engineering proceedings

MDPI

Proceeding Paper

AI-Powered Sign Language Detection Using YOLO-v11 for Communication Equality †

Ivana Lucia Kharisma [1,*], Irma Nurmalasari [1], Yuni Lestari [1], Salma Dela Septiani [1], Kamdan [1] and Muchtar Ali Setyo Yudono [2]

[1] Informatics Engineering Department, Nusa Putra University, Sukabumi 43152, Indonesia; irma.nurmalasari_ti22@nusaputra.ac.id (I.N.); yuni.lestari_ti22@nusaputra.ac.id (Y.L.); salma.dela_ti22@nusaputra.ac.id (S.D.S.); kamdan@nusaputra.ac.id (K.)
[2] Electrical Engineering Department, Sultan Ageng Tirtayasa University, Serang 42182, Indonesia; muchtar.ali@untirta.ac.id
* Correspondence: ivana.lucia@nusaputra.ac.id
† Presented at the 7th International Global Conference Series on ICT Integration in Technical Education & Smart Society, Aizuwakamatsu City, Japan, 20–26 January 2025.

Abstract

Communication plays a vital role in conveying messages, expressing emotions, and sharing perceptions, becoming a fundamental aspect of human interaction with the environment. For individuals with hearing impairments, sign language serves as an essential communication tool, enabling interaction both within the deaf community and with non-deaf individuals. This study aims to bridge this misconception by developing an iconic language recognition system using the Deep Learning-based YOLO-v11 algorithm. YOLO-v11, a state-of-the-art object detection algorithm, is known for its speed, accuracy, and efficiency. The system uses image recognition to identify hand gestures in ASL and translates them into text or speech, facilitating inclusive communication. The accuracy of the training model is 94.67%, and the accuracy of the testing model is 93.02%, indicating that the model has excellent performance in recognizing sign language from the training and testing datasets. Additionally, the model is very reliable in recognizing the classes "Hello", "I Love You", "No", and "Thank You" with a sensitivity close to or equal to 100%. This research contributes to advancing communication equality for individuals with hearing impairments, promoting inclusivity, and supporting their integration into society.

Keywords: sign language; YOLO; communication; equality

check for updates

Academic Editors: Debopriyo Roy, George F. Fragulis and Peter Ilic

Published: 8 September 2025

Citation: Kharisma, I.L.; Nurmalasari, I.; Lestari, Y.; Septiani, S.D.; Kamdan; Yudono, M.A.S. AI-Powered Sign Language Detection Using YOLO-v11 for Communication Equality. *Eng. Proc.* **2025**, *107*, 83. https://doi.org/10.3390/engproc2025107083

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Communication plays an important role in sending messages, expressing feelings, and conveying perceptions, and it is one of the main ways humans interact with their environment, which involves capturing sounds and interpreting the language used by others who intend to communicate [1]. Sign language is basically a communication tool used by the deaf community that aims to facilitate communication, both among the deaf and with individuals who are not deaf. Sign language is expressed through hand gestures, facial expressions, and body movements [2]. In addition, the use of sign language also helps increase the inclusion and involvement of students with disabilities in the general school environment [3]. Normal hearing plays an important role in the acquisition and production of spoken language because it allows children to be part of the spoken language around them. However, for individuals with hearing disabilities, the challenge is to develop an

important means of supporting communication [4]. Any gesture or visual language that uses certain hand, arm, and finger shapes and movements, along with eye, face, head, and body movements, is called sign language [5]. People with disabilities are a group of people who have limitations that prevent them from participating and playing a role in daily life. In 2010, the Central Statistics Agency (BPS) reported that as many as 3,024,271 Indonesians, out of 191,709,144 people, had physical disabilities, including hearing and speech disorders [3].

This study aims to help individuals interact and share information with each other without communication gaps, and to develop a sign language system model to make it easier to communicate with people with disabilities using the Deep Learning method with the YOLO (You Only Look Once) algorithm. In this study, we also decided to use ASL in the form of symbols or speech and in the form of words or sentences [6]. The method used in this study is the YOLO-v11 algorithm. YOLO is a real-time object detection algorithm developed by Joseph Redmon and Ali Farhadi in 2015 [7]. In their research, due to its simple architecture, YOLO is also said to be very fast in identifying objects, and the average accuracy obtained reached 88% in the ImageNet 2012 validation [8]

YOLO-v11 is the latest and greatest (SOTA) model that builds on the success of the previous YOLO version and introduces new features and enhancements to further improve performance and flexibility. YOLO-v11 is designed to be fast, accurate, and easy to use, making it an excellent choice for a variety of object detection and tracking tasks, instance segmentation, image classification, and pose estimation [9]. The YOLO-v11 algorithm is included in image detection that uses convolutional neural networks (CNNs) to predict bounding boxes and object class probabilities in input images at 45 FPS (2 milliseconds/image) [10]. In this study, the data used is a manually created ASL dataset containing 4000 images forming hand gestures in various positions [11].

