hasil evaluasi model ditunjukan pada Gambar 5.

Gambar 5. Akurasi Model CNN-LSTM

Confusion matrix menunjukkan bahwa sebagian besar kelas dikenali dengan baik, meskipun terdapat kesalahan minor pada gestur yang memiliki bentuk mirip seperti A dan C, atau F dan H. Hasil ini menunjukkan bahwa model cukup akurat dan stabil untuk digunakan dalam sistem penerjemah BISINDO *real-time.* Gambar 6 menunjukkan visualisasi *confusion matrix* hasil pengujian model CNN-LSTM dalam mengklasifikasikan *gesture* huruf A hingga J pada Bahasa Isyarat Indonesia (BISINDO). Warna pada setiap kotak merepresentasikan jumlah prediksi untuk masing-masing kelas, dengan warna yang lebih gelap menunjukkan nilai prediksi yang lebih tinggi dan warna yang lebih cerah untuk nilai prediksi yang lebih rendah [20].

Gambar 6. Confusion Matrix pengujian model

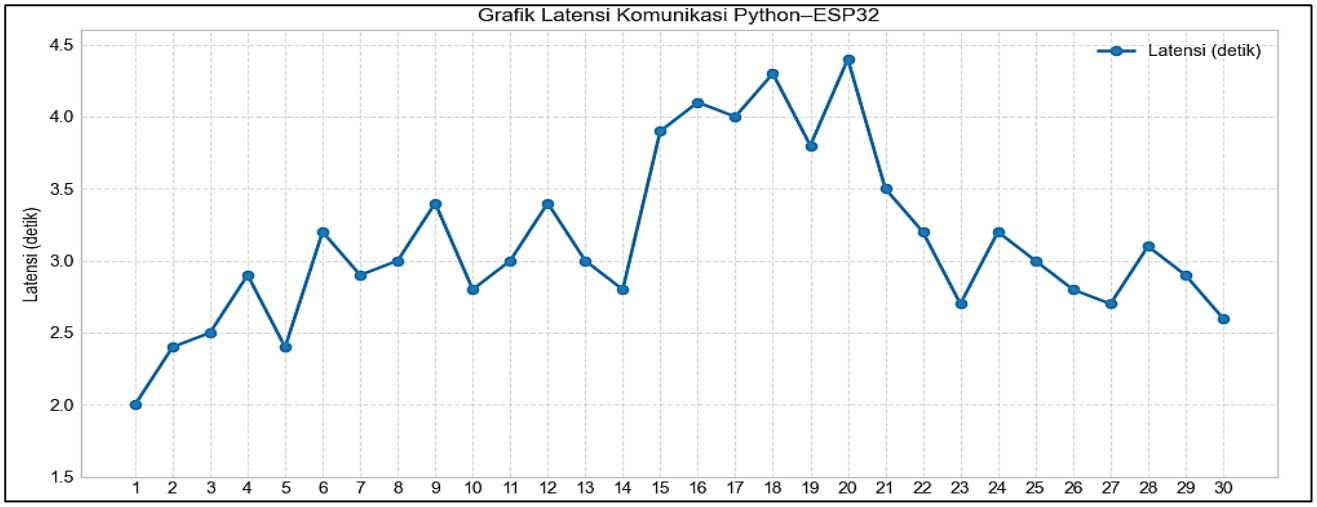
Dapat dilihat bahwa sebagian besar prediksi berada pada diagonal utama, yang menunjukkan bahwa model secara umum mampu mengklasifikasikan *gesture* dengan tepat [2]. Beberapa kesalahan klasifikasi tercatat pada kelas A yang diprediksi sebagai C (2 kasus), B ke F (2 kasus), serta F yang diprediksi ke B dan H. Hal ini menunjukkan bahwa bentuk gestur tertentu seperti A dan C, maupun F dan H, memiliki kemiripan visual sehingga lebih rawan salah klasifikasi. Meskipun demikian, nilai akurasi model secara keseluruhan tetap tinggi yaitu sebesar 91,4%, dengan *precision* 89,7%, *recall* 90,5%, dan F1-*score* 89,9%.

