Jurnal Teknik Informatika (JUTIF)
P-ISSN: 2723-3863
E-ISSN: 2723-3871

Vol. 6, No. 4, Agustus 2025, Page. 2602-2624
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.4.4953

VGG16-Based Feature Extraction for Arabic Alphabet Sign Language Classification to Support Qur'anic Tadarus Accessibility

Aris Rakhmadi[*1], Anton Yudhana[2], Sunardi[3]

[1]Department of Informatics Engineering, Universitas Muhammadiyah Surakarta, Indonesia
[1]Department of Informatics, Universitas Ahmad Dahlan, Indonesia
[2,3]Department of Electrical Engineering, Universitas Ahmad Dahlan, Indonesia

Email: [1]aris.rakhmadi@ums.ac.id

Received : Jun 25, 2025; Revised : Aug 12, 2025; Accepted : Aug 14, 2025; Published : Aug 24, 2025

Abstract

This study addresses the limited availability of automated recognition systems for Arabic Alphabet Sign Language (ArSL), particularly in facilitating Qur'anic Tadarus for the deaf and hard-of-hearing community. While research on American and Indonesian sign languages has advanced significantly, ArSL studies, especially for static alphabet gestures, remain underrepresented. The aim of this research is to develop an accurate and efficient ArSL classifier using the VGG16 convolutional neural network with transfer learning. The study employs the publicly available RGB Arabic Alphabets Sign Language Dataset, comprising 7,856 annotated images across 31 Hijaiyah letters, collected under varied backgrounds and lighting conditions. The proposed model integrates pretrained ImageNet weights with a customized classification head, trained through a two-stage fine-tuning process with data augmentation. The model achieves 97.07% test accuracy, performing competitively against a ResNet-18 baseline (98.0%) while offering a simpler architecture suitable for resource-constrained deployments. Evaluation using precision, recall, F1-score, and confusion matrix shows consistently high performance, with minor misclassifications among visually similar letters. This work demonstrates a novel application of VGG16-based deep learning for ArSL recognition, contributing to inclusive religious education and accessibility technologies.

Keywords: *Arabic Sign Language, Deep Learning, Qur'anic Accessibility, Transfer Learning, VGG16.*

1. INTRODUCTION

For people who are deaf or have hearing loss, sign language is a key tool for effective communication worldwide. It serves not only as a medium for daily interactions but also as a key enabler for educational participation, religious practices, and broader social inclusion [1], [2]. For the Muslim community, particularly those with hearing impairments, accessing religious education, such as Qur'anic recitation, presents a significant challenge due to the lack of accessible resources and technological support.

The Arabic alphabet is fundamental in the context of religious education, especially in reading the Qur'an. Arabic Alphabet Sign Language (ArSL) consists of 31 distinct manual gestures representing each letter of the Arabic script [3], [4]. Mastery of these gestures is essential for spelling, word construction, and religious literacy. Despite its importance, the development of automatic recognition systems for Arabic alphabet sign language remains underrepresented compared to other sign languages, such as American Sign Language (ASL) [5] or Indonesian Sign Language (SIBI) [6].

Recent developments in artificial intelligence, particularly in computer vision and deep learning, have significantly advanced the field of Sign Language Recognition (SLR). Convolutional Neural Networks (CNNs) have shown remarkable capability in capturing spatial features from visual data, resulting in substantial improvements in the recognition of static hand gestures across different sign

Vol. 6, No. 4, Agustus 2025, Page. 2602-2624
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.4.4953

languages [7], [8]. A wide range of research has confirmed the efficacy of CNNs in tasks involving alphabet-based sign recognition, facilitating precise and rapid translation of hand gestures into written or spoken language [9], [10].

However, research on Arabic sign language recognition, particularly for the alphabet, is still limited [11]. Most existing works focus on small datasets, handcrafted features, or classical machine learning approaches, which often lack scalability and robustness when applied to diverse real-world conditions. Furthermore, while many studies address general communication signs, few have concentrated on alphabet-based recognition as a foundational tool for literacy and religious education.

Comparative analyses between Arabic and non-Arabic sign language research also reveal significant gaps. Studies on American Sign Language (ASL) in [12] have benefited from large-scale, well-annotated datasets collected from diverse sources, enabling the development of deep learning models with high accuracy and strong generalization across various users and environments. These advancements in ASL research demonstrate how comprehensive datasets and rigorous model training can substantially improve recognition performance [13], [14].

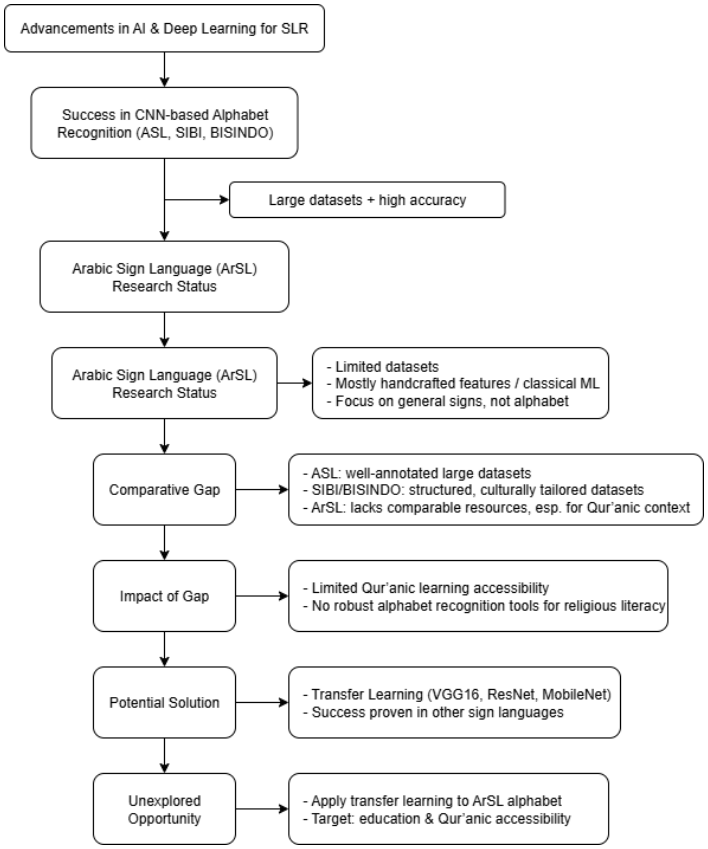


Figure 1. Literature Gap in Arabic Sign Language Research for Qur'anic Accessibility

Similarly, research on Indonesian Sign Language, including both SIBI and BISINDO, has also leveraged structured datasets and consistent annotation protocols as reported in [15], [16], [17], [18]. Such resources allow the creation of models tailored to the linguistic and cultural characteristics of the Indonesian deaf community. In contrast, Arabic Sign Language (ArSL) still lacks comparable dataset availability and domain-specific model development, particularly for applications that could facilitate Qur'anic education and religious accessibility for the deaf community. These shortcomings are further illustrated in the literature gap diagram (Figure 1), which contrasts the maturity of ASL, SIBI, and BISINDO research with the comparatively limited progress in ArSL, particularly within Qur'anic educational contexts.

Vol. 6, No. 4, Agustus 2025, Page. 2602-2624
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.4.4953

This lack of technological advancement creates a critical gap, particularly in enabling accessible Qur'anic learning for the deaf community [19], [20], [21]. Current technologies do not sufficiently support the recognition of Arabic alphabet gestures, which is a prerequisite for helping individuals engage in Qur'anic Tadarus, an essential practice in Islamic education that involves reading, reciting, and reflecting upon the Qur'an [22].

Deep learning approaches, especially those utilizing transfer learning with pretrained models, have developed as a hopeful solution to address these challenges. Various image classification problems have been effectively addressed using models like VGG16, ResNet, and MobileNet, which have shown outstanding accuracy [23], [24]. Among these, VGG16 stands out for its balance between architectural simplicity and classification accuracy, making it well-suited for tasks involving static gesture recognition with limited datasets [25].

Despite the success of transfer learning in other sign language contexts, its application to Arabic alphabet sign language, specifically in support of religious learning, remains underexplored [26], [27]. Beyond advancing sign language recognition, this initiative plays a key role in promoting broader digital inclusion for the Muslim deaf and hard-of-hearing population.

Addressing this research gap requires the development of a reliable, efficient, and accurate Arabic alphabet sign language recognition system that can serve as a foundation for educational tools, particularly those aimed at facilitating Qur'anic learning. Such a system would significantly enhance accessibility, allowing individuals with hearing disabilities to participate more fully in religious education and community life.

This study proposes an Arabic Alphabet Sign Language classifier based on the VGG16 deep learning architecture using transfer learning. The model is designed to classify static hand gesture images into 31 classes of the Arabic alphabet. The research aims to contribute both technically, by enhancing gesture recognition accuracy, and socially, by supporting inclusive education and religious accessibility for the deaf community.

2. METHOD

This study aims to develop a strong Arabic Alphabet Sign Language (ArSL) classifier using deep learning techniques. Specifically, we employ a VGG16 model with pretrained weights, which helps in feature extraction, followed by the classification of 31 distinct Arabic alphabet letters in ArSL. The methodology is arranged into three main phases: dataset preparation and preprocessing, model architecture and training, and model evaluation and finalization. Each of these phases, as illustrated in Figure 2, plays a critical role in ensuring that the model performs optimally and generalizes well to unseen data.

2.1. Dataset Preparation and Preprocessing

In this study, the dataset employed was the RGB Arabic Alphabets Sign Language Dataset, which is publicly available on Kaggle and compiled by Muhammad Albarham [28]. It comprises 7,856 RGB images, each representing a static hand gesture for one of the 31 Hijaiyah letters. The images exhibit variations in background, lighting, and hand orientation, providing diversity in visual characteristics that enhance the dataset's suitability for sign language recognition tasks. Representative examples illustrating these variations are shown in Figure 3. Each image is manually annotated and stored in Parquet format, an efficient storage type for large-scale data. The dataset is divided into three subsets: training, validation, and testing [29], [30]. The training subset is used for model fitting, the validation subset for hyperparameter tuning, and the testing subset, kept entirely separate from training, for final performance evaluation. This stratified division helps minimize overfitting and ensures a reliable assessment of the model's generalization capability.

