Lightweight Hybrid CNN-LSTM for Sign Language Recognition in Low-Resource Environments

1st Muhammad Anang Ayman Ramadhana
Computer Science Department, School of Computer Science
Bina Nusantara University
Jakarta, Indonesia
muhammad.ramadhana003@binus.ac.id

2nd Benediktus Darmawan
Computer Science Department, School of Computer Science
Bina Nusantara University
Jakarta, Indonesia
benediktus.darmawan001@binus.ac.id

3rd Nicholaus Hendrik Jeremy
Computer Science Department, School of Computer Science
Bina Nusantara University
Jakarta, Indonesia
nicholaus.jeremy@binus.ac.id

4th Rhio Sutoyo
Computer Science Department, School of Computer Science
Bina Nusantara University
Jakarta, Indonesia
rsutoyo@binus.edu

***Abstract*—Communication is vital, yet individuals with hearing and speech impairments, including approximately 70 million deaf sign language users globally, face significant barriers. This research addresses the urgent need for effective Sign Language Recognition (SLR) systems, particularly in low-resource environments like Indonesia where limited hardware and connectivity hinder the deployment of advanced solutions. While deep learning models like Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks have improved SLR, existing systems often lack optimization for low-end devices. This study aimed to develop a lightweight, accurate, and efficient SLR system for Indonesian Sign Language (BISINDO) capable of local offline operation. A hybrid CNN-LSTM architecture, utilizing MobileNetV2 and LSTM with attention, was trained on low-resolution BISINDO image data and optimized using L2 regularization and post-training quantization. The model achieved 96.06 percent training and 95.30 percent validation accuracy, with a file size under 25 megabytes and 40 to 50 millisecond inference time, demonstrating strong generalization. Despite some misclassifications of visually similar signs due to dataset limitations, this work offers a viable solution for democratizing communication technology in resource-constrained communities, with future efforts focusing on dataset expansion and continuous recognition.**

Keywords— sign-language recognition, low-resource environments, BISINDO, deep learning

I. Introduction

Communication serves as a fundamental aspect of human interaction, facilitating the exchange of information, emotions, and ideas. However, individuals with hearing and speech impairments face significant challenges in communication, particularly in low-resource communities where access to assistive technologies is limited. According to estimates, there are approximately 70 million deaf individuals worldwide who rely on sign language as their primary means of communication [1]. Despite its vital role, sign language remains inaccessible to much of the hearing population, creating a linguistic divide that isolates users in both social and practical contexts.

This divide underscores the urgent need for sign language recognition systems (SLR), which could automate translation and foster inclusion. While SLR technology holds tremendous potential to bridge communication gaps, its widespread adoption faces systemic challenges. Most advanced solutions are confined to high-resource environments equipped with state-of-the-art hardware, high-resolution cameras, and stable internet connectivity. Such conditions are rarely met in regions like Indonesia, home to nearly 5 million deaf individuals [2], where access to premium devices is a luxury most cannot afford. In Indonesia, many people rely on budget devices with underperforming processors, subpar camera quality, and limited storage capacity, while large portions of the population face inconsistent access to 4G connectivity. These limitations severely hinder the practicality of deploying sophisticated, cloud-based SLR models. To dismantle these barriers, the path forward is clear: the creation of a lightweight, accurate, and efficient sign language recognition system—one that operates locally on affordable devices, democratizing access to communication tools that transcend infrastructural and economic divides.

Deep learning, particularly using Convolutional Neural Networks (CNNs) for spatial features and Long Short-Term Memory (LSTM) networks for temporal analysis, has significantly advanced sign language recognition. However, existing models present a trade-off between accuracy and efficiency. For example, attention-based CNN-LSTM frameworks, while accurate, are computationally demanding for real-time use [3]. Similarly, cascaded CNN-LSTM models achieve competitive accuracy but are not optimized for low-resource environments [4]. Even models designed for real-time performance often face degradation on edge devices [1], and while efficient architectures like MobileNetV2-LSTMs have been successful in other domains, their application to sign language is not yet fully explored, optimized, or validated for real-time gesture recognition [5].