2. Related Works

ASL is a sign language used by people with disabilities to communicate; therefore, the study entitled "A Comprehensive Application for Sign Language Alphabet and World Recognition, Text-to-Action Conversion for Learners, Multi-Language Support and Integrated Voice Output Functionality" aims to introduce a comprehensive application designed to facilitate learning in communicating for sign language users [12]. In another study, a machine translation system was also applied, which aimed to convert spoken Turkish into Turkish Sign Language (TID) [13]. Another study also used a semantic analysis algorithm for simple sentences that were developed and introduced, aimed at translating Russian text into Russian sign language based on a comparison of the proposed syntactic structures [14].

Several studies have used convolutional neural networks (CNNs) and computer vision algorithms to translate ASL into text and speech in local languages, such as in a study in Nepal that achieved over 99% accuracy [15]. Furthermore, other studies have shown that sign languages have dialectal variations that make automatic recognition difficult, so methods such as 3D convolutional networks and skeleton-based recognition have been used. The sign language transformer model excelled with a BLEU-4 score of 21.80, more than double that of the previous model (9.58) [16]. Another study using R-CNN, 3D CNN, and LSTM achieved 99% accuracy in recognizing the sign language vocabulary, while CorrNet reduced the Word Error Rate to 18.8% on the training set [17]. Translation in PHEONIX-Weather-2014T achieved a BLEU-4 of 24.32, lagging behind English–German (30.9) [18]. The TGCN pose-based model achieved an accuracy of 62.63% for 2000 words [19]. Another proposal leverages a lightweight 3D convolution module; exploratory results illustrate the performance of RealTimeSignNet on standard sign dialect datasets, achieving

an accuracy of 88.1 on the wide continuous sign dialect dataset (continuous SLR), 98.2% on the discontinuous sign dialect dataset (500 SLR), and 91.50% on the English Sign Dialect dataset (WLAS) [20].

On the other hand, the study proposes a novel multi-lingual multi-modal SLR framework, MLMSign, achieving high precision ( ) on six benchmark datasets (i.e., Massey, Inactive ASL, NUS II, TSL Fingerspelling, BdSL36v1, and PSL) [21]. This study addresses the computational complexity associated with sign language recognition (SLR) methods. The proposed method operates using (ResGCN). The method is tested on five challenging SLR datasets—WLASL-100, WLASL-300, WLASL-1000, LSA-64, and MINDS-Libras—and achieves impressive accuracies of 83.33%, 72.90%, 64.92%, 100%, and 96.70%, respectively [22].

By utilizing the ADDSL dataset, with the YOLOv5 method, a demonstration of the single-stage question locator (YOLOv5) produces an average induction time of 9.02 ms per image and a best accuracy of 92% [23]. In another study using YOLOv2, Vitis AI, and FINN for sign dialect recognition on a Field Programmable Entryway Cluster (FPGA), a significant average accuracy (mAP) score of 61.2% was achieved on the Indian Sign Dialect (ISL) Hindi dataset [24]. The YOLOv8 method has been applied for recognizing and parsing sign dialect signals in real-time, with the top recognition precision of the proposed approach being 99.4% on the AASL dataset [25]. Another thought is particularly focused on the YOLO-v9 event, which was released in 2024. Overall, although both models have great capabilities in real-time hand gesture recognition, YOLO-v9e is dominant in terms of precision and classification. However, if discovery speed can be a primary requirement, then YOLO-v9c may be a higher choice. Both models can accurately identify all 26 ASL letters, illustrating their capabilities in hand gesture recognition applications [26].