Vol. 6, No. 4, Agustus 2025, Page. 2602-2624
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.4.4953

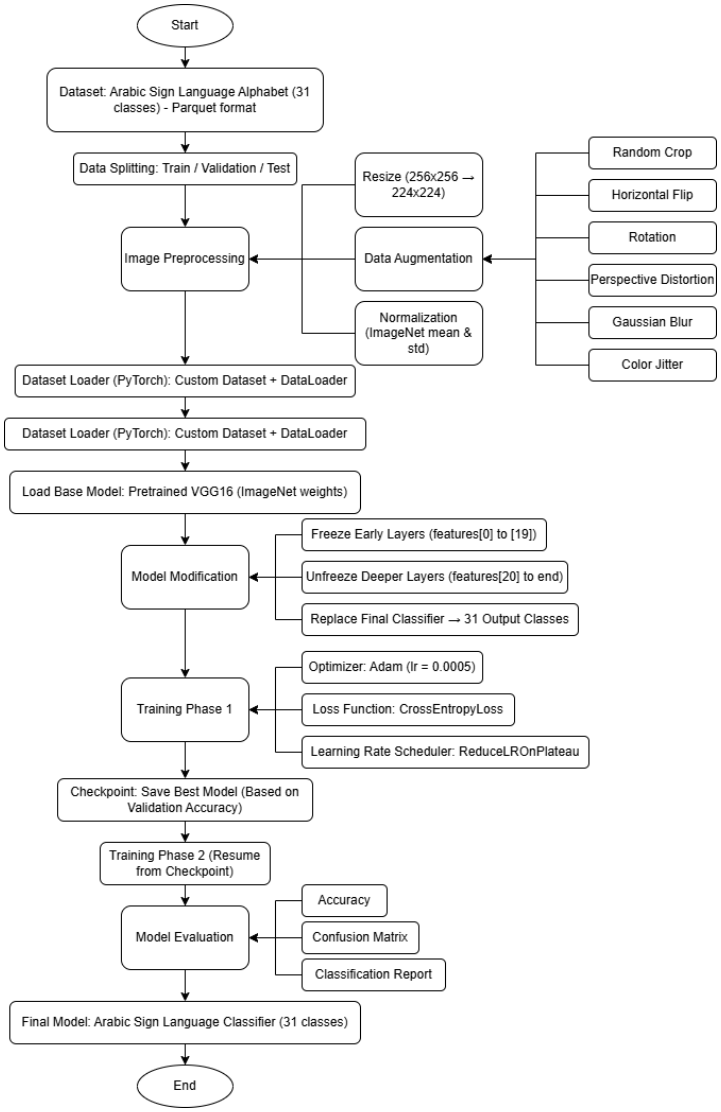


Figure 2. Workflow of the VGG16-Based ArSL-A Recognition System

To ensure uniformity and compatibility with the VGG16 model, it is necessary to resize all images to the same dimensions. Initially, the images in the dataset are of size 256 × 256 pixels. These are resized to 224x224 pixels, which is the required input size for the VGG16 architecture. This resizing step is critical because the model expects fixed-size inputs, and varying image sizes could lead to performance issues during the training process. Standardizing image dimensions also helps optimize computational resources during model training [31].

To enhance the model's robustness and its ability to perform effectively in diverse real-world environments, a range of data augmentation techniques is employed [32]. These include random cropping, horizontal flipping, rotation, perspective distortion, Gaussian blur, and color jitter. Such transformations artificially increase the variability within the dataset by simulating different visual conditions, such as varying angles, lighting, and distortions. Incorporating this variability into the training process helps the model become more resilient to noise and fluctuations commonly encountered in real-world scenarios, thereby improving its generalization capabilities and classification performance.

Normalization serves as a crucial preprocessing step by scaling the pixel values of the images to a consistent range. In this study, normalization is performed using the mean and standard deviation values derived from ImageNet, the large-scale dataset on which VGG16 was originally trained. This process facilitates faster convergence during training by ensuring that all input features share a similar

Vol. 6, No. 4, Agustus 2025, Page. 2602-2624
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.4.4953

scale, thereby preventing any single feature from disproportionately influencing the learning process. As a result, the model can more effectively identify meaningful patterns for Arabic Sign Language (ArSL) classification, ultimately enhancing both the optimization process and the overall learning performance.

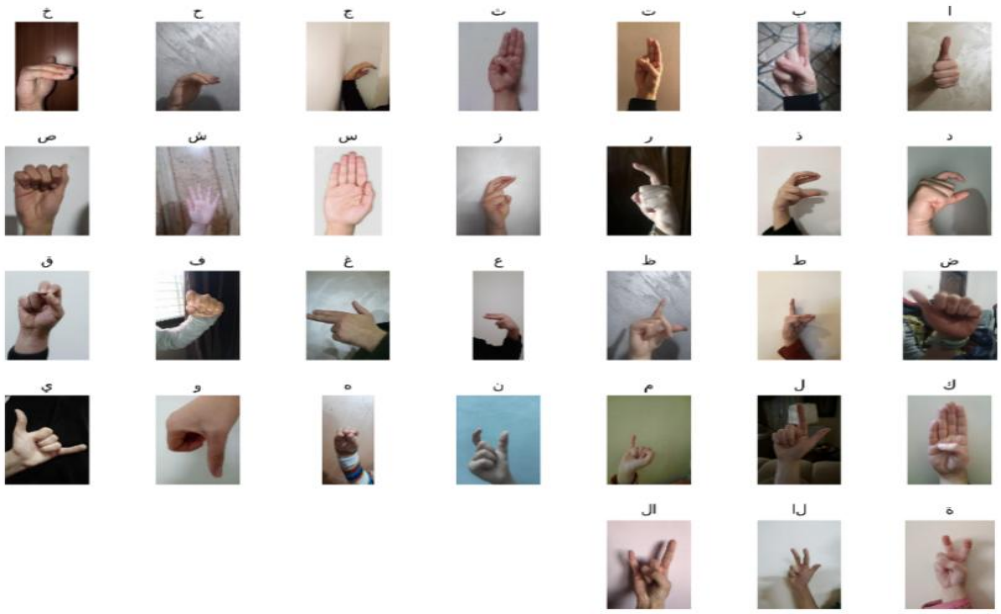


Figure 3. Example RGB images from the Arabic Alphabet Sign Language Dataset, illustrating variations in hand shapes and background settings across different classes.

2.2. Model Architecture and Training

The second phase of the methodology involves selecting a suitable model architecture, followed by the training process. In this phase, the pretrained VGG16 model is fine-tuned to classify the 31 letters of the Arabic alphabet in sign language. This ensures that the model can effectively recognize ArSL gestures. VGG16 is a deep convolutional neural network (CNN) that has proven highly effective in various image classification tasks. For this study, we utilize the VGG16 model, which is pretrained on ImageNet. Transfer learning is used to leverage the knowledge acquired by VGG16 from millions of images across diverse categories. By initializing the model with pre-trained weights, we can significantly reduce training time and improve the model's performance. This strategy enables the model to begin with robust feature extraction capabilities, which can be further refined for the specific task of ArSL classification.

To adapt the VGG16 model for the ArSL classification task, several modifications are made. First, the early layers (features [0] to [19]) of the VGG16 model are frozen. Freezing these layers ensures that the model retains the general, low-level features, such as edges, textures, and simple shapes, that are common across various image types. These features are not specific to ArSL and can be reused for this task. The deeper layers (features [20] to the end) are unfrozen, allowing them to learn more complex, task-specific features that are crucial for recognizing the hand gestures in ArSL. Additionally, the final fully connected layer is replaced with a new classifier that produces 31 output classes, corresponding to the 31 letters of the Arabic alphabet. This modification ensures that the model can output the correct ArSL gesture classification.

To enhance the model's training efficiency and performance, the Adam optimizer is utilized [33], [34]. This adaptive optimization algorithm dynamically adjusts the learning rate for each parameter, contributing to a more stable and accelerated training process. A learning rate of 0.0005 is chosen,

Vol. 6, No. 4, Agustus 2025, Page. 2602-2624
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.4.4953

balancing the need for meaningful weight updates with the risk of overshooting optimal solutions. For the loss function, CrossEntropyLoss is employed, which is well-suited for multi-class classification problems. It quantifies the discrepancy between the predicted class probabilities and the actual labels, penalizing incorrect predictions and guiding the model toward improved classification accuracy over successive training iterations.

To further improve the training process, a learning rate scheduler is introduced. Specifically, the ReduceLROnPlateau scheduler is used, which reduces the learning rate when the validation loss stagnates or plateaus. This strategy helps to fine-tune the model as it nears optimal performance. By gradually lowering the learning rate, the model can make more refined adjustments to its weights, ensuring that it converges to a better solution and avoids overshooting the optimal weights. This helps the model achieve better results, particularly in the later stages of training.

The training process is divided into two distinct phases to ensure more effective and controlled learning [35]. In the initial phase, only the newly added classifier layers are trained, while the pretrained convolutional layers of VGG16 remain frozen. This approach allows the model to adapt to the specific characteristics of Arabic Sign Language (ArSL) while preserving the general low-level features learned from ImageNet. Throughout this phase, the model is trained using the training dataset and evaluated periodically on the validation set. Validation performance is monitored to track learning progress and to detect any signs of overfitting to the training data.

To preserve the best version of the model, checkpointing is employed. The model is saved at regular intervals based on its performance on the validation set, specifically when it achieves a higher validation accuracy [36]. This ensures that the most optimal version of the model is retained, and training can be resumed from the best checkpoint in case of interruptions. This step is instrumental in avoiding the loss of progress during lengthy training processes.

In the second phase of training, the entire VGG16 network is unfrozen, allowing all layers to be updated. This fine-tuning phase enables the model to refine its learned feature representations, thereby enhancing its ability to accurately recognize Arabic Sign Language (ArSL) gestures. A lower learning rate is employed during this stage to ensure that weight adjustments are subtle and precise, minimizing the risk of disrupting previously learned features. By allowing the network to adapt more deeply to the ArSL dataset, this phase helps the model capture complex patterns and subtle variations, leading to improved overall performance.

2.3. Model Evaluation and Finalization

After the training process is complete, the model is evaluated using the test set, comprising data that was not exposed to the model during training, making it suitable for assessing the model's ability to generalize to unseen inputs [37]. Multiple evaluation metrics are employed to comprehensively assess performance, including Accuracy, Precision, Recall, F1-Score, and the Confusion Matrix.