Despite these advancements, achieving an optimal balance between accuracy and computational efficiency remains a challenge, particularly in low-resource environments. High-performing models often rely on computationally expensive architectures that are unsuitable for deployment on low-end devices [6]. The absence of a robust, lightweight hybrid CNN-LSTM model specifically optimized for sign language recognition in such settings highlights a critical research gap. Furthermore, existing models lack effective techniques for model compression, which is essential for real-world deployment in resource-limited contexts [7].

This paper aims to address an important issue which are: the lack of SLR systems that can balance between accuracy, efficiency, and temporal modelling for low-resource environments. This research propose to utilize a hybrid network that uses the benefits of CNN's such as MobileNetV2 and LSTM with an attention mechanism to recognize isolated sign language words.

The system is quantized for 4GB RAM devices and trained on low-resolution (720p) Indonesian Sign Language (BISINDO) videos. The objectives are twofold:

1) Design a hybrid CNN-LSTM architecture that captures both spatial and temporal features of sign language using low-resolution inputs.
2) Optimize the model for offline deployment on low-end devices via quantization and regularization.

This study follows an experimental approach where the model is trained and evaluated on a dataset consisting of various sign language gesture. Various performance metrics such as accuracy, loss and inference time are analyzed to assess the effectiveness of this implementation of a hybrid MobileNetV2-LSTM architecture.

This study also applies model optimization techniques such as quantization and pruning. These techniques aims to enhance the model's compatibility for use on low-resource devices [8].

The remainder of this paper is structured as such: The literature review section discusses the technologies available to be used on sign language recognition.In addition, it also examines the existing research on sign language recognition, showing the contributions and limitations of past studies. The methodology sections outlines on the research and experimental framework. This includes dataset selection, model architecture, and optimization techniques. The results and discussion section highlights the improvements in the accuracy and computational efficiency of the model. The conclusion section summarizes key findings and proposes direction for future research. This includes enhancing the model performance and expanding dataset diversity.

II. Literature Review

To support the design and implementation of the proposed sign language recognition system, this literature review explores key developments in the field. It begins by defining sign language and outlining the structure and purpose of sign language recognition systems. Next, it examines the deep learning models commonly used in SLR, followed by a review of available datasets and their limitations. Finally, this review highlights recent state-of-the-art models to establish a foundation for comparison and identify opportunities for further advancement.

A. Sign Language and Sign Language Recognition Systems

Sign languages are natural languages that use manual (hand shapes and movements) and non-manual (facial expressions, body posture) cues to convey meaning. Sign language recognition (SLR) systems automatically interpret these visual signals into text or speech, aiding communication for the deaf and hard of hearing [9]. Typically, SLR involves four stages: data acquisition, feature extraction, temporal modeling, and classification, enabling real-time recognition without special hardware [10]. These systems enhance accessibility across education, healthcare, and public services.

Despite progress, SLR research still faces major challenges in low-resource languages. Limited annotated datasets constrain model performance [11], and achieving signer-independence is difficult due to variability in signing styles and physical traits. Kim et al. [12] also highlight deployment issues on edge devices, where limited processing power and frame rates demand sampling strategies and model compression. These challenges emphasize the need for lightweight architectures and transfer learning to improve SLR accessibility in resource-constrained settings.

B. Deep Learning Models

In the past decade, the development of deep learning models has produced several iterations and improvements. It results in various models, such as Convolutional Neural Network and Recurrent Neural Network.

1) Convolutional Neural Network (CNN): CNN is a deep learning architecture effective for image recognition, combining input and kernel matrices to extract visual features [13]. It uses stacked convolutional layers with non-linear activations and pooling to detect patterns and reduce spatial resolution. Fully connected layers then map features to class scores through backpropagation. In this research, MobileNetV2 is used as the spatial backbone to encode hand-shaped features into compact vectors.

2) Recurrent Neural Network (RNN): Recurrent Neural Networks (RNNs) are designed for sequential data, but the Long Short-Term Memory (LSTM) model improves upon them by overcoming issues with long-term dependencies [14]. LSTMs use a dedicated memory cell and three gates (input, forget, and output) to regulate information flow, allowing them to be stacked for hierarchical temporal feature extraction. Typically trained with ReLU activation and the Adam optimizer for stable convergence [15], the LSTM in this study serves as the temporal modeling backbone to capture the motion dynamics of hand gestures.