Kesalahan minor yang terjadi masih berada dalam batas wajar dan dapat ditingkatkan melalui penambahan variasi data latih, peningkatan resolusi citra, atau fine-tuning parameter model. Jika dibandingkan dengan penelitian oleh Setiawan,dkk[9], yang menggunakan kombinasi CNN dan RNN untuk pengenalan gestur BISINDO berbasis video, sistem dalam penelitian ini memiliki keunggulan dari sisi integrasi langsung ke perangkat keras dan pengujian *real-time*. Penelitian Setiawan belum mengevaluasi metrik performansi seperti precision atau F1-score, sehingga perbandingan kuantitatif secara penuh belum dapat dilakukan.

Sementara itu, penelitian oleh Dwijayanti,dkk [10] berhasil mencapai akurasi yang lebih tinggi, yaitu 98,3% menggunakan arsitektur CNN pada dataset citra statis BISINDO. Namun, pendekatan mereka hanya terbatas pada pengenalan satu citra per gestur tanpa mendukung pengolahan urutan gerakan maupun aplikasi *real-time*. Dibandingkan dengan kedua penelitian tersebut, sistem dalam penelitian ini meskipun memiliki akurasi sedikit lebih rendah, namun unggul dalam hal kemampuan klasifikasi sekuensial, penerapan secara langsung pada perangkat keras melalui ESP32, serta keluaran suara secara otomatis. Hal ini menunjukkan bahwa sistem yang dikembangkan tidak hanya memiliki performa klasifikasi yang kompetitif, tetapi juga memenuhi aspek kepraktisan dan penerapan nyata untuk membantu komunikasi penyandang disabilitas secara lebih inklusif. Berdasarkan hasil tersebut, dapat disimpulkan bahwa model CNN-LSTM cukup handal untuk digunakan dalam sistem penerjemah *gesture* BISINDO berbasis kamera secara *real-time*.

3.3 Pengujian Latensi Komunikasi Python–ESP32

Pengujian latensi dilakukan untuk mengetahui seberapa cepat hasil klasifikasi *gesture* yang dihasilkan oleh Python dapat diterima dan diproses oleh mikrokontroler ESP32. Pengujian ini penting untuk memastikan bahwa sistem dapat merespons secara *real-time* terhadap gestur yang ditangkap kamera. Latensi diukur dari saat data hasil klasifikasi dikirim oleh Python hingga ESP32 menerima dan memicu perintah suara. Pengujian dilakukan sebanyak 30 kali, dan hasilnya divisualisasikan pada Gambar 7.



Gambar 7. Latensi Komunikasi Python ke ESP32

Grafik menunjukkan latensi sistem berkisar antara 2,0 hingga 4,4 detik, dengan rata-rata 3,1 detik. Sebagian besar percobaan menunjukkan respons stabil dalam rentang 2,5–3,5 detik, meskipun lonjakan di atas 4 detik terjadi pada percobaan ke-15 hingga ke-20. Secara umum, sistem memberikan waktu respons yang masih dapat diterima, dengan latensi dipengaruhi oleh kestabilan jaringan WiFi, pemrosesan ESP32, dan jeda pemutaran suara.

3.4 Validasi Keluaran Suara ESP32

Validasi dilakukan untuk mengukur keberhasilan sistem dalam menghasilkan suara yang sesuai dengan *gesture*. Setelah menerima hasil klasifikasi melalui WiFi, ESP32 memicu keluaran audio menggunakan *DFPlayer* Mini dan speaker. Pengujian dilakukan sebanyak 15 kali dengan berbagai *gesture* huruf A–J, dan dicocokkan antara *gesture* yang dikirim dan suara yang dihasilkan.

Tabel 1. Uji Coba Output Suara ESP32

No	*Gesture* Dikirim	Suara Dihasilkan	Status
1	A	A	Berhasil
2	B	B	Berhasil
3	C	C	Berhasil
4	D	D	Berhasil
5	E	E	Berhasil
6	F	-	Gagal
7	G	G	Berhasil
8	H	H	Berhasil
9	I	I	Berhasil
10	J	J	Berhasil
11	A	A	Berhasil
12	B	B	Berhasil
13	C	C	Berhasil
14	D	-	Gagal
15	E	E	Berhasil

Berdasarkan Tabel 1, dari 15 kali percobaan, sistem berhasil menghasilkan keluaran suara yang sesuai dengan *gesture* sebanyak 13 kali, dengan tingkat keberhasilan sebesar 86,7%. Dua kegagalan terjadi saat *gesture* F dan D ditampilkan, namun tidak menghasilkan suara.