The mathematical definitions of these metrics are as follows:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

$$F1 - Score = 2\ x\ \frac{Precision\ x\ Recall}{Precision+Recall} \quad (4)$$

Vol. 6, No. 4, Agustus 2025, Page. 2602-2624
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.4.4953

Where:

- TP = True Positives
- TN = True Negatives
- FP = False Positives
- FN = False Negatives

For a more granular analysis, a confusion matrix is used to visualize class-specific performance, revealing patterns of misclassification across gesture categories [38]. Furthermore, a classification report is generated, providing key metrics such as precision, recall, and F1-score. These metrics offer deeper insight into the model's effectiveness in correctly identifying gestures while minimizing both false positives and false negatives.

To contextualize the performance of the proposed VGG16-based ArSL classifier, we also compare its results with those of previously reported methods on the same dataset. Only methods that use the RGB Arabic Alphabets Sign Language Dataset [28] or an equivalent dataset with identical class definitions are considered. The evaluation metrics (Accuracy, Precision, Recall, and F1-Score) and testing protocol are kept consistent across all methods to ensure fairness in comparison. For published studies where only partial metrics are reported, missing values are derived from the available confusion matrices or classification reports. This methodological alignment allows for a direct and unbiased assessment of the relative strengths and weaknesses of the proposed model compared to existing approaches.

After evaluation, the final ArSL classifier is obtained. This model is capable of classifying images of ArSL gestures into 31 distinct classes, each corresponding to a letter in the Arabic alphabet. The model's high performance demonstrates its effectiveness in solving the ArSL classification task and its potential for use in real-world applications [39]. With its robust ability to accurately classify gestures, the model holds significant promise for supporting sign language users in communication, particularly in applications like Qur'anic Tadarus, which can be made more accessible to the hearing-impaired community.

3. RESULT

The dataset used in this study consists of 7,856 RGB images representing 31 static hand gestures of the Arabic Alphabet Sign Language (ArSL). Table 1 presents the number of images for each class, along with their names in Indonesian, English, and Arabic script. The distribution shows that the dataset is relatively balanced, with the number of images per class ranging from 201 to 307, minimizing the risk of bias toward any particular class during model training. The most populated class is Ba' with 307 images, while the least populated is Zain with 201 images. The relatively even distribution ensures that no single class dominates the dataset, thereby supporting fair and unbiased training and evaluation. This balanced composition also reduces the likelihood that the high overall model performance is driven solely by majority class recognition.

The performance of the Arabic Sign Language (ArSL) classification model was evaluated using several key metrics. These included training and validation accuracy/loss curves, the confusion matrix, and performance indicators such as precision, recall, and F1-score [40]. These metrics provide a comprehensive understanding of how well the model performed in classifying the 31 letters of the Arabic alphabet in ArSL, shedding light on both its strengths and areas for improvement.

The model's performance was further analyzed through various metrics. The training and validation accuracy and loss were examined to assess the model's learning process and generalization capabilities [41]. The confusion matrix was explored to identify common misclassifications and understand the challenges in distinguishing visually similar ArSL gestures. Finally, precision, recall,

Vol. 6, No. 4, Agustus 2025, Page. 2602-2624
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.4.4953

and F1-score were presented to evaluate the model's effectiveness in classifying each ArSL gesture and its overall performance across all classes.

Table 1. Distribution of Images per Class in the Arabic Alphabet Sign Language (ArSL) Dataset

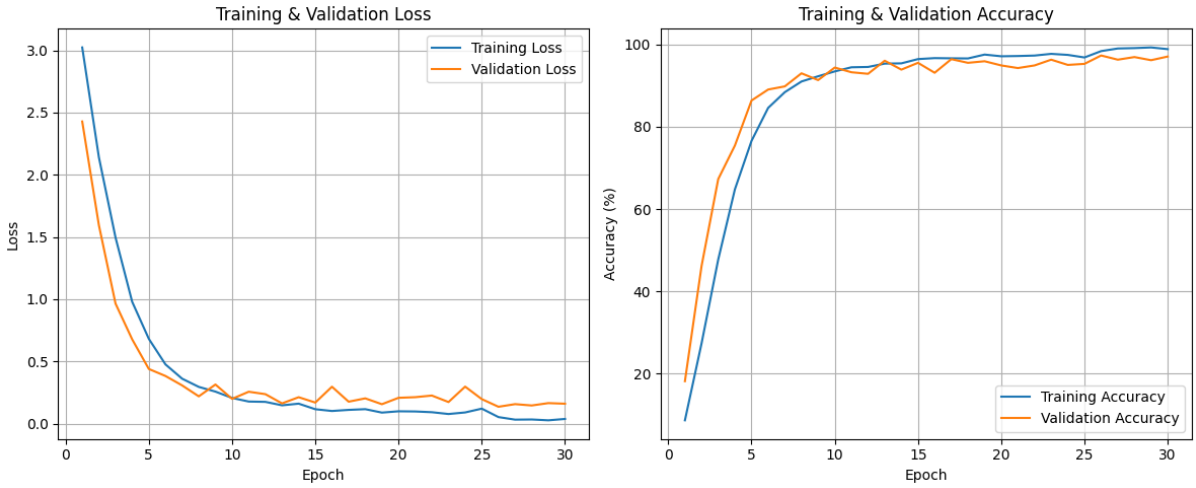
No.	Indonesian Name	English Name	Arabic Script	Number of Images
1	Alif	ALEF	أ (ألف)	287
2	Ba’	BEH	ب (باء)	307
3	Ta’	TEH	ت (تاء)	226
4	Tsa’	THEH	ث (ثاء)	305
5	Jim	JEEM	ج (جيم)	210
6	Ha	HAH	ح (حاء)	246
7	Kha	KHAH	خ (خاء)	250
8	Dal	DAL	د (دال)	235
9	Dzal	THAL	ذ (ذال)	202
10	Ra’	REH	ر (راء)	227
11	Zain	ZAIN	ز (زاي)	201
12	Sin	SEEN	س (سين)	266
13	Syin	SHEEN	ش (شين)	278
14	Shad	SAD	ص (صاد)	270
15	Dhad	DAD	ض (ضاد)	266
16	Tha’	TAH	ط (طاء)	227
17	Zha	ZAH	ظ (ظاء)	232
18	‘Ain	AIN	ع (عين)	244
19	Ghain	GHAIN	غ (غين)	231
20	Fa’	FEH	ف (فاء)	255
21	Qaf	QAF	ق (قاف)	219
22	Kaf	KAF	ك (كاف)	264
23	Lam	LAM	ل (لام)	260
24	Mim	MEEM	م (ميم)	253
25	Nun	NOON	ن (نون)	237
26	Ha	HEH	هـ (هاء)	253
27	Waw	WAW	و (واو)	249
28	Ya’	YEH	ي (ياء)	272
29	Ta’ Marbuthah	TEH MARBUTA	ة (تاء مربوطة)	257
30	Alim Lam	AL	ال	276
31	Lam Alif	LAA	لا	268

3.1. Performance Evaluation of the Model

The performance evaluation of the VGG16-based Arabic Alphabet Sign Language (ArSL) classifier is primarily centered on assessing its learning dynamics during the training and validation phases. Figure 4 presents the learning curves for both training and validation, measured in terms of accuracy and loss across 30 epochs. These curves provide valuable insights into how well the model adapts to the training data and generalizes to unseen validation data. From the outset, the model demonstrates a rapid reduction in both training and validation loss, indicating effective learning from the input data during the early epochs. This steep decline is particularly evident within the first 10

Vol. 6, No. 4, Agustus 2025, Page. 2602-2624
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.4.4953

epochs, suggesting that the pretrained VGG16 architecture with frozen early layers facilitates the fast acquisition of foundational visual features relevant to ArSL gestures.



(a) (b)

Figure 4. (a) Training and Validation Loss Curves and (b) Training and Validation Accuracy Curves of the VGG16-Based Arabic Alphabet Sign Language Classifier.

The trajectory of the training loss reveals a sharp descent from an initial value of approximately 3.0 to below 0.5 within the first five epochs. This rapid convergence continues steadily, with the training loss reducing to as low as 0.0334 by epoch 28. Meanwhile, the validation loss exhibits a similar decreasing trend, dropping from 2.4 in the first epoch to approximately 0.146 by epoch 28. These results reflect a highly stable optimization process where both the training and validation losses converge without significant divergence, indicating a well-regularized model. The close alignment between the training and validation losses throughout the epochs serves as an essential indicator that the model successfully avoids overfitting.

In parallel, the accuracy curves for both training and validation exhibit a consistent upward trend. Beginning from a relatively low baseline (below 20% in the initial epoch), the model achieves substantial improvements in accuracy rapidly. By epoch 10, the validation accuracy already surpasses 90%, while training accuracy follows closely, reaching comparable levels. The consistency between the training and validation accuracy suggests that the learned representations are generalizable and not merely memorizing the training data. This pattern underscores the effectiveness of the transfer learning strategy employed, wherein the pretrained convolutional layers capture universal low-level patterns, while the fine-tuned upper layers specialize in the intricacies of ArSL gesture recognition.

The model continues to refine its performance beyond epoch 10, with training accuracy eventually reaching a peak of 99.28% at epoch 29. Concurrently, the highest validation accuracy achieved is 97.33%, indicating excellent generalization capabilities. Notably, this peak validation accuracy does not occur precisely at the final epoch but rather slightly earlier, suggesting that the model achieves optimal performance before full convergence. This behavior is common in deep learning models where continued training beyond the optimal point may lead to minor oscillations in accuracy due to stochastic gradients, albeit without significant signs of overfitting in this case.