In this study, using image recognition technology or pattern recognition, the system can identify hand gestures and translate them into text or voice that can be understood by individuals who cannot communicate using sign language. This study uses the latest version of the YOLO method, namely YOLO-v11. YOLO-v11 is the latest and greatest (SOTA) model that builds on the success of previous YOLO versions and introduces new features and enhancements to further improve performance and flexibility. YOLO-v11 is designed to be fast, accurate, and easy to use, making it an excellent choice for a variety of object detection and tracking tasks, instance segmentation, image classification, and pose estimation. The YOLO architecture is heavily influenced by the GoogleNet backbone, with a neural network consisting of 24 convolutional layers to perform feature extraction, followed by 2 FCNs (Fully Connected Layers) to make bounding box coordinate predictions and object class classification. Architecture of YOLO illustrated in Figure 1.

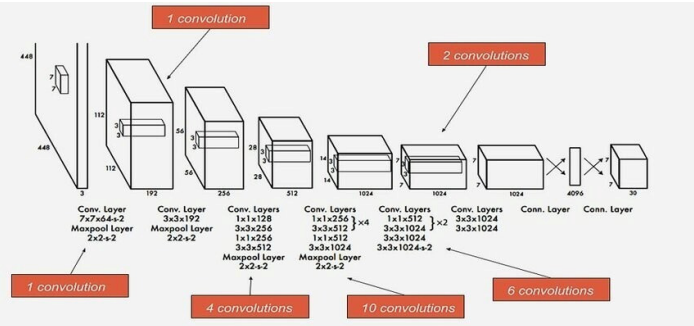


Figure 1. YOLO architecture [27].

Sign language is the main topic that will be used as the basic material for the image dataset. The image dataset for ASL itself has three categories, namely: letters/alphabet,

numbers, and symbols (per word), illustrated in Figure 2. This study uses a dataset in the form of symbols, words, or sentences.



Figure 2. ASL symbols [28].

3. Materials and Methods

The research wants to develop a sign language detection system using YOLO-v11, which is generally widely known as one of the most effective approaches in object detection, because of its ability to process images quickly and efficiently. The process in this research itself is data collection, pre-processing, model training, as well as evaluation and testing, as shown in Figure 3.

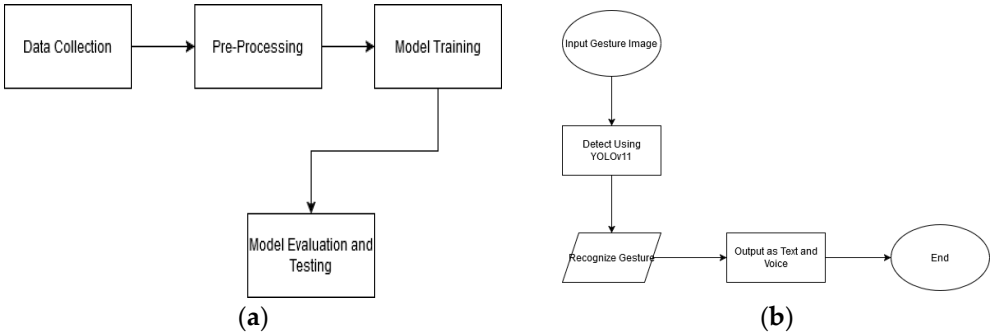


Figure 3. (**a**) Process flow in research, (**b**) system in sign language detection.

The first image, or Figure 3a, provides an overview of the model training process, from data collection to the training phase, where the model learns to recognize patterns from the given data. This process involves data pre-processing, dataset partitioning, and the use of machine learning algorithms to optimize the model.

Meanwhile, the second image, or Figure 3b, focuses more on the implementation of the trained model. Once the training process is complete, the model can be used to detect and recognize gestures in real-time. This is performed using video input or images taken from the camera, where the model classifies and processes the existing gestures directly. In other words, the second image shows the practical application of the trained model in the context of real-world use, namely, live and interactive gesture recognition.

3.1. Data Collection

This research consists of four classes of image objects: “Hello”, “Thank You”, “No”, “Yes”, and “I Love You”, where the dataset consists of 4000 images, consisting of 800 images per class. The following shows images of objects for each class, example given in Figure 4.



(**a**) (**b**) (**c**) (**d**) (**e**)

Figure 4. (**a**) "Hello", (**b**) "Thank You", (**c**) "No", (**d**) "Yes", and (**e**) I Love You.

3.1.1. Pre-Processing

Labeling

The process begins with labeling the dataset to provide class information so that the model can properly learn patterns during training. Labeling is performed by marking each sample, such as a hand gesture, with the appropriate label, either manually or with semi-automatic tools. Accurate labeling is essential to ensure the model receives the correct information, allowing for proper detection between the input features and the desired output. This process needs to be consistent and of high quality to avoid errors and biases in the model, especially in complex or overlapping gestures.