In conclusion, the combination of CNNs and RNNs, particularly MobileNetV2 and LSTM, provides an effective framework for recognizing spatiotemporal data like sign language gestures. CNNs extract spatial features, while LSTMs model

temporal dynamics, enabling accurate interpretation of gesture sequences. This synergy makes the approach well-suited for sign language recognition tasks.

C. Optimization Techniques

Quantization is a technique that converts a model's high-precision parameters into discrete low-bit formats (2-bit integer, 4-bit integer, 8-bit integer) to reduce both storage requirements and arithmetic complexity on hardware accelerators. Quantization techniques can be divided into:

1) Post-Training Quantization (PTQ): Training weights with FP32 and quantizing the results into smaller data types [16].
2) Quantization Aware Training (QAT): Training weights for maximizing their accuracy with the quantized data type [16].

Regularization is a technique to prevent overfitting by adding a penalty term to the loss function during training. L1 regularization (also called Lasso) adds the absolute value of the weights to the loss (can force some weights to become exactly zero). L2 regularization (also called Ridge) adds the squared value of the weights to the loss (discourages large weights but does not eliminate them completely).

In summary, quantization and regularization are key for optimizing deep learning models in resource-limited environments. Quantization reduces memory and computation, while regularization prevents overfitting. Together, they enable efficient, lightweight, and robust models for real-time or low-power deployment.

D. Datasets

While many public datasets for sign language recognition exist, such as ASLLVD for American Sign Language and RWTH-PHOENIX-Weather for German Sign Language, they primarily focus on major languages and continuous, video-based recognition tasks [17], [18]. This creates a resource gap for image-based, alphabet-level recognition in less-resourced languages like BISINDO. To address this, the study utilized several specific BISINDO hand sign datasets, including one by Rhio et al. with 520 images [21]. Additionally, a primary dataset of 260 images was constructed under constrained conditions to simulate a low-resource setting. These datasets, detailed in Table I, provided the necessary data for training and evaluation.

Data augmentation expands training data and simulates real-world variability to improve model robustness, especially when samples are limited. Techniques include geometric transformations like rotation, scaling, translation, shearing, and flipping to teach spatial invariance. Photometric changes involve adjusting brightness, contrast, and applying blur to mimic varied lighting and focus conditions [13].

E. State-of-the-art model

Based on a review of current research, the hybrid CNN-LSTM architecture is a proven and effective framework for sign language recognition. Studies show that combining a lightweight CNN such as MobileNetV2 for spatial feature extraction with an LSTM for temporal analysis successfully captures the complex dynamics of sign gestures [3], [4]. For example, a MobileNetV2-LSTM model with an attention mechanism achieved high accuracy in a benchmark dataset, validating the potential of the architecture [3]. Other research further supports by demonstrating that hybrid models can effectively capture both temporal patterns and long-term dependencies in sign language sequences [22].

Despite these advances, a significant gap remains in optimizing these models for low-resource environments. While studies on BISINDO have demonstrated real-time static alphabet detection using lightweight models like MobileNetV2 and YOLOv5, they often highlight the need to extend capabilities to dynamic gestures and optimize for mobile deployment [21], [23]. The existing state-of-the-art models are often not designed for the computational and storage constraints of low-end devices. This underscores the critical need for a solution that is not only accurate but also optimized for efficient offline performance [3], [22].

III. Methodology

This section outlines the proposed methodology for recognizing BISINDO (Indonesian Sign Language) hand signs using a deep learning approach. The pipeline is designed to handle both low-resource and standard data scenarios and is structured around a CNN-LSTM hybrid model based on MobileNetV2.

The overview workflow of the system is illustrated in Fig. 1, which presents each stage of the process, from data preparation to model development, training, and result analysis.