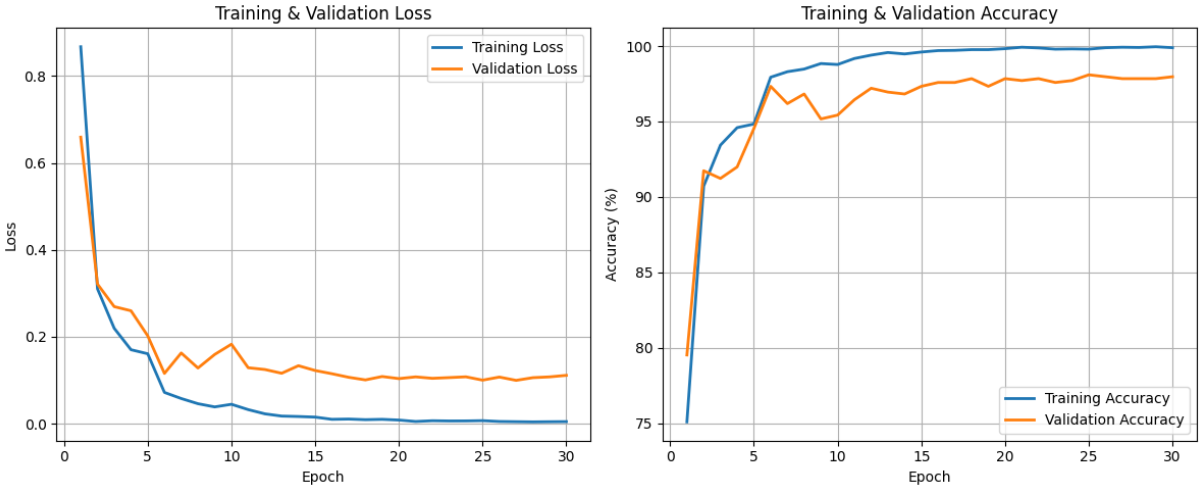
Examining the final epochs provides additional confirmation of the model's robustness. At epoch 30, the training loss stands at 0.0377, while the validation loss slightly increases to 0.1601. The corresponding training and validation accuracies are 98.87% and 97.07%, respectively. Although there is a minor increase in validation loss towards the end, this fluctuation remains within an acceptable range

Vol. 6, No. 4, Agustus 2025, Page. 2602-2624
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.4.4953

and does not indicate a general degradation in performance. Such slight variations are expected in neural network training and can be attributed to the stochastic nature of mini-batch gradient descent.

Another significant observation from the learning curves is the effect of the learning rate scheduler (ReduceLROnPlateau). This scheduler effectively mitigates stagnation by adaptively lowering the learning rate whenever the validation loss plateaus. As seen in the curves, the periods following the reduction of the learning rate correspond to further improvements in validation accuracy and stabilization of loss. This adaptive learning rate strategy plays a crucial role in ensuring that the model fine-tunes effectively, especially during the second phase of training, where all layers of the VGG16 network are unfrozen.

The performance evaluation indicates that the proposed VGG16-based ArSL classifier exhibits outstanding learning behavior, characterized by fast convergence, high accuracy, and a minimal generalization gap. The close alignment between training and validation metrics across the epochs highlights the model's ability to capture relevant features of the ArSL dataset without overfitting. This high level of performance supports the viability of deploying this model in real-world applications aimed at improving accessibility for the hearing-impaired community, particularly in enabling Qur'anic Tadarus through sign language recognition.



(a) (b)

Figure 5. (a) Loss Curves for Training and Validation, and (b) Accuracy Curves for Training and Validation of the Resnet18-Based Arabic Alphabet Sign Language Classifier.

For comparative purposes, the learning curves of the ResNet-18 model from our previous study are also examined alongside the proposed VGG16 architecture. Both models were trained on the same RGB Arabic Alphabets Sign Language dataset, with identical preprocessing steps, image dimensions, and stratified data splitting (70% training, 15% validation, and 15% testing). The training and validation loss–accuracy curves for VGG16 are presented in Figure 4(a) and Figure 4(b), while the corresponding curves for ResNet-18 are shown in Figure 5(a) and Figure 5(b).

In terms of loss reduction, both models exhibit a steep decline during the early epochs, indicating rapid acquisition of discriminative features. The VGG16 model’s training loss dropped from approximately 3.0 to below 0.5 by epoch 5 and reached 0.0334 by epoch 28, with validation loss decreasing from 2.4 to around 0.146 in the same period. By comparison, ResNet-18 achieved faster convergence, with training loss falling below 0.08 by epoch 6 and validation loss stabilizing within the range of 0.10–0.13 after epoch 11, reflecting slightly more consistent generalization in the later training stages.

Vol. 6, No. 4, Agustus 2025, Page. 2602-2624
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.4.4953

The accuracy curves show a similar trend. VGG16's validation accuracy surpassed 90% by epoch 10 and peaked at 97.33% before stabilizing at 97.07% by the final epoch. ResNet-18 reached 97.33% as early as epoch 6 and achieved a peak validation accuracy of 98.09% at epoch 25. This difference suggests that the residual connections in ResNet-18 facilitate more efficient gradient flow, enabling faster convergence compared to the sequential convolutional blocks of VGG16.

Although ResNet-18 yielded slightly higher accuracy and more stable loss convergence, the proposed VGG16 model remains highly competitive, achieving minimal overfitting and maintaining a small gap between training and validation metrics. Considering its simpler architecture and reduced computational complexity relative to ResNet-18, VGG16 offers a viable alternative for deployment in resource-constrained environments without sacrificing substantial accuracy.

3.2. Confusion Matrix Analysis

The confusion matrix, as illustrated in Figure 3, plays a crucial role in offering a comprehensive evaluation of the VGG16-based Arabic Alphabet Sign Language (ArSL) classifier's performance. It provides a detailed breakdown of correct and incorrect predictions for each of the 31 classes, enabling a deeper understanding of how well the model differentiates between similar gestures. The confusion matrix highlights not only the overall predictive accuracy but also reveals specific patterns of misclassification, which are instrumental in diagnosing weaknesses in the model's learning behavior. When coupled with the class-wise accuracy statistics presented in Table 1, it provides a holistic view of the model's performance across all categories.

A close inspection of the confusion matrix (Figure 6) reveals a dominant diagonal trend, indicating that the vast majority of predictions are correct. This suggests that the model has successfully learned to capture the distinguishing features of most ArSL gestures. The corresponding accuracy values in Table 1 further support this observation, showing that 17 out of 31 classes achieved a flawless 100% accuracy. These classes include *Ha, Ain, Alif, Alif_Lam, Dhad, Dzal, Lam, Lam_Alif, Mim, Qaf, Sin, Syin, Ta_Marbuta, Tha, Tsa, Wau,* and *Zay*, each demonstrating perfect predictive performance with no misclassifications. This consistency between the confusion matrix and the class-wise accuracy table confirms the model's robustness in recognizing gestures that have clear and distinct visual characteristics.

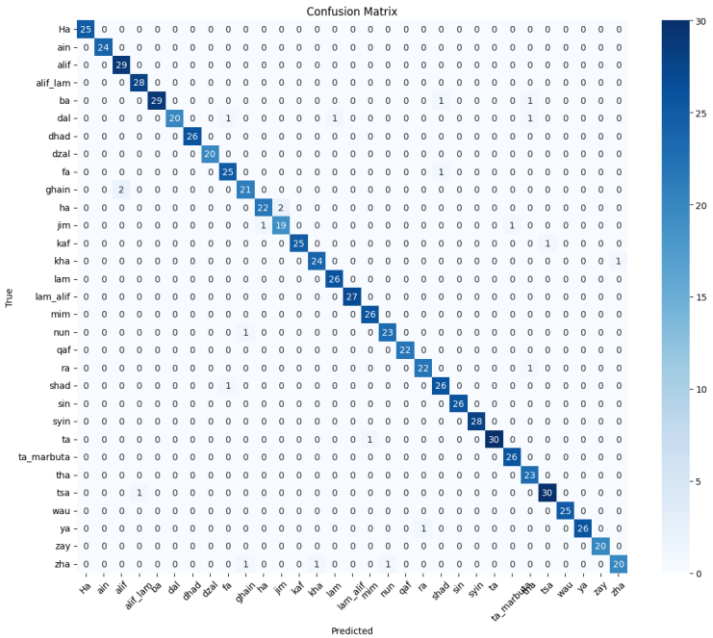


Figure 6. Confusion Matrix of the Proposed VGG16-based Arabic Alphabet Sign Language Classifier.

Vol. 6, No. 4, Agustus 2025, Page. 2602-2624
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.4.4953

The model's effectiveness on these perfectly classified classes can be attributed to several factors. Firstly, the distinctive hand shapes and orientations associated with these signs are likely highly separable in the feature space extracted by the VGG16 convolutional layers. Secondly, the data augmentation strategies employed during training, including rotations, cropping, flipping, and color jitter, help the model generalize better to unseen examples. Lastly, the transfer learning approach, leveraging pre-trained ImageNet weights, provides a strong foundation for recognizing visual features, which is then fine-tuned for the specific patterns present in ArSL gestures.

Table 2. Class-Wise Accuracy of the VGG16-Based ArSL Classifier.

No.	Class	Accuracy (%)
1	Ha	100.00
2	Ain	100.00
3	Alif	100.00
4	Alif_Lam	100.00
5	Ba	93.55
6	Dal	86.96
7	Dhad	100.00
8	Dzal	100.00
9	Fa	96.15
10	Ghain	91.30
11	Ha (Haa)	91.67
12	Jim	90.48
13	Kaf	96.15
14	Kha	96.00
15	Lam	100.00
16	Lam_Alif	100.00
17	Mim	100.00
18	Nun	95.83
19	Qaf	100.00
20	Ra	95.65
21	Shad	96.30
22	Sin	100.00
23	Syin	100.00
24	Ta	96.77
25	Ta_Marbuta	100.00
26	Tha	100.00
27	Tsa	96.77
28	Wau	100.00
29	Ya	96.30
30	Zay	100.00
31	Zha	86.96

Despite the strong overall performance, the confusion matrix and Table 1 collectively highlight a small subset of classes where the model struggles. The most notable is the *Dal* class, which achieves an accuracy of 86.96%, as shown in Table 2. The confusion matrix indicates that Dal is frequently misclassified as *Dzal* and *Zha*, both of which are phonetically and visually similar in Arabic Sign

Vol. 6, No. 4, Agustus 2025, Page. 2602-2624
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.4.4953

Language. This pattern suggests that the current feature representations may not be sufficiently sensitive to the subtle differences in finger positioning or hand orientation that distinguish these letters.