Bounding Box

In the context of image object detection, the bounding box provides coordinates for the location of the image object. For example, in an image showing hands forming the word "Hello" in sign language, the bounding box highlights the area where the hand is located. In addition to location, bounding boxes indicate the relative sizes of objects in the image. This information is important to help the model recognize objects even if their sizes are different in different images.

In Figure 5, we can see the results of the labeling and bounding box process, where each object in the image is labeled according to the relevant category, such as the ASL handgrip gesture meaning "Yes" is labeled "Yes". The bounding box, a box that surrounds the object, serves to mark the object's location specifically, helping the model focus on a specific area, such as the hand or face. This process is important to allow the model to detect and distinguish objects in various positions and sizes, improving detection accuracy.

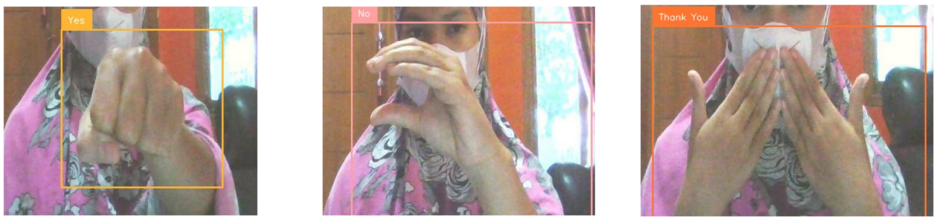


Figure 5. Result of image labeling and bounding box.

3.2. Model Training

In this study, the dataset was divided into 75% for training, 15% for validation, and 10% for testing so that the 4000 images, consisting of 800 images per class, were divided into 3019 images for training, 583 for validation, and 401 for testing, all in a 640 × 640 pixel resolution. The YOLO-v11 model was trained with a batch size of 16, 50 epochs, and a speed of 0.2 ms for pre-processing, 2.4 ms for inference, 0.0 ms for loss, and 2.3 ms for post-processing per image; the model experienced quite a significant increase in performance during the first few epochs. The model trained in Google Colab with a NVIDIA GeForce RTX 3080 GPU took about 0.936 h. Very good epochs and significant improvements can be seen between epochs 40–50, where the model shows steady improvements in almost all metrics (box loss, class loss, DFL loss, precision, recall, and mAP). Epoch 50 is the

peak that recorded a very high mAP50 value (0.994) and a good mAP50-95 value (0.731), indicating that the model has achieved optimal performance in detecting objects. The results summarized in Table 1.

Table 1. This is a table of epoch results that have increased during the process.

Epoch	mAP@50	mAP@50-95
40th	0.993	0.707
41th	0.994	0.718
42th	0.994	0.717
43th	0.994	0.724
44th	0.994	0.725
45th	0.994	0.727
46th	0.994	0.725
47th	0.994	0.733
48th	0.994	0.732
49th	0.994	0.731
50th	0.994	0.731

One of the main components that supports the success of this training is the selection of the Adam optimization algorithm (Adaptive Moment Estimation). The Adam optimization algorithm is designed to speed up the learning process in neural networks. Adam adaptively sets the learning rate for each parameter in the model by using two moment estimates.

3.3. Model Evaluation and Testing

From the graphs above, it can be concluded that the machine learning model being evaluated shows quite good performance. We can conclude from training process, illustrated in Figure 6a, the loss value that continues to decrease indicates that the model is getting better at learning from training data. In addition, high precision and recall values indicate that the model is able to detect objects with good accuracy, both in terms of location and classification. The orange line is a smoothed result of the blue line, to make it easier to understand the main trends of the training progress.