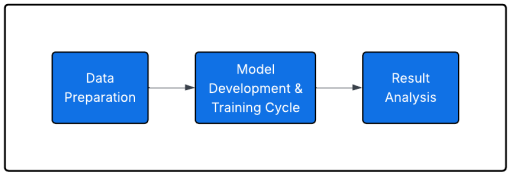


Fig. 1: Overview Diagram of Proposed Methodology

A. Data preparation

The development of an effective sign language recognition model begins with meticulous data preparation. This subsection details this crucial phase, outlining the strategies for data collection from various sources, the methodology employed for accurate image labeling, and the comprehensive suite of preprocessing techniques applied to optimize the dataset for robust model training and subsequent evaluation

1) Data collection: In this study, a custom BISINDO (Indonesian Sign Language) hand sign dataset was constructed. This dataset utilized 10 images from each of the 26 BISINDO alphabet characters, resulting in a primary dataset of 260 images designed to simulate a low-resource setting. Furthermore, a publicly available dataset by Rhio et al. [21], which includes 20 images per character (520 images in total), was

TABLE I: PUBLICLY AVAILABLE SIGN LANGUAGE RECOGNITION DATASET

Dataset	Language	Modality	Task Type	Total Samples	Usage in This Study
ASLLVD [17]	ASL	RGB (Multi-angle video)	Isolated	∼9,800 tokens	No
RWTH-PHOENIX-Weather [18]	German Sign Language	RGB video	Continuous	∼9,000+	No
AUTSL [19]	Turkish SL	RGB, Depth, Skeleton	Isolated	38,336	No
CSL-Daily [19]	Chinese SL	RGB video	Isolated/Continuous	1000+ classes	No
Noer [20]	BISINDO	RGB image	Isolated (Alphabet)	312	No
Rhio et al. [21]	BISINDO	RGB image	Isolated (Alphabet)	520	Yes
Our Primary Dataset	BISINDO	RGB image	Isolated (Alphabet)	260	Yes

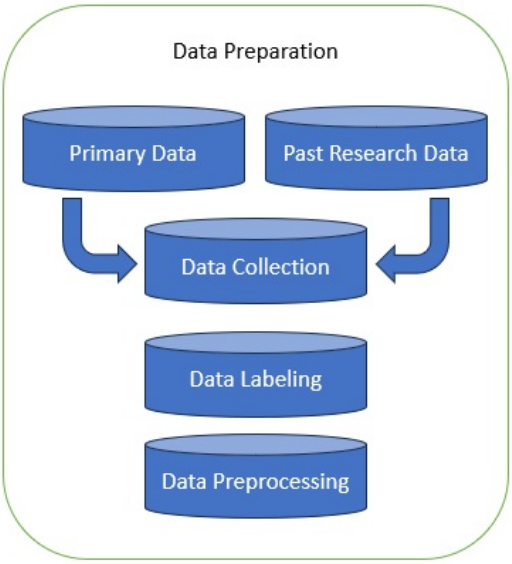


Fig. 2: Data preparation details.

also used. These two datasets were combined to simulate both low-resource and standard-resource scenarios.

Fig. 3: Primary data collected for ”A”.

2) Data labelling: Data labeling was carried out using Python's labelImg module, where each image was manually annotated on the basis of the corresponding BISINDO hand sign. This ensured accurate and consistent labeling across the dataset, which is essential to train a reliable supervised learning model.

3) Data preprocessing: All images undergo a series of preprocessing steps to improve model performance. These include resizing the images to a consistent resolution of 224×224 pixels and normalizing the pixel values. Data augmentation techniques such as applying blur and brightness adjustments are applied to enhance generalization and simulate variability in real-world settings. These preprocessing steps help reduce noise, handle class imbalance, and improve the robustness of the model during training. Finally, the dataset was split into training and testing sets with a ratio of 7:3 to ensure proper evaluation of model performance.

B. Model development & Training Cycle

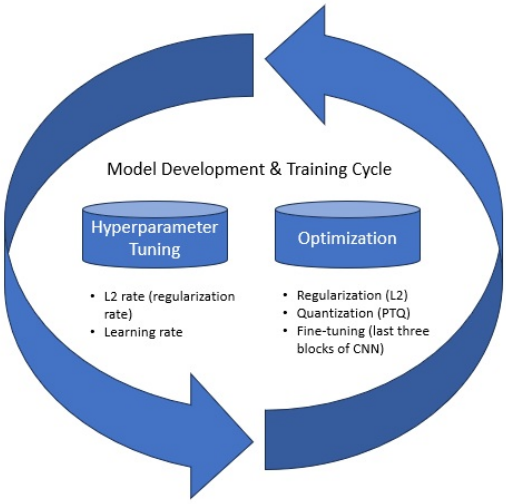


Fig. 4: Model development details.