Similarly, the *Ghain* class demonstrates a reduced accuracy of 91.30%, with the confusion matrix showing multiple misclassifications into the *Ain* class. This confusion is linguistically and visually plausible, given that *Ain* and *Ghain* are articulated from the same pharyngeal region in spoken Arabic and may be represented with analogous hand configurations in sign language. The overlap in gesture features likely contributes to the observed model confusion, signaling a need for enhanced discriminative capabilities either through more specialized data or architectural improvements.

The *Jim* class, with an accuracy of 90.48%, presents another case where confusion occurs, particularly with the *Kaf* class. The confusion matrix reflects this, showing misclassification events between these two classes. This suggests that both gestures may share common visual traits, such as finger curvature or palm orientation, that are challenging for the model to disentangle. Likewise, the *Nun* class, which records an accuracy of 95.83%, is occasionally confused with *Mim* and *Qaf*. These gestures likely share morphological similarities in hand closure and finger placement, leading to sporadic misclassification.

Cross-class confusion is also observed between *Kaf* and *Kha*, both achieving accuracies above 96%, yet each with at least one misclassification toward the other. The confusion matrix corroborates this, implying that these gestures share structural hand features that are not always distinctly captured by the convolutional filters of VGG16. While these misclassifications are minor relative to the total sample size, they highlight the limitations of static image recognition in fully capturing the nuances of three-dimensional gestures.

Perhaps the most significant underperformance is observed in the *Zha* class, which, along with *Dal*, holds the lowest accuracy at 86.96%, as reported in Table 1. The confusion matrix shows dispersed predictions for *Zha*, potentially into related classes like *Dzal* and *Dal*. This is indicative of inherent visual similarities or ambiguities in the dataset that the current model architecture finds challenging to resolve. Such findings underscore the need for either higher-resolution imagery, more diverse data collection, or advanced model components like spatial attention mechanisms to focus on subtle discriminative regions of the hand.

Despite these localized issues, it is important to note that the overall distribution of correct versus incorrect predictions in the confusion matrix overwhelmingly favors correct classifications. This is further supported by the fact that the majority of classes maintain accuracies well above 95%, as indicated in Table 2. The few errors that occur are not randomly distributed but are systematically associated with visually or linguistically similar classes, suggesting that the model has effectively learned the dominant structure of the data but still struggles with edge cases involving high inter-class similarity.

The analysis of the confusion matrix, reinforced by the detailed class-wise accuracy provided in Table 2, confirms that the VGG16-based ArSL classifier demonstrates exceptional performance in recognizing Arabic alphabet gestures. The findings reveal a model that is highly reliable across most categories while also identifying specific gesture pairs where confusion persists. Addressing these limitations in future work may involve augmenting the dataset with more diverse samples, incorporating depth or motion cues, or integrating advanced deep learning techniques such as attention mechanisms. These enhancements are expected to further improve the classifier's capability, particularly in real-world applications aimed at supporting Qur'anic Tadarus accessibility for the deaf and hard-of-hearing community.

For comparative purposes, Figure 7 illustrates the confusion matrix of the ResNet-18 model reported in our previous work, trained and evaluated on the same RGB Arabic Alphabets Sign Language

Jurnal Teknik Informatika (JUTIF)
P-ISSN: 2723-3863
E-ISSN: 2723-3871
Vol. 6, No. 4, Agustus 2025, Page. 2602-2624
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.4.4953

dataset. Similar to the VGG16 confusion matrix shown in Figure 6, the ResNet-18 matrix exhibits a strong diagonal dominance, indicating that the majority of predictions fall into the correct class.

A closer inspection reveals that ResNet-18 achieves fewer misclassifications for certain challenging classes. For example, in the VGG16 results (Figure 6), the Dal class is often confused with Dzal and Zha, and Zha shows scattered misclassifications into related classes. In contrast, the ResNet-18 matrix reduces these errors, correctly classifying all but one sample for both Dal and Zha. This suggests that the residual connections in ResNet-18 may enhance its ability to capture fine-grained spatial differences between visually similar hand gestures.

Nevertheless, both models share similar confusion patterns for other gesture pairs, such as Ghain versus Ain and Jim versus Kaf, which remain difficult to separate in a purely static image recognition context. This indicates that certain ambiguities are more likely caused by inherent visual similarities in the gestures rather than architectural limitations.

Overall, while ResNet-18 offers a modest improvement in disambiguating some visually similar classes, the VGG16-based classifier still demonstrates competitive performance across most categories with a simpler architecture, making it suitable for deployment scenarios where computational resources are limited.

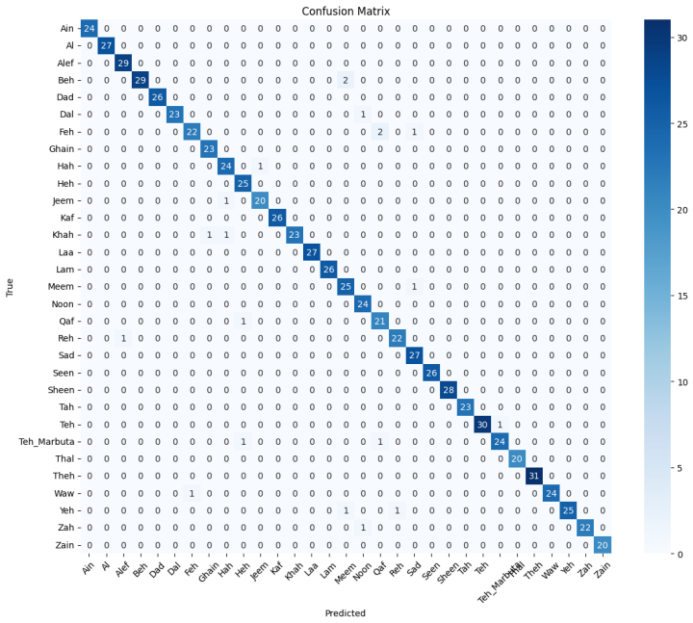


Figure 7. Confusion Matrix Illustrating ResNet-18 Model Performance on the 31-Class Arabic Sign Language Dataset.

3.3. Classification Report and Indicator Analysis

The classification report provides an essential quantitative summary of the model's performance, offering detailed insights into the precision, recall, and F1-score for each of the 31 Arabic alphabet sign language classes. As presented in Table 3, the classifier achieves an impressive overall test accuracy of 97.07% with a low test loss of 0.1135, underscoring the model's robustness and ability to generalize well to unseen data. These metrics are complemented by macro-averaged and weighted-averaged precision, recall, and F1 Scores, all of which equal 0.97, indicating balanced performance across both high-frequency and low-frequency classes within the dataset.

A closer inspection of Table 3 reveals that the majority of classes achieve exceptionally high performance, with precision, recall, and F1-scores nearing or equal to 1.00. Specifically, classes such as *Ha, Ain, Dhad, Dzal, Lam_Alif, Qaf, Sin, Syin, Wau,* and *Zay* demonstrate perfect scores across all three metrics. This level of accuracy reflects the model's capacity to perfectly capture the discriminative

Vol. 6, No. 4, Agustus 2025, Page. 2602-2624
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.4.4953

features of these signs, suggesting that these particular gestures exhibit highly distinct visual patterns in terms of hand shape, orientation, and positioning, which are well captured by the VGG16-based convolutional feature extractor.

Additionally, classes such as *Alif_Lam, Ta_Marbuta, Ta, Mim,* and *Lam* also maintain excellent performance with F1-scores ranging from 0.98 to 1.00, further indicating that the model effectively learns the characteristic representations of these gestures. This high degree of accuracy can be attributed to several factors, including effective data preprocessing, normalization aligned with ImageNet standards, and comprehensive data augmentation that simulates realistic variability in hand gestures. The transfer learning strategy further enhances this by allowing the model to utilize generalized low-level features while learning specific patterns relevant to the Arabic Sign Language (ArSL) dataset.

Table 3. Precision, Recall, and F1-Score for 31 ArSL Classes.

No.	Class	Precision	Recall	F1-Score	Support
1	Ha	1.00	1.00	1.00	25
2	Ain	1.00	1.00	1.00	24
3	Alif	0.94	1.00	0.97	29
4	Alif_Lam	0.97	1.00	0.98	28
5	Ba	1.00	0.94	0.97	31
6	Dal	1.00	0.87	0.93	23
7	Dhad	1.00	1.00	1.00	26
8	Dzal	1.00	1.00	1.00	20
9	Fa	0.93	0.96	0.94	26
10	Ghain	0.91	0.91	0.91	23
11	Ha (Haa)	0.96	0.92	0.94	24
12	Jim	0.90	0.90	0.90	21
13	Kaf	1.00	0.96	0.98	26
14	Kha	0.96	0.96	0.96	25
15	Lam	0.96	1.00	0.98	26
16	Lam_Alif	1.00	1.00	1.00	27
17	Mim	0.96	1.00	0.98	26
18	Nun	0.96	0.96	0.96	24
19	Qaf	1.00	1.00	1.00	22
20	Ra	0.96	0.96	0.96	23
21	Shad	0.93	0.96	0.95	27
22	Sin	1.00	1.00	1.00	26
23	Syin	1.00	1.00	1.00	28
24	Ta	1.00	0.97	0.98	31
25	Ta_Marbuta	0.96	1.00	0.98	26
26	Tha	0.88	1.00	0.94	23
27	Tsa	0.97	0.97	0.97	31
28	Wau	1.00	1.00	1.00	25
29	Ya	1.00	0.96	0.98	27
30	Zay	1.00	1.00	1.00	20
31	Zha	0.95	0.87	0.91	23
	Overall Accuracy			**0.97**	**786**
	Macro Average	0.97	0.97	0.97	786
	Weighted Average	0.97	0.97	0.97	786

Vol. 6, No. 4, Agustus 2025, Page. 2602-2624
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.4.4953

The performance evaluation metrics, namely Accuracy, Precision, Recall, and F1-Score, were computed as described in Equations (1)–(4) in the Method section. To illustrate the calculation, consider the *Dal* class in the VGG16 confusion matrix (Figure 6). The classifier correctly predicted 20 samples as *Dal* (TP = 20), misclassified 3 samples of *Dal* into other classes (FN = 3), and made no incorrect predictions of other classes as *Dal* (FP = 0). Based on Equations (2)–(4), the precision for *Dal* is $\frac{20}{20+0} = 1.00$, the recall is $\frac{20}{20+3} \approx 0.87$, and the F1-score is $2 * \left(\frac{1.00*0.87}{1.00+0.87}\right) \approx 0.93$. These results match the values reported in Table 3.