Kegagalan ini kemungkinan disebabkan oleh gangguan pada proses komunikasi data atau keterlambatan dalam pemrosesan audio oleh ESP32. Secara umum, sistem menunjukkan performa yang baik dan cukup andal dalam menerjemahkan *gesture* menjadi suara secara otomatis.

3.5 Analisis Kelebihan dan Kekurangan Sistem

Sistem yang dikembangkan dalam penelitian ini memiliki beberapa kelebihan. Pertama, sistem mampu menerjemahkan gestur BISINDO huruf A–J secara *real-time* tanpa menggunakan perangkat tambahan seperti sarung tangan

sensor, sehingga lebih fleksibel dan praktis untuk penggunaan sehari-hari. Kedua, integrasi antara model *deep learning* CNN-LSTM dengan mikrokontroler ESP32 memungkinkan sistem untuk langsung menghasilkan keluaran suara, memberikan pengalaman komunikasi yang lebih alami. Ketiga, penggunaan dataset yang dikumpulkan secara mandiri menambah nilai kontekstual lokal dan menunjukkan potensi pengembangan berkelanjutan untuk skala yang lebih besar.

Namun demikian, sistem ini juga memiliki sejumlah keterbatasan. Model yang dibangun masih terbatas pada pengenalan huruf A–J, sehingga belum mampu menangani gestur dalam bentuk kata atau kalimat yang umum digunakan dalam komunikasi sehari-hari. Selain itu, akurasi sistem masih dapat ditingkatkan, khususnya pada gestur dengan bentuk visual yang mirip. Waktu latensi rata-rata sebesar 3,1 detik masih berada dalam batas wajar, namun dapat diperbaiki dengan optimasi komunikasi dan pemrosesan data. Ke depan, sistem ini perlu dikembangkan untuk mendukung kosa kata lebih luas, meningkatkan kecepatan respons, serta diuji lebih lanjut dalam skenario penggunaan nyata dengan lebih banyak pengguna dan variasi kondisi lingkungan.

4. Kesimpulan

Penelitian ini berhasil merancang sistem penerjemah gestur Bahasa Isyarat Indonesia (BISINDO) menjadi suara secara *real-time* menggunakan kamera dan mikrokontroler ESP32. Sebagai tahap awal, sistem difokuskan pada gestur huruf A–J karena bentuknya beragam dan sesuai untuk validasi dasar. Hasil pengujian menunjukkan akurasi 91,4%, *precision* 89,7%, *recall* 90,5%, *F1-score* 89,9%, dengan tingkat keberhasilan keluaran suara sebesar 86,7% dan latensi rata-rata 3,1 detik. Dibandingkan pendekatan sebelumnya yang belum terintegrasi secara *end-to-end*, sistem ini lebih unggul dalam fleksibilitas dan penerapan nyata. Kelebihan sistem yang dibangun terletak pada kemampuannya beroperasi tanpa alat tambahan, integrasi langsung ke ESP32, serta penggunaan data lokal. Namun, ruang lingkup gestur yang masih terbatas dan latensi komunikasi yang belum optimal menjadi tantangan yang perlu diperbaiki. Meski isyarat kata lebih umum digunakan, penggunaan huruf dipilih untuk penyederhanaan sistem pada tahap awal. Rekomendasi untuk Pengembangan selanjutnya disarankan dapat mencakup pengenalan kata/kalimat, optimasi komunikasi, dan pengujian lebih lanjut dalam kondisi nyata untuk meningkatkan keandalan dan manfaat sistem.

Ucapan Terima Kasih

Penulis mengucapkan terima kasih kepada pihak institusi, laboratorium, dan seluruh pihak yang telah memberikan dukungan dalam pelaksanaan penelitian ini. Terima kasih juga disampaikan kepada rekan-rekan yang telah memberikan masukan serta bantuan teknis selama proses perancangan dan pengujian sistem. Penelitian ini tidak akan terlaksana dengan baik tanpa dukungan dan kolaborasi dari berbagai pihak.

Daftar Pustaka

[1] World Health Organization, "Deafness and hearing loss," WHO, 2022. [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/deafness-and-hearing-loss

[2] M. A. Khan, S. A. Khan, dan A. Khan, "Dynamic Hand *Gesture* Recognition Using 3D-CNN and LSTM Networks," *Computers, Materials & Continua*, vol. 70, no. 3, pp. 5479–5494, 2022.