A hybrid MobileNetV2-LSTM architecture was adopted to create an efficient sign language recognition system for low-resource environments. MobileNetV2 (a lightweight CNN) serves as the spatial backbone to extract features efficiently on budget devices. An LSTM network was then integrated to capture temporal dynamics of sign gestures across frames without high computational cost of 3D convolutions. While the architecture provided a strong foundation, it demanded careful refinement. It was necessary to fine-tune to keep the model lightweight without sacrificing accuracy. To address these challenges, efforts focused on two critical processes:

1) Hyperparameter Tuning: To optimize performance, hyperparameter tuning was conducted. This involved adjusting the learning rate and the regularization strength (l2) of the model. A random search strategy is used to systematically test combinations of these parameters, evaluating model accuracy and loss to determine the optimal configuration.

2) Optimization: Optimization techniques such as regularization and quantization were applied to further reduce the complexity of the model without significantly sacrificing accuracy. L2 regularization penalized large weights, leading to a more stable and generalizable model. Quantization significantly reduces model size by converting 32-bit floating-point weights to 8-bit integers which is implemented using the TensorFlow Model Optimization Toolkit. Additionally, the model was fine-tuned by unfreezing the last three layers of the MobileNetV2 backbone to adapt it specifically for BISINDO hand sign recognition.

C. Result analysis

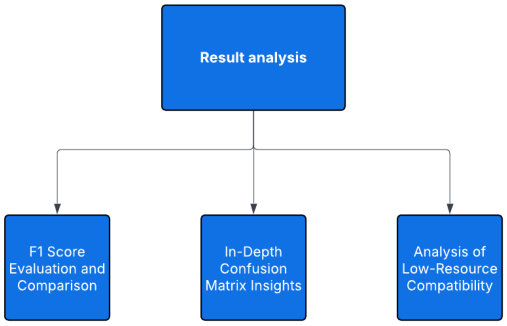


Fig. 5: Result analysis details.

To evaluate the performance of the proposed CNN-LSTM model, experiments will be conducted using various hyperparameter configurations and optimization techniques. The F1 score will be used as the primary evaluation metric, as it effectively balances precision and recall, making it suitable for imbalanced datasets. Additionally, the final model file size will be recorded to assess the efficiency and practicality of deploying the model on low-resource devices. This comparison will highlight the trade-offs between model complexity, performance, and storage efficiency. A confusion matrix analysis will be conducted for each configuration to reveal patterns of misclassification, which will show which hand signs are frequently misclassified. Such insights will be critical for understanding model limitations and guiding future improvements.

IV. Results and Discussion

The model developed in this research is a sequential architecture that processes image sequences and translates them into alphabetical outputs (A–Z). It starts with an input layer for image sequences, followed by a TimeDistributed layer using MobileNetV2 to extract spatial features from individual frames. These features are reshaped and passed through a second TimeDistributed layer for temporal preparation. An LSTM layer then captures temporal dynamics across frames. A dropout layer reduces overfitting by randomly deactivating neurons during training. Finally, a Dense layer maps the features to a 26-dimensional output representing the A–Z classes.

The model was trained for 35 epochs. During the first 10, transfer learning was applied by freezing all MobileNetV2 layers to train only the newly added layers. This allowed the model to learn general visual features without changing pre-trained weights. In the remaining 25 epochs, the last three blocks of MobileNetV2 were unfrozen and fine-tuned to better adapt to BISINDO hand signs by refining higher-level visual features relevant to the dataset.