Despite the overall strong performance, the classification report also highlights certain classes where the model's predictive ability is relatively lower. Notably, the *Dal* class exhibits a recall of 0.87, resulting in an F1-score of 0.93. This indicates that while the model predicts *Dal* correctly in most cases, it fails to correctly identify approximately 13% of *Dal* instances, often confusing them with phonetically and visually similar classes such as *Dzal* or *Zha*. This trend aligns with the confusion matrix analysis discussed previously, which indicates systematic challenges in distinguishing between these closely related gestures.

Similarly, the *Zha* class presents a comparable challenge, with a recall of 0.87 and an F1-score of 0.91, signifying that the model tends to miss a small but consistent proportion of *Zha* instances. Given that *Dal* and *Zha* both share phonological and gestural similarities in Arabic sign articulation, these findings suggest that the model's feature representations may not yet be sufficiently sensitive to the subtle differences that distinguish these classes. This limitation could be attributed to factors such as limited variation in the dataset or insufficient gesture diversity for these particular signs.

The *Ghain* class also exhibits slightly reduced performance, with both precision and recall at 0.91, leading to an F1-score of 0.91. This pattern of errors likely stems from confusion with the *Ain* class, as both gestures may involve visually similar hand configurations associated with pharyngeal sounds in spoken Arabic. Such errors point to the inherent difficulty in disambiguating signs that are similar not only in visual appearance but also in their phonetic articulation roots, especially in a static image recognition context that lacks temporal cues.

The *Jim* class further illustrates this challenge, achieving an F1-score of 0.90, the lowest among all classes. The confusion matrix suggests that *Jim* is often misclassified as *Kaf*, likely due to overlapping hand shapes or orientations. While these errors are not widespread, they are consistent and indicate that specific gesture pairs require more refined feature extraction, potentially beyond what is provided by a VGG16-based static image model alone.

An interesting observation arises from the analysis of the *Tha* class. This class achieves a recall of 1.00, meaning that all *Tha* instances are correctly identified, yet it records a precision of 0.88, resulting in an F1-score of 0.94. This imbalance suggests that while the model is susceptible to identifying *Tha*, it suffers from false positives, namely mistakenly classifying other gestures as *Tha*. Such a pattern indicates overprediction towards this class, potentially due to overlapping features with other signs or a bias introduced by data imbalance during training.

From a macro-level perspective, the equality between the macro average and weighted average metrics (both precision, recall, and F1-score at 0.97) demonstrates that the model effectively handles class imbalance. This balance indicates that the classifier does not disproportionately favor classes with higher support (i.e., number of instances) over those with fewer samples. It reflects the effectiveness of stratified dataset splitting, as well as the impact of augmentation techniques in mitigating class imbalance challenges during training.

The classification report, as detailed in Table 2, corroborates the findings from the confusion matrix and class-wise accuracy analysis. It confirms that the VGG16-based model performs exceptionally well across the majority of Arabic alphabet sign gestures, with most classes achieving

Vol. 6, No. 4, Agustus 2025, Page. 2602-2624
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.4.4953

near-perfect performance. The few courses with relatively lower precision or recall are consistently those that exhibit high visual or phonetic similarity, suggesting that future improvements could focus on enhancing feature granularity, employing multimodal data (e.g., depth or temporal sequences), or adopting advanced neural architectures with attention mechanisms to capture fine-grained distinctions better. These enhancements would be instrumental in further advancing the reliability of ArSL recognition systems for practical applications, particularly in enabling Qur'anic Tadarus accessibility for the deaf and hard-of-hearing community.

4. DISCUSSIONS

The experimental results underscore the robustness of the VGG16-based deep learning approach for Arabic Alphabet Sign Language (ArSL) recognition, particularly in enhancing Qur'anic Tadarus accessibility for the deaf and hard-of-hearing community. Achieving an overall test accuracy of 97.07%, alongside consistently high precision, recall, and F1-scores for most classes, the model demonstrates strong generalization capabilities. This performance validates the effectiveness of employing transfer learning with the VGG16 architecture, where freezing early convolutional layers while fine-tuning deeper layers enables the model to leverage pretrained general visual features while adapting to the specific characteristics of ArSL gestures.

The analysis of learning dynamics, as visualized in the learning curves, indicates stable convergence without overfitting. Both training and validation losses decline steeply during the initial epochs and stabilize in later stages, while accuracy consistently improves. This trajectory reflects effective learning facilitated by data augmentation, an appropriate learning rate schedule, and a two-phase fine-tuning strategy. The absence of divergence between training and validation metrics indicates the model's ability to strike a balance between memorizing training data and generalizing to unseen instances.

To place these findings in context, we compare the proposed VGG16 model with our previously reported ResNet-18 baseline trained and evaluated under the same 31-class ArSLA protocol. The ResNet-18 model achieved a peak validation accuracy of 98.09% and an overall accuracy of 98%, with macro- and weighted-averaged precision, recall, and F1-scores of 0.98. In comparison, the VGG16 model in this study attained a peak validation accuracy of 97.33% and stabilized at 97.07% by the final epoch, with macro- and weighted-averaged precision, recall, and F1-scores of 0.97. These results indicate a modest absolute gain (≈1%) for ResNet-18 and slightly faster convergence, while VGG16 remains competitive with a simpler backbone—an attractive trade-off for deployment on resource-constrained devices. (See Figure 4(a)–(b) for VGG16 curves and Figure 5(a)–(b) for ResNet-18; optional summary in Table 2)

Further insights emerge from the confusion matrix, which reveals a generally high classification accuracy with certain exceptions. Specific classes such as *Dal*, *Zha*, *Ghain*, and *Jim* exhibit comparatively higher misclassification rates. This pattern suggests inherent visual ambiguities between certain signs, where subtle differences in hand posture, finger articulation, or orientation challenge the model's ability to differentiate them using static images alone. Notably, confusion often occurs between phonologically or visually related pairs, such as *Dal* and *Zha*, reflecting intrinsic challenges in gesture recognition tasks.

A cross-model inspection of confusion matrices further clarifies the performance gap. Both models exhibit strong diagonal dominance, but ResNet-18 reduces several misclassifications on difficult pairs (e.g., Dal vs. Zha; Ghain vs. Ain), aligning with its higher aggregate scores. At the same time, overlapping error patterns across both backbones suggest that some ambiguities are intrinsic to static imagery and may ultimately require temporal or multimodal cues to resolve reliably. (Refer to Figure 6 for the VGG16 confusion matrix and Figure 7 for ResNet-18.)

Vol. 6, No. 4, Agustus 2025, Page. 2602-2624
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.4.4953

The classification report substantiates these findings, showing strong overall macro and weighted F1-scores of 0.97, yet revealing marginally lower scores for classes with overlapping visual features. For instance, *Jim* exhibits an F1-score of 0.90, while *Zha* scores 0.91, indicating persistent classification difficulties despite overall high performance. These results align with broader challenges reported in sign language recognition research, where gestures differentiated by minor spatial variations often require more sophisticated spatial representation capabilities than standard CNNs provide.

Methodologically, three design choices appear central to the observed stability and accuracy: (i) a two-phase fine-tuning regimen (freezing early layers before unfreezing the full VGG16 network) enabling rapid adaptation followed by task-specific refinement; (ii) an adaptive learning-rate scheduler (ReduceLROnPlateau) that sustains progress after validation-loss plateaus; and (iii) rich data augmentation (rotation, cropping, flipping, perspective transformation, Gaussian blur, and color jitter) to improve invariance and mitigate overfitting. Together with the relatively balanced per-class distribution, these choices explain the tight alignment between training and validation metrics and the strong terminal performance.

A key contributing factor to these limitations lies in the static nature of the dataset. Despite extensive augmentation to simulate variability, the dataset lacks temporal dynamics, signer diversity, and contextual transitions between gestures. In practical applications, users present signs dynamically, with variations in speed, orientation, and signer-specific styles. Consequently, while the model performs excellently on isolated static gestures, its adaptability to continuous or natural signing remains an open question, highlighting a critical avenue for further exploration.

Additionally, while VGG16 serves as an effective backbone for this task, it is computationally intensive compared to modern architectures, such as MobileNet, EfficientNet, or Vision Transformers (ViT). Exploring these architectures could yield models that offer comparable or improved accuracy with significantly lower computational demands, enabling real-time deployment on mobile devices or embedded systems, which is a crucial consideration for assistive technologies intended for daily use in educational or religious contexts.

An alternative strategy for overcoming current limitations involves incorporating temporal modeling. Techniques such as 3D Convolutional Neural Networks (3D-CNN), CNN-LSTM hybrids, or Transformer-based video models can capture temporal dependencies inherent in sign language, allowing the system to discern gesture transitions and motion-based cues. Implementing such models would extend the current system from static alphabet recognition toward more comprehensive dynamic word or phrase-level recognition, significantly broadening its real-world applicability.

While the proposed VGG16-based ArSL classifier demonstrates high effectiveness for static gesture recognition, the analysis highlights critical areas for future enhancement. Addressing challenges related to gesture ambiguity, signer variability, and temporal dynamics will be essential for transitioning from isolated alphabet recognition toward robust, real-world-ready sign language communication systems. Such advancements are pivotal for fostering inclusivity in Qur'anic learning and broader accessibility initiatives for the deaf and hard-of-hearing community.

5. LIMITATION AND FUTURE WORKS

Despite the strong performance achieved by the VGG16-based Arabic Alphabet Sign Language (ArSL) classifier, several limitations must be acknowledged. First, the model is designed to recognize static hand gestures rather than dynamic sign sequences. Sign languages, including ArSL, inherently involve temporal components, such as movement trajectories, speed variations, and transitions between gestures [42], [43]. The reliance on static image input restricts the system's ability to capture these dynamic features, limiting its applicability to isolated alphabet recognition rather than full-word or sentence-level communication.

Vol. 6, No. 4, Agustus 2025, Page. 2602-2624
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.4.4953

Another significant limitation stems from the dataset's scope and diversity. Although the dataset effectively represents 31 Arabic alphabet signs, it is composed of controlled, static images with limited variation in signer demographics, backgrounds, and lighting conditions. This constraint raises concerns about the model's generalizability to real-world environments, where users present diverse hand shapes, skin tones, camera angles, and occlusions. The absence of signer diversity could lead to biased performance when deployed across broader user populations.