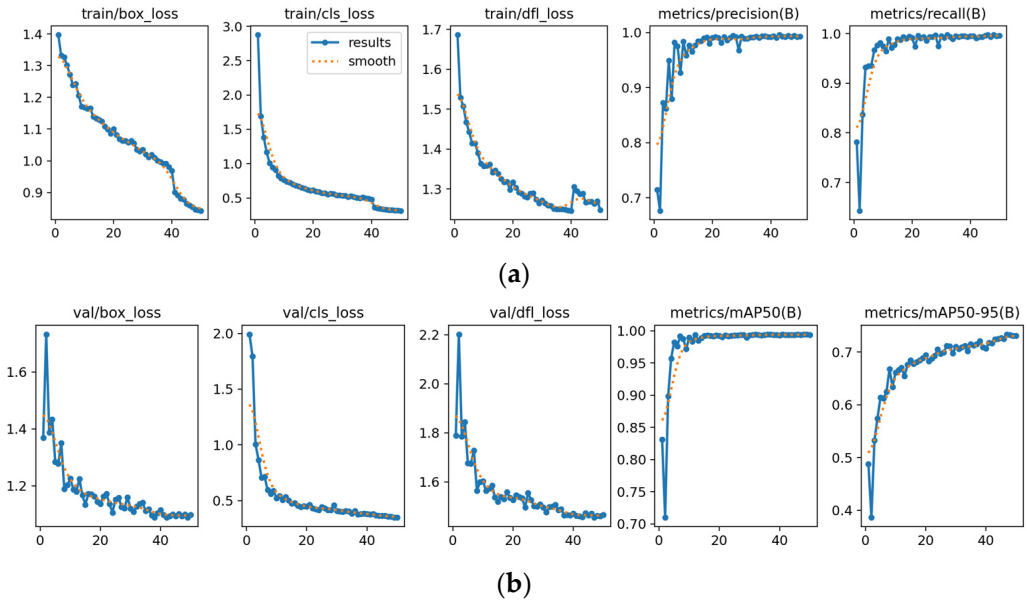


Figure 6. (**a**) Result of model evaluation and testing; (**b**) result of model evaluation and testing.

The graphs in Figure 6b provide a good overview of the performance of the object detection model during the validation process. It can be seen that the high mAP value also indicates that the model is able to detect objects with good accuracy. On the other hand, fluctuations in the loss and metric values on the validation data indicate that the training process may not be completely stable The orange line is a smoothed result of the blue line, to make it easier to understand the main trends of the validation progress.

4. Result and Discussion

The results of the testing dataset use the Google Colab training process. The total number of datasets is 4000 images divided into five classes: "Hello", "Thank You", "No", "Yes", and "I Love You", with 3019 images, including 583 for validation and 401 for testing. The image size is 640 × 640 pixels, and the testing was performed using 50 epochs, shown in Figure 7.

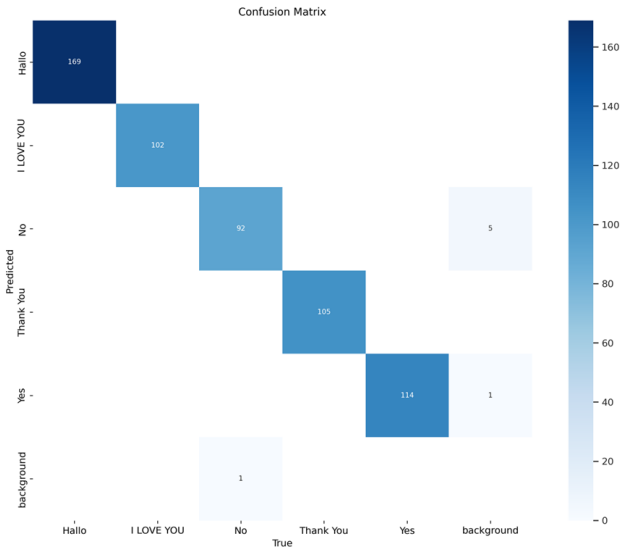


Figure 7. Confusion Matrix.

The Confusion Matrix above shows that most of the predictions are on the main diagonal, which means the model has a high level of accuracy. This indicates that the model correctly identifies the majority of classes, with relatively few misclassifications. However, it can be seen that the model still often misrecognizes "Background" or mixes it with other classes, particularly in cases where the background is visually similar to certain objects or gestures. This suggests that the model may have difficulty distinguishing between the background and objects with similar features, which could be a result of insufficient training data or a lack of clarity in some instances of the background.

As seen from Table 2, the model has high precision and recall (P: 99.2%, R: 99.3%) with excellent average detection performance (mAP50) (99.4%). The mAP50-95 result is at 73.3%, indicating adequate performance, but it can be improved for a tighter IoU threshold. While viewed individually or per class, it is seen that the "Thank You" class has the best performance on the mAP50-95 metric (77.6%), while the "Yes" class has the lowest performance (68.5%). However, precision and recall are very good for all classes, indicating the model can detect objects consistently. In terms of model efficiency, with an average inference time of 10ms per image, this model is very fast and suitable for real-time applications.

Table 2. Result of YOLO model performance evaluation.

