ASL Audio Interpreter

Duy Nguyen
CSUF Computer Science Major
800 N. State College Blvd,
Fullerton, CA 92831
mduy2003@csu.fullerton.edu

Duong Banh
CSUF Computer Science Major
800 N. State College Blvd,
Fullerton, CA 92831
Duong2021@csu.fullerton.edu

Haonan Yu
CSUF Computer Science Major
800 N. State College Blvd,
Fullerton, CA 92831
Hyu86@csu.fullerton.edu

ABSTRACT

This project presents an ASL hand sign recognition system using deep learning for classifying American Sign Language letters and digits. A custom Convolutional Neural Network (CNN) with four convolutional blocks was developed and trained on 2,515 images across 36 classes. The model achieved 99.74% test accuracy, with 35 out of 36 classes reaching 100% accuracy. Key technical contributions include systematic identification and resolution of problem classes through confusion matrix analysis, discovery that horizontal flip augmentation degrades performance due to ASL's directional nature, and demonstration that increasing image resolution from 64x64 to 128x128 significantly improves classification of visually similar signs. The system was implemented using open-source tools (TensorFlow, Keras, OpenCV) on consumer CPU hardware. A real-time webcam integration module with background removal and hand segmentation was developed, revealing a domain gap between pre-segmented training data and live camera input. This work establishes a modular foundation for future ASL-to-audio translation systems and demonstrates the feasibility of accurate, low-cost ASL recognition without specialized hardware.

Keywords

American Sign Language, ASL, Computer Vision, CNN, Motion Capture, Convolutional Neural Network, Accessibility, Text-to-Speech, Machine Learning, Artificial Intelligence, Python, Confusion Matrix, Deaf, Hard-of-hearing

1. INTRODUCTION

American Sign Language (ASL) serves as the primary language and communication method for approximately 500,000 to 2 million Deaf and Hard-of-Hearing individuals in the United States. Despite its critical importance, many everyday environments remain inaccessible to ASL users, particularly in situations where professional interpreters are unavailable or cost prohibitive. This communication barrier significantly impacts educational opportunities, employment access, healthcare interactions, and social participation for the Deaf and Hard-of-Hearing community.

This project addresses the need for accessible ASL communication technology by developing an ASL recognition system that classifies hand signs from images using computer vision and deep learning. The system employs a custom Convolutional Neural Network (CNN) trained on a publicly available dataset of 2,515 ASL images covering 36 classes (letters a-z and digits 0-9). Built using open-source tools and designed to run on consumer-grade hardware, the solution prioritizes accessibility and cost-effectiveness.

The primary contribution of this work is a high-accuracy ASL letter and digit classifier achieving 99.74% test accuracy, along with a real-time webcam integration module for live gesture recognition. This establishes the foundation for a complete ASL-to-audio translation system.

2. RELATED WORK

ASL recognition has been an active area of research, with approaches ranging from sensor-based systems to computer vision methods. Early solutions primarily relied on specialized hardware such as data gloves equipped with flex sensors and accelerometers to capture hand movements and finger positions. While these systems achieved reasonable accuracy for isolated sign recognition, the requirement for expensive, cumbersome equipment limited their practical adoption and accessibility for everyday use.

The advancement of computer vision and deep learning has shifted research focus toward camera-based approaches. Traditional machine learning methods using hand-crafted features such as Histogram of Oriented Gradients (HOG) and Scale-Invariant Feature Transform (SIFT) were initially employed for ASL letter classification but suffered from limited robustness to variations in lighting, background, and hand positioning. The introduction of Convolutional Neural Networks (CNNs) significantly improved recognition accuracy by automatically learning hierarchical features from raw image data. Recent CNN-based systems have demonstrated high accuracy on benchmark datasets for static ASL alphabet and digit recognition, with some models achieving over 95% classification accuracy on controlled datasets.