Furthermore, the confusion observed between specific gesture pairs, such as *Dal* and *Zha* or *Ghain* and *Ain*, underscores the inherent challenges of distinguishing visually similar signs in a static context. This issue suggests that the current architecture may not sufficiently capture fine-grained spatial nuances or contextual cues required for disambiguation. While data augmentation mitigates some variability, it cannot fully replicate the complexity of real-world gesture articulation.

From a computational perspective, the use of VGG16, while effective, involves considerable computational overhead. The architecture contains a high number of parameters, resulting in longer training times and higher memory requirements. Although suitable for research and proof-of-concept development, this complexity may hinder deployment on edge devices or mobile platforms commonly used by the deaf and hard-of-hearing community for accessibility tools.

Addressing these limitations presents several promising avenues for future work. First, integrating temporal modeling techniques, such as 3D Convolutional Neural Networks (3D-CNN), Convolutional LSTM (ConvLSTM), or Transformer-based video models, would enable the system to capture the motion dynamics and temporal dependencies inherent in sign language communication. Such models could transition the classifier from static alphabet recognition toward continuous sign recognition, significantly expanding its practical utility.

Second, expanding the dataset to include a broader range of signers, environments, and gesture variations is essential for enhancing the model's robustness. Crowdsourcing data from diverse users or collaborating with sign language communities can help create a more comprehensive and representative dataset. Additionally, introducing multi-angle recordings or depth information could help address issues related to gesture ambiguity and occlusion.

Another direction involves exploring lightweight yet robust architectures such as MobileNetV3, EfficientNetV2, or Vision Transformers (ViTs). These models offer a favorable balance between computational efficiency and classification accuracy, making them suitable for deployment on portable devices. Implementing such models would not only reduce computational costs but also facilitate real-time inference, a critical requirement for assistive technologies.

The system could be further enhanced by adopting multimodal learning strategies. Integrating visual gesture recognition with additional modalities, such as hand skeletal tracking, depth sensing, or electromyography (EMG) signals, has the potential to substantially improve classification accuracy, especially when dealing with ambiguous gestures or partial occlusions. Multimodal approaches offer greater robustness against environmental variability and differences among signers, making the system more adaptable to real-world conditions.

The current VGG16-based ArSL classifier demonstrates strong performance in recognizing static representations of Arabic alphabet signs. However, expanding its capabilities to handle dynamic, continuous sign communication remains an essential challenge. Overcoming the limitations identified in this research—through improvements in dataset diversity, model architecture, and the incorporation of temporal or multimodal learning techniques—will be crucial for advancing the system toward more comprehensive accessibility solutions. This progress is particularly significant for enhancing Qur'anic Tadarus experiences and promoting educational inclusivity for deaf and hard-of-hearing communities.

Vol. 6, No. 4, Agustus 2025, Page. 2602-2624
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.4.4953

6. CONCLUSION

This study presents the development and evaluation of a VGG16-based deep learning model for Arabic Alphabet Sign Language (ArSL) recognition, designed to enhance accessibility to Qur'anic Tadarus for the deaf and hard-of-hearing community. Leveraging transfer learning with a pretrained VGG16 architecture, the proposed model achieves a high level of accuracy, reaching 97.07% on the test dataset. The results demonstrate strong generalization capabilities, with high precision, recall, and F1-scores across the majority of the 31 Arabic alphabet classes.

The findings indicate that the combination of effective data preprocessing, data augmentation, and a fine-tuning strategy enables the model to distinguish complex hand gestures in static image settings accurately. Analysis of the confusion matrix and classification report reveals that while the model performs exceptionally well overall, specific gesture pairs—particularly those with subtle visual similarities—remain prone to misclassification. These challenges align with the known limitations in static image-based sign language recognition. For context, compared with our ResNet-18 baseline trained under the same 31-class protocol, the proposed VGG16 model remains competitive despite a modest (~1%) gap in peak validation accuracy, offering a simpler backbone that is attractive for resource-constrained deployments.

Despite these achievements, the study also identifies several critical limitations. The reliance on static image inputs restricts the model's ability to capture the dynamic nature of real-world sign language communication. Additionally, the dataset's limited diversity in terms of signer demographics and environmental conditions poses challenges to broader generalizability. The computational demands of the VGG16 architecture, although manageable for research purposes, may limit its practicality for deployment on resource-constrained devices.

Future research directions should focus on addressing these limitations by incorporating temporal modeling techniques, expanding the dataset with more diverse and realistic samples, and exploring more efficient model architectures such as MobileNet, EfficientNet, or Transformer-based approaches. Furthermore, integrating multimodal data—combining visual inputs with depth information, skeletal tracking, or electromyography (EMG)—offers a promising approach to enhance robustness, particularly in handling ambiguous or occluded signs.

The results of this research contribute a meaningful advancement toward the development of accessible sign language recognition systems tailored for Qur'anic Tadarus and broader educational contexts. By continuing to refine the model and expand its capabilities, this work has the potential to significantly improve communication accessibility and foster greater inclusion for the deaf and hard-of-hearing community within religious and educational settings.

CONFLICT OF INTEREST

The authors declare that there is no conflict of interest between the authors or with the research object in this paper.

ACKNOWLEDGEMENT

The authors would like to express their sincere gratitude to Prof. Muchlas, Rector of Universitas Ahmad Dahlan; Prof. Sofyan Anif, former Rector of Universitas Muhammadiyah Surakarta; and Prof. Harun Joko Prayitno, the current Rector of Universitas Muhammadiyah Surakarta, for their continuous encouragement and support. Deep appreciation is also extended to Prof. Abdul Fadlil, Prof. Imam Riadi, Prof. Tole Sutikno, and all faculty members of the Doctoral Program in Informatics at Universitas Ahmad Dahlan for their valuable guidance, academic mentorship, and unwavering support throughout this research.

Vol. 6, No. 4, Agustus 2025, Page. 2602-2624
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.4.4953

REFERENCES

[1] S. Baghavathi Priya, P. V. R. S. Rao, and T. S. Madeswaran, "Sign to Speak: Real-time Recognition for Enhance Communication," in *Proceedings of the 3rd International Conference on Applied Artificial Intelligence and Computing, ICAAIC 2024*, Institute of Electrical and Electronics Engineers Inc., 2024, pp. 871–876. doi: 10.1109/ICAAIC60222.2024.10575697.

[2] D. Bisht, M. Kojage, M. Shukla, Y. Prakash Patil, and P. Bagade, "Smart Communication System Using Sign Language Interpretation," in *THE 31ST Conference of Fruct Asssociation*, 2025. doi: 10.23919/FRUCT54823.2022.9770914.

[3] M. H. Ismail, S. A. Dawwd, and F. H. Ali, "Static hand gesture recognition of Arabic sign language by using deep CNNs," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 24, no. 1, pp. 178–188, Oct. 2021, doi: 10.11591/ijeecs.v24.i1.pp178-188.

[4] V. Kandukuri, S. R. Gundedi, V. Kamble, and V. Satpute, "Deaf and Mute Sign Language Translator on Static Alphabets Gestures using MobileNet," in *2023 2nd International Conference on Paradigm Shifts in Communications Embedded Systems, Machine Learning and Signal Processing, PCEMS 2023*, Institute of Electrical and Electronics Engineers Inc., 2023. doi: 10.1109/PCEMS58491.2023.10136074.

[5] P. Jayadharshini, L. Krishnasamy, S. Santhiya, K. U. Harsine, P. Geetha, and B. Logu, "Enhanced Accessibility for Recognizing Indian and American Sign Language through Deep Learning Techniques," in *2024 IEEE 4th International Conference on ICT in Business Industry and Government, ICTBIG 2024*, Institute of Electrical and Electronics Engineers Inc., 2024. doi: 10.1109/ICTBIG64922.2024.10911064.

[6] A. N. Handayani, M. I. Akbar, H. Ar-Rosyid, M. Ilham, R. A. Asmara, and O. Fukuda, "Design of SIBI Sign Language Recognition Using Artificial Neural Network Backpropagation," in *2022 2nd International Conference on Intelligent Cybernetics Technology & Applications (ICICyTA)*, 2022, pp. 192–197. doi: 10.1109/ICICyTA57421.2022.10038205.

[7] A. Deshpande, A. Shriwas, V. Deshmukh, and S. Kale, "Sign Language Recognition System using CNN," in *Proceedings of the International Conference on Intelligent and Innovative Technologies in Computing, Electrical and Electronics, ICIITCEE 2023*, Institute of Electrical and Electronics Engineers Inc., 2023, pp. 906–911. doi: 10.1109/IITCEE57236.2023.10091051.

[8] C. Author, A.-S. Khaled, and P. Wanda, "How to Stepping up Characters Recognition using CNN Algorithm?," *International Journal of Informatics and Computation (IJICOM)*, vol. 4, no. 2, 2022, doi: 10.35842/ijicom.

[9] U. Fadlilah, A. K. Mahamad, and B. Handaga, "The Development of Android for Indonesian Sign Language Using Tensorflow Lite and CNN: An Initial Study," in *Journal of Physics: Conference Series*, IOP Publishing Ltd, Apr. 2021. doi: 10.1088/1742-6596/1858/1/012085.

[10] A. Rakhmadi and A. Yudhana, "A Study of Worldwide Patterns in Alphabet Sign Language Recognition Using Convolutional and Recurrent Neural Networks," *Jurnal Teknik Informatika (JUTIF)*, vol. 6, no. 1, pp. 187–204, Feb. 2025, doi: 10.52436/1.jutif.2025.6.1.4202.

[11] A. Khan *et al.*, "Deep Learning Approaches for Continuous Sign Language Recognition: A Comprehensive Review," 2025, *Institute of Electrical and Electronics Engineers Inc.* doi: 10.1109/ACCESS.2025.3554046.

[12] B. Alsharif, A. S. Altaher, A. Altaher, M. Ilyas, and E. Alalwany, "Deep Learning Technology to Recognize American Sign Language Alphabet," *Sensors*, vol. 23, no. 18, Sep. 2023, doi: 10.3390/s23187970.