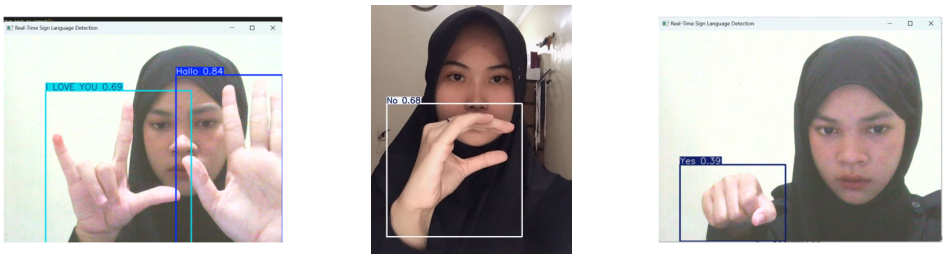
Class	Image	Instances	P	R	mAP50	mAP50-95
All	583	583	0.992	0.993	0.994	0.733
Hello	169	169	1	0.999	0.995	0.740
I Love You	102	102	0.997	1	0.995	0.774
No	93	93	0.978	0.968	0.990	0.685
Thank You	105	105	0.996	1	0.995	0.776
Yes	114	114	0.991	1	0.995	0.690

In Figure 8, the first row shows that the training was conducted using a GPU with CUDA, which is faster than a CPU. The model was trained to recognize 3019 classes. The number of classes comes from the test dataset used. Training Accuracy: The model accuracy on the training data is 94.67%, which means the model successfully classifies most of the training data correctly. Testing Accuracy: The model accuracy on the test data is 93.02%, which shows that the model can generalize well to new data that has never been seen before, stated in Figure 8.

```
Using device: cuda
Number of classes: 3019
Training Epoch 1/5: 100%|██████████| 432/432 [01:19<00:00,  5.42it/s]
Epoch [1/5], Loss: 0.4883
Training Epoch 2/5: 100%|██████████| 432/432 [01:14<00:00,  5.81it/s]
Epoch [2/5], Loss: 0.0673
Training Epoch 3/5: 100%|██████████| 432/432 [01:14<00:00,  5.82it/s]
Epoch [3/5], Loss: 0.0454
Training Epoch 4/5: 100%|██████████| 432/432 [01:14<00:00,  5.82it/s]
Epoch [4/5], Loss: 0.0178
Training Epoch 5/5: 100%|██████████| 432/432 [01:14<00:00,  5.78it/s]
Epoch [5/5], Loss: 0.0168
Calculating Accuracy: 100%|██████████| 432/432 [00:28<00:00, 15.41it/s]
Calculating Accuracy: 100%|██████████| 58/58 [00:04<00:00, 14.19it/s]Train Accuracy: 0.9467
Test Accuracy: 0.9302
Loaded 3019 training samples and 401 testing samples.
```

Figure 8. Result of Training Accuracy vs. Testing Accuracy.

In the web cam system test, it was able to detect American Sign Language (ASL) gestures effectively and efficiently in real-time. As can be seen in Figure 9a, the model is able to detect the I Love You sign with a confidence score of 0.69, the Hello sign 0.84. The No sign is detected with a confidence score of 0.68, in Figure 9b. For the Yes sign, it is detected with a confidence score of 0.39 as shown in Figure 9c. The system not only recognized the ASL gestures with high accuracy, but it also translated the detected signs into corresponding text, providing immediate feedback to the user. Moreover, the system incorporated a voice output feature that transformed the detected sign language gestures into spoken words, making it even more accessible for individuals who are not familiar with sign language. This audio feedback further enhances the usability of the system, especially for non-sign language users, by bridging the communication gap effectively. The real-time processing capability ensured that the user received prompt and accurate translations without noticeable delays, which is crucial for maintaining a natural flow in conversations.



(**a**) (**b**) (**c**)

Figure 9. Web cam output. (**a**) I Love You and Hello Sign; (**b**) No Sign; (**c**) Yes Sign.

5. Conclusions

This study, using YOLO-v11 as the main method in detecting sign language, obtained a training model accuracy of 94.67% and a testing model accuracy of 93.02%, indicating that the model has very good performance in recognizing sign language from the training and testing datasets. Additionally, the model is very reliable in recognizing the classes "Hello", "I Love You", "No", and "Thank You" with a sensitivity close to or equal to 100%. This shows that the model has the ability to recognize the main gestures very well. The sensitivity for the "Background" class is still low, which means that the model often misrecognizes "Background" or mixes it with other classes. However, there are several things that need to be improved, namely: adding training data for classes that are often mispredicted, adding representative data, using a loss function that handles class imbalance, and optimizing the threshold or data augmentation for the class. With this research, it is hoped that sign language detection can be applied in life so that it can help people with disabilities in terms of socializing, especially communicating with others, and that communication equality is achieved.