TABLE II: Summary of the model

Accuracy	Loss	Validation Accuracy	Validation Loss
0.9606	0.5048	0.9530	0.4402

As a result, the model achieved outstanding performance, with a training accuracy of **0.9606** and training loss of **0.5048**, while the validation accuracy and loss reached **0.9530** and **0.4402** respectively, as shown in Table II. These metrics indicate a well-generalized model with minimal overfitting. Furthermore, the training dynamics are illustrated in Fig. 7a and Fig. 7b, which presents the loss and accuracy curves over the 35 epochs. The graphs show a consistent improvement in both training and validation performance, with noticeable spikes around epoch 10 due to the transition from frozen to fine-tuned MobileNetV2 layers. Despite that, the model quickly recovered and continued improving, converging to high accuracy with stable loss. Additionally, the final model size remained under **25 MB** with an average inference time of approximately **40-50 ms**, indicating its compatibility with low-resource environments. This efficiency is largely attributed to the use of regularization and post-training quantization, which helped reduce computational load without sacrificing performance.

Despite the excellent overall performance, as demonstrated in the confusion matrix shown in Fig. 8, the model still has some limitations, particularly in distinguishing between sign language gestures that are visually similar. Misclassifications are most evident between certain letters such as ‘M’ and ‘N’, as well as ‘Q’ and ‘S’, where overlapping features may have caused ambiguity during prediction. These errors are primarily due to the relatively small and limited dataset used, which restricts the model’s exposure to diverse hand shapes, orientations, and user variations. Future research should consider increasing and augmenting the dataset significantly to enhance the model’s robustness and ensure more reliable classification of visually similar signs.

V. Conclusion

This research presents a lightweight hybrid CNN-LSTM architecture designed for BISINDO (Indonesian Sign Language) alphabet recognition, with a specific focus on low-resource environments. By combining MobileNetV2 for spatial feature extraction and LSTM for temporal modeling, the model effectively captures both visual and sequential aspects of hand gestures. Optimization techniques such as L2 regularization and post-training quantization significantly improved the model’s efficiency without compromising accuracy, resulting

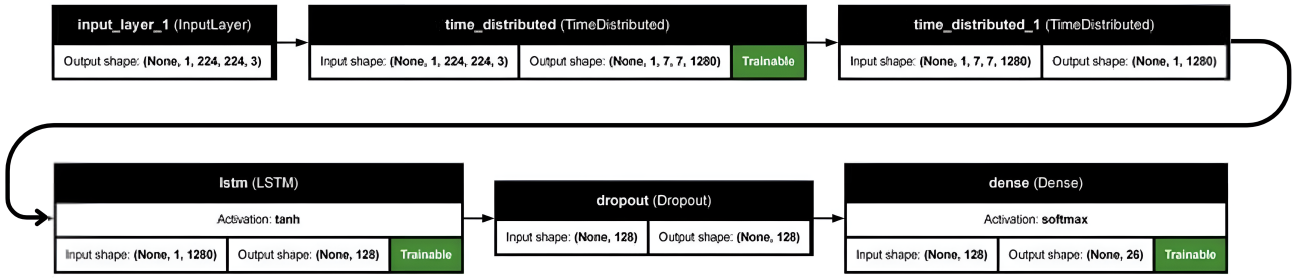
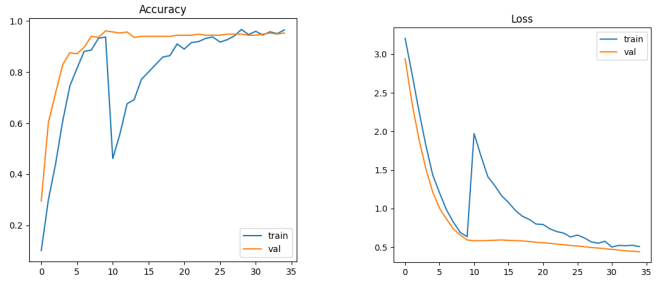


Fig. 6: Model developed.



(a) Train and validation accuracy over the 35 epochs.

(b) Train and validation loss over the 35 epochs.

Fig. 7: Training and validation metrics over 35 epochs.