[13] P. Adla, P. Pola, P. S. Kiran, Sh. Kumar, and Pitchai R, "Lightweight American Sign Language and Gesture Recognition using YOLOv8," in *15th IEEE International Conference on Computational Intelligence and Communication Networks*, IEEE, 2023.

[14] B. Sundar and T. Bagyammal, "American Sign Language Recognition for Alphabets Using MediaPipe and LSTM," in *Procedia Computer Science*, Elsevier B.V., 2022, pp. 642–651. doi: 10.1016/j.procs.2022.12.066.

[15] E. A. Wijaya, S. B. Meliala, M. F. Hidayat, and I. A. Iswanto, "Sign Language Translator for SIBI," in *2024 6th International Conference on Cybernetics and Intelligent System, ICORIS 2024*, Institute of Electrical and Electronics Engineers Inc., 2024. doi: 10.1109/ICORIS63540.2024.10903922.

Vol. 6, No. 4, Agustus 2025, Page. 2602-2624
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.4.4953

[16] I. D. M. B. A. Darmawan *et al.*, "Advancing Total Communication in SIBI: A Proposed Conceptual Framework for Sign Language Translation," in *2023 International Conference on Smart-Green Technology in Electrical and Information Systems (ICSGTEIS)*, 2023, pp. 23–28. doi: 10.1109/ICSGTEIS60500.2023.10424020.

[17] M. Arthur Limantara and D. Tristianto, "SIBI Alphabet Detection System Based on Convolutional Neural Network (CNN) Method as Learning Media," *Internet of Things and Artificial Intelligence Journal*, vol. 4, no. 1, pp. 143–161, Mar. 2024, doi: 10.31763/iota.v4i1.716.

[18] U. Fadlilah *et al.*, "Android Application Prototype of Basic BISINDO Introduction and Practice for General People," Sep. 2022.

[19] M. B. Sasongko and N. Usman, "Analysis of The Quran Isyarat Learning Management at The Magelang Deaf Education Foundation," *Al Ulya*, vol. 10, no. 01, pp. 20–40, 2025, doi: 10.32665/alulya.v10i1.3559.

[20] B. Pamungkas, R. Wahab, and S. Suwarjo, "Teaching of the Quran and Hadiths Using Sign Language to Islamic Boarding School Students with Hearing Impairment," *International Journal of Learning, Teaching and Educational Research*, vol. 22, no. 5, pp. 227–242, May 2023, doi: 10.26803/ijlter.22.5.11.

[21] N. Irfan, P. Pakistan, and S. Seerat Hassan, "Perception of Teachers about Challenges Faced by Deaf Students in Learning Holy Quran (Muslim's Religious Book) Faisal Amjad," *Research Journal (MRJ)*, vol. 5, 2024.

[22] I. Dzulkifli, "Quranic Education for Deaf Students in Malaysia; Implementation and Challenges," *Nadwa: Jurnal Pendidikan Islam*, 2022, doi: 10.21580/nw.2022.16.1.10823.

[23] A. Alnuaim, M. Zakariah, W. A. Hatamleh, H. Tarazi, V. Tripathi, and E. T. Amoatey, "Human-Computer Interaction with Hand Gesture Recognition Using ResNet and MobileNet," *Comput Intell Neurosci*, vol. 2022, 2022, doi: 10.1155/2022/8777355.

[24] X. Han, F. Lu, and G. Tian, "Sign Language Recognition Based on Lightweight 3D MobileNet-v2 and Knowledge Distillation," in *ICETIS 2022*, Harbin, Jan. 2022.

[25] M. D. Meitantya, C. A. Sari, E. H. Rachmawanto, and R. R. Ali, "VGG-16 Architecture on CNN for American Sign Language Classification," *Jurnal Teknik Informatika (Jutif)*, vol. 5, no. 4, pp. 1165–1171, Jul. 2024, doi: 10.52436/1.jutif.2024.5.4.2160.

[26] A. M. J. Al Moustafa *et al.*, "Arabic Sign Language Recognition Systems: A Systematic Review," *Indian Journal of Computer Science and Engineering (IJCSE)*, vol. 15, 2024, doi: 10.21817/indjcse/2024/v15i1/241501008.

[27] M. Al-Barham, A. A. Sa'aleek, M. Al-Odat, G. Hamad, M. Al-Yaman, and A. Elnagar, "Arabic Sign Language Recognition Using Deep Learning Models," in *2022 13th International Conference on Information and Communication Systems, ICICS 2022*, Institute of Electrical and Electronics Engineers Inc., 2022, pp. 226–231. doi: 10.1109/ICICS55353.2022.9811162.

[28] M. Al-Barham *et al.*, "RGB Arabic Alphabets Sign Language Dataset," Jan. 2023, [Online]. Available: http://arxiv.org/abs/2301.11932

[29] A. M. Farouk *et al.*, "A New Approach for Arabic Sign Language Recognition (ArSLR)," in *NILES 2024 - 6th Novel Intelligent and Leading Emerging Sciences Conference, Proceedings*, Institute of Electrical and Electronics Engineers Inc., 2024, pp. 545–550. doi: 10.1109/NILES63360.2024.10753193.

[30] B. Y. Al-Khuraym and M. M. Ben Ismail, "Arabic Sign Language Recognition using Lightweight CNN-based Architecture," *International Journal of Advanced Computer Science and Applications*, vol. 13, no. 4, pp. 319–328, 2022, doi: 10.14569/IJACSA.2022.0130438.

[31] M. H. Ismail, S. A. Dawwd, and F. H. Ali, "Arabic Sign Language Detection Using Deep Learning Based Pose Estimation," in *Proceedings of 2021 2nd Information Technology to Enhance E-Learning and other Application Conference, IT-ELA 2021*, Institute of Electrical and Electronics Engineers Inc., 2021, pp. 161–166. doi: 10.1109/IT-ELA52201.2021.9773404.

[32] M. R. Kale, C. Kaur, I. Kumar, H. S. Mahdi, V. Uprikar, and V. Sankari, "Development of an Arabic Sign Language Recognition System Utilizing Deep Convolutional Neural Network," in *2025 5th International Conference on Advances in Electrical, Computing, Communication and*

Vol. 6, No. 4, Agustus 2025, Page. 2602-2624
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.4.4953

Sustainable Technologies, ICAECT 2025, Institute of Electrical and Electronics Engineers Inc., 2025. doi: 10.1109/ICAECT63952.2025.10958963.

[33] A. Mohan, D. Mohan, S. Vats, V. Sharma, and V. Kukreja, "Classification of Sign Language Gestures using CNN with Adam Optimizer," in *2024 2nd International Conference on Disruptive Technologies, ICDT 2024*, Institute of Electrical and Electronics Engineers Inc., 2024, pp. 430–433. doi: 10.1109/ICDT61202.2024.10489158.

[34] D. O. Melinte and L. Vladareanu, "Facial Expressions Recognition for Human–Robot Interaction Using Deep Convolutional Neural Networks with Rectified Adam Optimizer," *Sensors*, vol. 20, no. 8, 2020, doi: 10.3390/s20082393.

[35] O. G. Ajayi and J. Ashi, "Effect of varying training epochs of a Faster Region-Based Convolutional Neural Network on the Accuracy of an Automatic Weed Classification Scheme," *Smart Agricultural Technology*, vol. 3, Feb. 2023, doi: 10.1016/j.atech.2022.100128.

[36] S. G. S. Kumar and J. Abbass, "Enhancing Sign Language Communication: Advanced Gesture Recognition Models for Indian Sign Language," in *2025 International Research Conference on Smart Computing and Systems Engineering (SCSE)*, IEEE, Apr. 2025, pp. 1–6. doi: 10.1109/SCSE65633.2025.11031017.

[37] S. A. Nivishna Shree, D. Sasikala, and S. A. Nehaa Shree, "Performance Analysis of Sign Language Recognition using Deep Neural Network Architecture," in *Proceedings - 4th International Conference on Smart Technologies, Communication and Robotics 2025, STCR 2025*, Institute of Electrical and Electronics Engineers Inc., 2025. doi: 10.1109/STCR62650.2025.11020537.

[38] M.-U.-K. Rico, "Performance Analysis of CNN Model for Image Classification with Intel OpenVINO on CPU and GPU," Thesis, University of Windsor, Windsor, Ontario, Canada, 2023.

[39] S. Kurundkar, A. Joshi, A. Thaploo, S. Auti, and A. Awalgaonkar, "Real-Time Sign Language Detection," in *ViTECoN 2023 - 2nd IEEE International Conference on Vision Towards Emerging Trends in Communication and Networking Technologies, Proceedings*, Institute of Electrical and Electronics Engineers Inc., 2023. doi: 10.1109/ViTECoN58111.2023.10157784.

[40] P. A. Prabha, M. Daanish, N. Kumar, and N. Kumaar, "Deep-SignSpeak: Deep Learning based Sign Language Recognition and Regional Language Translation," in *2024 1st International Conference on Advanced Computing and Emerging Technologies, ACET 2024*, Institute of Electrical and Electronics Engineers Inc., 2024. doi: 10.1109/ACET61898.2024.10730664.

[41] V. K. Sambhav and R. Rajmohan, "Automated CNN model for Indian Sign Language Gesture Recognition," in *Proceedings of the 2024 10th International Conference on Communication and Signal Processing, ICCSP 2024*, Institute of Electrical and Electronics Engineers Inc., 2024, pp. 1483–1487. doi: 10.1109/ICCSP60870.2024.10544149.

[42] S. A. Shaba, S. I. Shroddha, M. J. Hossain, and M. F. Monir, "NeuralGesture Communication: Translating one Sign Language to Another Sign Language Using Deep Learning Model and gTTs," in *IEEE Vehicular Technology Conference*, Institute of Electrical and Electronics Engineers Inc., 2024. doi: 10.1109/VTC2024-Spring62846.2024.10683444.

[43] M. Soundarya, M. Yazhini, N. Thirumala Sree, N. Sornamalaya, and C. Vinitha, "Sign Language Recognition Using Machine Learning," in *Proceedings - 3rd International Conference on Advances in Computing, Communication and Applied Informatics, ACCAI 2024*, Institute of Electrical and Electronics Engineers Inc., 2024. doi: 10.1109/ACCAI61061.2024.10602025.