Author Contributions: Conceptualization, I.L.K., M.A.S.Y., I.N., Y.L. and S.D.S.; Methodology, I.N., Y.L. and S.D.S.; Software, I.N., I.L.K. and K.; Validation, I.L.K. and M.A.S.Y.; Formal Analysis, Y.L. and S.D.S.; Resources, I.N., Y.L. and S.D.S.; Writing—Original Draft Preparation, I.N., Y.L. and S.D.S.; Writing—Review and Editing, I.L.K., K. and I.N. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Medellu, J.; Sambul, A.; Lumenta, A.S.M. Aplikasi Pendeteksian Bentuk Gestur Tangan dalam Bahasa Isyarat. *J. Tek. Inform.* **2022**, *17*, 4.
2. Hamama, S.; Nurseha, M.A. Memahami Komunikasi Verbal dalam Interaksi Manusia. *Selasar J. Komun. Penyuluhan* **2023**, *4*, 1. Available online: https://ejournal.iainu-kebumen.ac.id/index.php/selasar/article/view/1887 (accessed on 20 June 2025).
3. Rahmah, W.A.; Janah, S. Efektivitas Penggunaan Bahasa Isyarat bagi Penyandang. *J. Ilm. Multidisiplin Terpadu* **2024**, *8*, 6.
4. Laely, T.A.; Aerin, W. Pengembangan Keterampilan Berbahasa Lisan Anak Tunarungu Melalui Terapi Bermain di TK Masyitoh Talang Tegal. *Prosiding ACIECE* **2019**, *4*, 319–329. Available online: https://conference.uin-suka.ac.id/index.php/aciece/article/view/143 (accessed on 20 June 2025).
5. Armas, A.M.; Unde, A.A.; Fatimah, J.M. Konsep Diri dan Kompetensi Komunikasi Penyandang Disabilitas dalam Menumbuhkan Kepercayaan Diri dan Aktualisasi Diri di Dunia Kewirausahaan Kota Makassar. *Kareba J. Ilmu Komun.* **2017**, *6*, 2. [CrossRef]