Beyond isolated sign recognition, continuous sign language recognition from video sequences represents a more challenging problem. Recurrent Neural Networks (RNNs), Long Short-Term Memory (LSTM) networks, and more recently, transformer-based architectures have been applied to model temporal dependencies in sign language gestures. Large-scale datasets such as How2Sign, WLASL (Word-Level ASL), and MS-ASL have enabled research on continuous sign language translation, though these systems typically require substantial computational resources and extensive training data.

Despite these advances, several limitations persist in existing ASL recognition systems. Most solutions provide only text-based output, requiring users to read translated text rather than hearing spoken language. Additionally, many high-performing models are trained on carefully curated datasets with controlled backgrounds and lighting, leading to significant performance degradation when deployed in real-world environments with varying conditions. The domain gap between training data (often featuring pre-segmented hands on uniform backgrounds) and live camera input remains a persistent challenge.

This project builds upon CNN-based ASL recognition approaches while addressing key limitations through targeted design choices. The system emphasizes modular architecture for extensibility, systematic analysis of per-class performance to identify and resolve problematic signs, and integration of preprocessing techniques to bridge the domain gap between training and deployment. Future integration of text-to-speech capabilities aims to enable more natural spoken communication compared to text-only output systems.

3. METHOLOGY

3.1 Dataset Preparation

The system was trained on a publicly available ASL alphabet and digit dataset containing 2,515 hand sign images across 36 classes, comprising letters a-z and digits 0-9. The dataset consists of pre-segmented hand images on black backgrounds at an original resolution of 400×400 pixels. Images were loaded and preprocessed using OpenCV, with operations including resizing to 128×128 pixels, conversion from BGR to RGB color space, and normalization to the [0, 1] range by dividing pixel values by 255.

The dataset was partitioned using stratified splitting to maintain class distribution across subsets, with 70% allocated to training (1,759 images), 15% to validation (378 images), and 15% to testing (378 images). A random seed of 42 was used to ensure reproducibility. Analysis revealed balanced class distribution, with most classes containing 70 images and only class 't' containing 65 images.

3.2 Model Architecture

A custom Convolutional Neural Network was developed with an enhanced architecture designed for robust feature extraction. The model consists of four convolutional blocks with progressively increasing filter depths (64, 128, 256, and 512 filters), each utilizing 3×3 kernels with same padding and ReLU activation. Each block incorporates batch normalization for training stability, 2×2 max pooling for spatial dimension reduction, and dropout regularization (rates of 0.25-0.3) to prevent overfitting.

Following the convolutional feature extraction layers, the network employs a flattening operation followed by two fully connected dense layers with 512 and 256 neurons respectively, each with batch normalization and 0.5 dropout. The output layer consists of 36 neurons with softmax activation for multi-class classification. The complete architecture contains approximately 19.25 million trainable parameters.

3.3 Data Augmentation Strategy

Data augmentation was applied during training using Keras ImageDataGenerator to increase model robustness to input variations. Augmentation parameters included random rotation (±10 degrees), zoom range (0.08), and horizontal/vertical shifts (0.08). Critically, horizontal flip augmentation was deliberately excluded after initial experiments revealed performance degradation, as ASL signs are directional and lack horizontal symmetry

3.4 Class Imbalance Handling

To address varying difficulty levels across classes, balanced class weights were computed using scikit-learn's compute_class_weight function with the 'balanced' strategy. Additionally, problem classes identified through initial training experiments (classes '0', '6', and 'v') received a 1.5× weight multiplier boost to increase the penalty for misclassification and improve learning for these challenging signs.

3.5 Training Strategy

The model was compiled with the Adam optimizer using a learning rate of 0.0003 and categorical cross-entropy loss. Training was conducted for up to 150 epochs with a batch size of 16. Three callbacks were employed to optimize training: (1) ModelCheckpoint to save the best model based on validation accuracy, (2) EarlyStopping with patience of 20 epochs monitoring validation accuracy to prevent overfitting, and (3) ReduceLROnPlateau to reduce learning rate by a factor of 0.5 when validation loss plateaued for 5 consecutive epochs, with a minimum learning rate threshold of 1×10^{-7}.