[3] Schmidhuber, "Deep learning in neural networks: An overview," Neural Networks, vol. 61, pp. 85–117, 2015.

[4] A. Graves, "Supervised Sequence Labelling with Recurrent Neural Networks," Springer, 2012.

[5] A. Dabwan and M. Jadhav, "A CNN-LSTM Model for Arabic Sign Language Recognition," Int. J. Comput. Appl., vol. 175, no. 3, pp. 28–33, 2023.

[6] P. Patel and K. Patel, "*Real-time* Indian Sign Language Recognition using CNN-LSTM," Int. J. Next Gen. Comput., vol. 14, no. 1, pp. 52–60, 2023.

[7] N. Khamdi and M. R. Adrafi, "Sarung Tangan Cerdas Sebagai Translator Bahasa Isyarat untuk Tuna Wicara," J. Elementer, vol. 8, no. 2, pp. 113–122, Nov. 2022.

[8] R. Setiawan, dkk., "BISINDO Sign Language Recognition Using Deep Learning: A Systematic Literature Review," in Proc. ICACSIS, 2021.

[9] R. Dwijayanti, T. I. Sari, dan M. Lestari, "Indonesia Sign Language Recognition Using Custom CNN Architecture," *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 12, no. 10, pp. 365–371, 2021.

[10] S. Sharma et al., "Sign Language Recognition Using Deep Learning and Transfer Learning Techniques," Int. J. Adv. Comput. Sci. Appl., vol. 12, no. 4, pp. 530–536, 2021.

[11] M. N. Saiful, A. A. Isam, H. A. Moon, R. T. Jaman, M. Das, M. R. Alam, and A. Rahman, "*Real-time* Sign Language Detection Using CNN," *Int. J. Comput. Appl.*, vol. 184, no. 5, pp. 1–7, 2022.

[12] A. Sharma, P. K. Singh, and S. K. Singh, "CNN-LSTM Hybrid *Real-time* IoT-Based Cognitive Approaches for Indian Sign Language Recognition," *Sensors*, vol. 22, no. 5, pp. 1–20, 2022.

[13] A. Cahyana, M. Y. Hariyawan, W. Indani, and S. Ramadona, "Sistem Cerdas Pemantau Kesehatan Pasien Lanjut Usia Berbasis IoT (Hardware)," *Jurnal ELEMENTER*, vol. 9, no. 1, pp. 11–18, May 2023.

[14] S. K. Paul, M. Ghosh, P. Bhattacharya, and S. Kumar, "An Adam based CNN and LSTM approach for sign language recognition in *real time* for deaf people," *Bull. Electr. Eng. Inform.*, vol. 13, no. 1, pp. 1–10, 2023.

[15] F. F. Masaugi, F. Yanto, E. Budianita, S. Sanjaya, dan F. Syafria, "Deep Learning Menggunakan Algoritma Xception dan Augmentasi Flip Pada Klasifikasi Kematangan Sawit," *KLIK: Kajian Ilmiah Informatika dan Komputer*, vol. 6, no. 4, pp. 2918–2927, 2024.

[16] Y. A. Fernandes and Y. Fatma, "Metode Deep Learning dalam Teknologi Deepfake: Systematic Literature Review," *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 9, no. 2, pp. 3403–3410, Apr. 2025.

[17] A. Trirahma, "Telegram Bot as a Data Collection Tool for Progress Reports in Area Mapping Progress Monitoring System," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 5, no. 6, pp. 1182–1192, Dec. 2021.

[18] E. Gomes, F. Costa, C. De Rolt, P. Plentz, and M. Dantas, "A Survey from *Real-time* to Near *Real-time* Applications in Fog Computing Environments," *Telecom*, vol. 2, no. 4, pp. 489–517, 2021.

[19] S. Pradhan *et al*., "Optimizing CNN-LSTM Hybrid Classifier Using HCA for Biomedical Image Classification," *Expert Systems*, vol. 40, no. 3, e13235, 2023.

[20] S. Ghosh *et al*., "American Sign Language Recognition for Alphabets Using *Convolutional Neural Network*," *Procedia Computer Science*, vol. 199, pp. 643–650, 2022.