6. Borman, R.I.; Priopradono, B.; Syah, A.R. Klasifikasi Objek Kode Tangan pada Pengenalan Isyarat Alphabet Bahasa Isyarat Indonesia (Bisindo). *Semin. Nas. Inform. UNJANI* **2017**, *1*. Available online: https://www.academia.edu/57264937/Klasifikasi_Objek_Kode_Tangan_pada_Pengenalan_Isyarat_Alphabet_Bahasa_Isyarat_Indonesia_BISINDO_ (accessed on 20 June 2025).
7. Hadi, M.A.; Ferdian, R.; Arief, L. Klasifikasi Tingkat Ancaman Kriminalitas Bersenjata Menggunakan Metode You Only Look Once (YOLO). *CHIPSET* **2017**, *6*, 1. Available online: http://chipset.fti.unand.ac.id/index.php/chipset/article/view/57/17 (accessed on 20 June 2025).
8. Redmon, J. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2016), Las Vegas, NV, USA, 26 June–1 July 2016; Volume 1. Available online: http://pjreddie.com/yolo/ (accessed on 20 June 2025).
9. DigitalOcean. Apa yang Baru di YOLOv11: Mengubah Deteksi Objek Sekali Lagi Bagian 1. *DigitalOcean Blog* **2024**, *10*.
10. Maleh, I.M.D.; Teguh, R.; Sahay, A.S.; Okta, S.; Pratama, M.P. Implementasi Algoritma You Only Look Once (YOLO) untuk Object Detection Sarang Orang Utan. *J. Inform. BSI* **2023**, *10*, 19–27. [CrossRef]
11. Arikeri, P. American Sign Language (ASL) Dataset; Kaggle. Available online: https://www.kaggle.com/datasets/prathumarikeri/american-sign-language-09az (accessed on 20 June 2025).
12. Priyadharshini, D.S.; Anandraj, R.; Ganesh Prasath, K.R.; Franklin Manogar, S.A. A Comprehensive Application for Sign Language Alphabet and Word Recognition. In Proceedings of the 2024 International Conference on Science Technology Engineering and Management (ICSTEM), Coimbatore, India, 26–27 April 2024. Available online: https://ieeexplore.ieee.org/document/10561024 (accessed on 20 June 2025).
13. Kayahan, D.; Güngör, T. A Hybrid Translation System from Turkish Spoken Language to Turkish Sign Language. In Proceedings of the 2019 IEEE International Symposium on INnovations in Intelligent SysTems and Applications (INISTA), Sofia, Bulgaria, 3–5 July 2019. Available online: https://ieeexplore.ieee.org/document/8778347 (accessed on 20 June 2025).
14. Grif, M.; Manueva, Y. Semantic Analyses of Text to Translate to Russian Sign Language. In Proceedings of the 2016 11th International Forum on Strategic Technology (IFOST), Novosibirsk, Russia, 1–3 June 2016. Available online: https://ieeexplore.ieee.org/document/7884107 (accessed on 20 June 2025).
15. Camgoz, N.C.; Koller, O.; Hadfield, S.; Bowden, R. Sign Language Transformers: Joint End-to-End Sign Language Recognition and Translation. *arXiv* **2020**, arXiv:2003.13830v1. [CrossRef]
16. Pranto, R.H.; Siddique, S. Real-Time Bangla Sign Language Translator. *arXiv* **2024**, arXiv:2412.16497v1.
17. Chen, Y.; Wei, F.; Sun, X.; Wu, Z.; Lin, S. A Simple Multi-Modality Transfer Learning Baseline for Sign Language Translation. *arXiv* **2022**, arXiv:2203.04287v2. [CrossRef]
18. Li, D.; Opazo, C.R.; Yu, X.; Li, H. Word-Level Deep Sign Language Recognition from Video. *arXiv* **2019**, arXiv:1910.11006v2. [CrossRef]
19. Izutov, E. ASL Recognition with Metric-Learning Based Lightweight Network. *arXiv* **2020**, arXiv:2004.05054v1. [CrossRef]
20. Imran, A.; Hulikal, M.S.; Gardi, H.A. Real Time American Sign Language Detection Using Yolo-v9. *arXiv* **2024**, arXiv:2407.17950v1. [CrossRef]
21. Sadeghzadeh, A.; Shah, A.S.; Islam, B. MLMSign: Multi-lingual Multi-modal Illumination-Invariant Sign Language Recognition. *Intell. Syst. Appl.* **2024**, *16*, 200384. [CrossRef]
22. Naz, N.; Sajid, H.; Ali, S.; Hasan, O.; Ehsan, M.K. MIPA-ResGCN: A Multi-Input Part Attention Enhanced Residual Graph Convolutional Framework for Sign Language Recognition. *Comput. Ind. Eng.* **2023**, *177*, 108024. [CrossRef]
23. Jaiswal, M.; Sharma, A.; Saini, S. Hardware Acceleration of Tiny YOLO Deep Neural Networks for Sign Language Recognition. *Decis. Support Syst.* **2025**, *180*, 113752. [CrossRef]
24. Talaat, F.M.; El-Shafai, W.; Soliman, N.F.; Algarni, A.D.; El-Samie, F.E.A.; Siam, A.I. Real-time Arabic Avatar for Deaf-Mute Communication. *Comput. Ind. Eng.* **2024**, *186*, 109043. [CrossRef]
25. Alsharif, B.; Alalwany, E.; Ilyas, M. Transfer Learning with YOLOv8 for Real-Time Recognition of American Sign Language Alphabet. *ResearchGate* **2024**, *preprint*. Available online: https://www.researchgate.net/publication/384845818_Transfer_learning_with_YOLOV8_for_real-time_recognition_system_of_American_Sign_Language_Alphabet (accessed on 20 June 2025). [CrossRef]
26. Jain, S. ADDSL: Hand Gesture Detection and Sign Language Recognition. *arXiv* **2023**, arXiv:2305.09736v1. [CrossRef]
27. Shenoda, M. Real-time Object Detection YOLOv1 Re-Implementation in PyTorch; ResearchGate. Available online: https://www.researchgate.net/publication/371136193_Real-time_Object_Detection_YOLOv1_Re-Implementation_in_PyTorch (accessed on 20 June 2025).
28. Cooljugator. How To Learn ASL: A Guide for Beginners. *Cooljugator Blog* **2023**, *10*. Available online: https://cooljugator.com/blog/how-to-learn-asl/ (accessed on 20 June 2025).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.