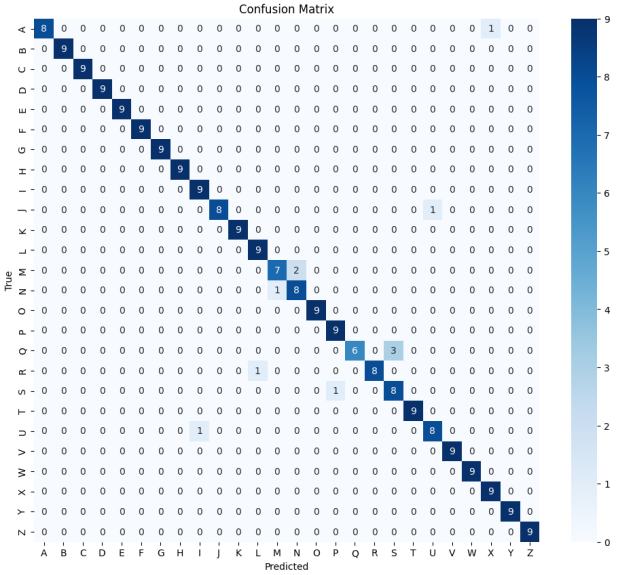


Fig. 8: Confusion Matrix

in a compact model under 25 MB and an average inference time of less than 50 ms (ideal for real-time deployment on budget devices).

The model achieved outstanding performance, with 96.06% training accuracy and 95.30% validation accuracy, demonstrating strong generalization as supported by consistent training curves and a low validation loss. However, some limitations remain, particularly in recognizing visually similar signs (e.g., 'M' vs. 'N', 'Q' vs. 'S'), primarily due to the limited dataset size. Future research should prioritize dataset expansion and diversification to improve model robustness. Additionally, exploring real-time continuous recognition, multi-modal input integration, and signer-independent generalization would further enhance practical applications. Overall, this study successfully bridges the gap between performance and accessibility, offering a viable solution for inclusive communication in resource-constrained communities.

VI. CRediT Authorship Contribution Statement

Muhammad Anang Ayman Ramadhana: Conceptualization, Methodology, Software, Investigation, Resources, Writing – Original Draft, Writing – Review & Editing, Visualization. **Benediktus Darmawan:** Conceptualization, Methodology, Resources, Investigation, Writing – Original Draft, Visualization, Writing – Review & Editing. **Rhio Sutoyo:** Supervision, Writing – Review & Editing, Project Administration. **Nicholaus Hendrik Jeremy:** Supervision, Project Administration.

VII. Open Data

The dataset used can be accessed through the following link: https://binusianorg-my.sharepoint.com/personal/muhammad_ramadhana003_binus_ac_id/_layouts/15/guestaccess.aspx?share=EqmE8NujgftBrV6t6DDP4uIBpJ8NIxwKH2wcCQffMRhBJw

References

[1] 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," in *2022 International Conference on Data Analytics for Business and Industry (ICDABI)*, 2022, pp. 697–701.

[2] BPS. Knuth: Computers and typesetting. [Online]. Available: https://sensus.bps.go.id/topik/tabular/sp2022/145/0/0

[3] D. Kumari and R. S. Anand, "Isolated video-based sign language recognition using a hybrid cnn-lstm framework based on attention mechanism," *Electronics*, vol. 13, no. 7, 2024. [Online]. Available: https://www.mdpi.com/2079-9292/13/7/1229

[4] H. Luqman and E. Elalfy, "Utilizing motion and spatial features for sign language gesture recognition using cascaded cnn and lstm models," *Turkish J. Electr. Eng. Comput. Sci.*, vol. 30, pp. 2508–2525, 2022. [Online]. Available: https://api.semanticscholar.org/CorpusID:254242425

[5] P. Nguyen Huu, N. Nguyen Thi, and T. P. Ngoc, "Proposing posture recognition system combining mobilenetv2 and lstm for medical surveillance," *IEEE Access*, vol. 10, pp. 1839–1849, 2022.

[6] H. Lokhande and S. Ganorkar, "Object detection in video surveillance using mobilenetv2 on resource-constrained low-power edge devices," *Bulletin of Electrical Engineering and Informatics*, vol. 14, pp. 357–365, 02 2025.

[7] Z. Li, H. Li, and L. Meng, "Model compression for deep neural networks: A survey," *Computers*, vol. 12, no. 3, 2023. [Online]. Available: https://www.mdpi.com/2073-431X/12/3/60

[8] H. He, L. Huang, Z. Huang, and T. Yang, "The compression techniques applied on deep learning model," *Highlights in Science, Engineering and Technology*, 2022. [Online]. Available: https://api.semanticscholar.org/CorpusID:251339623

[9] WHO, "Deafness and hearing loss," Feb 2025. [Online]. Available: https://www.who.int/health-topics/hearing-loss

[10] V. Jadhav, P. Agarwal, D. Mondhe, R. Patil, and L. Challissery Samu, "A survey of sign language recognition systems," *Journal of Innovative Image Processing*, vol. 4, pp. 237–246, 12 2022.

[11] R. Holmes, E. Rushe, F. Fowley, and A. Ventresque, "Improving signer independent sign language recognition for low resource languages," in *Proceedings of the 7th International Workshop on Sign Language Translation and Avatar Technology: The Junction of the Visual and the Textual: Challenges and Perspectives*, E. Efthimiou, S.-E. Fotinea, T. Hanke, J. C. McDonald, D. Shterionov, and R. Wolfe, Eds. Marseille, France: European Language Resources Association, Jun. 2022, pp. 45–52. [Online]. Available: https://aclanthology.org/2022.sltat-1.7/

[12] T. Kim and B. Kim, "Techniques for detecting the start and end points of sign language utterances to enhance recognition performance in mobile environments," *Applied Sciences*, vol. 14, no. 20, 2024. [Online]. Available: https://www.mdpi.com/2076-3417/14/20/9199

[13] B.-A. Awaluddin, C.-T. Chao, and J.-S. Chiou, "Investigating effective geometric transformation for image augmentation to improve static hand gestures with a pre-trained convolutional neural network," *Mathematics*, vol. 11, no. 23, 2023. [Online]. Available: https://www.mdpi.com/2227-7390/11/23/4783

[14] S.-G. Choi, Y. Park, and C.-B. Sohn, "Dataset transformation system for sign language recognition based on image classification network," *Applied Sciences*, vol. 12, no. 19, 2022. [Online]. Available: https://www.mdpi.com/2076-3417/12/19/10075

[15] B. Sundar and T. Bagyammal, "American sign language recognition for alphabets using mediapipe and lstm," *Procedia Computer Science*, vol. 215, pp. 642–651, 2022, 4th International Conference on Innovative Data Communication Technology and Application. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1877050922021378

[16] H. Lee, N. Lee, and S. Lee, "A method of deep learning model optimization for image classification on edge device," *Sensors*, vol. 22, no. 19, 2022. [Online]. Available: https://www.mdpi.com/1424-8220/22/19/7344

[17] C. Neidle, "American sign language lexicon video dataset (asllvd) corpus," Nov 2024. [Online]. Available: https://www.academia.edu/87376066/AmericanSignLanguageLexiconVideoDatasetASLLVDcorpus

[18] O. Koller, J. Forster, and H. Ney, "Continuous sign language recognition: Towards large vocabulary statistical recognition systems handling multiple signers," *Computer Vision and Image Understanding*, vol. 141, pp. 108–125, Dec. 2015.

[19] H. Zhou, W. Zhou, W. Qi, J. Pu, and H. Li, " Improving Sign Language Translation with Monolingual Data by Sign Back-Translation ," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Los Alamitos, CA, USA: IEEE Computer Society, Jun. 2021, pp. 1316–1325. [Online]. Available: https://doi.ieeecomputersociety.org/10.1109/CVPR46437.2021.00137

[20] A. Noer, "Bahasa isyarat indonesia (bisindo) alphabets," Nov 2021. [Online]. Available: https://www.kaggle.com/datasets/achmadnoer/alfabet-bisindo

[21] D. Joan, V. Vincent, K. Daniel, S. Achmad, and R. Sutoyo, "Bisindo hand-sign detection using transfer learning," 12 2023, pp. 1–7.

[22] A. Gupta, A. Sawan, S. Singh, and S. Kumari, "Dynamic sign language recognition with hybrid cnn-lstm and 1d convolutional layers," in *2024 11th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO)*, 2024, pp. 1–6.

[23] A. Munandar, Z. Yunizar, and S. Retno, "Indonesian sign language (bisindo) alphabet detection system using yolo (you only look once) algorithm," *Proceedings of Malikussaleh International Conference on Multidisciplinary Studies (MICoMS)*, vol. 4, p. 00001, 12 2024.