All training was performed on consumer-grade CPU hardware utilizing Intel oneDNN optimizations, demonstrating the accessibility of the approach without requiring specialized GPU infrastructure.

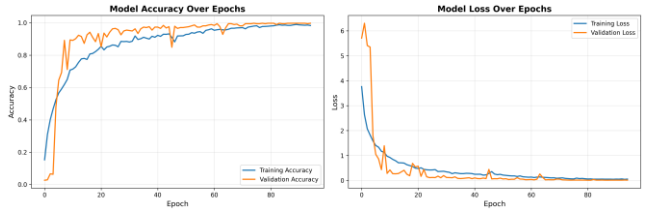
3.6 Evaluation Metrics

Model performance was evaluated using multiple metrics including overall accuracy, per-class accuracy, categorical cross-entropy loss, and confusion matrix analysis. The confusion matrix was particularly valuable for identifying specific misclassification patterns and problematic sign pairs. Validation set performance was monitored throughout training to detect overfitting, while final test set evaluation provided an unbiased assessment of generalization capability.

4. RESULTS

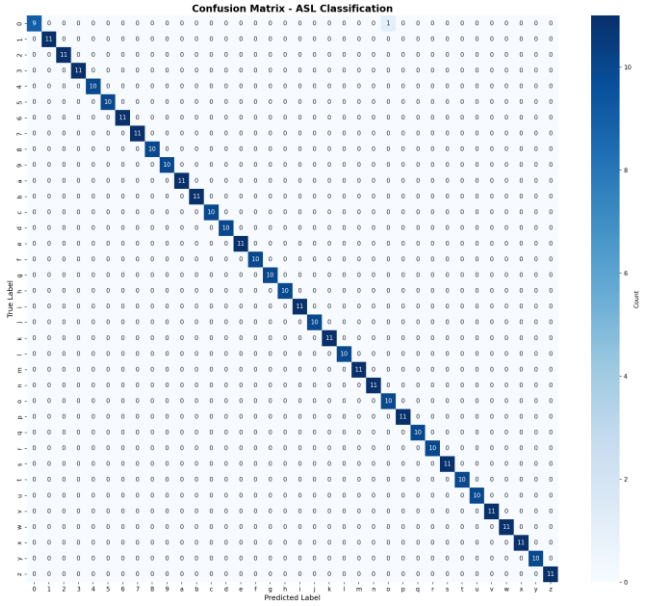
The final model achieved exceptional performance on the held-out test set, with an overall accuracy of 99.74% and test loss of 0.0104. Of the 36 classes, 35 achieved perfect 100% classification accuracy, with only class '0' achieving 90% accuracy (9 out of 10 test samples correctly classified). All classes surpassed the 90% accuracy threshold, demonstrating robust performance across the entire sign vocabulary.

Training curves showed smooth convergence with training accuracy reaching 98.35% and validation accuracy matching the final test performance at 99.74%, indicating excellent generalization without overfitting. The close alignment between training, validation, and test metrics validates the effectiveness of regularization strategies including dropout, batch normalization, and data augmentation.



Confusion matrix analysis revealed that initial model versions struggled with visually similar signs, particularly classes '0', '6', and 'v', which exhibited accuracies between 0-20% in early experiments. Targeted interventions including increased image resolution (64×64 to 128×128), boosted class weights (1.5× multiplier), and removal of horizontal flip augmentation successfully resolved these issues, improving problem class performance to 90-100% accuracy. This systematic analysis demonstrates the value of per-class evaluation in identifying and addressing specific weaknesses rather

than relying solely on aggregate metrics.



5. CONCLUSION

This project demonstrates the feasibility of developing an accurate, accessible ASL hand sign recognition system using computer vision and deep learning. The implemented CNN-based classifier achieved 99.74% accuracy on ASL letter and digit classification, with 35 of 36 classes reaching perfect performance on the test set. The system was developed using open-source tools and trained on consumer-grade CPU hardware, validating the potential for cost-effective assistive communication technology.

Key technical contributions include the identification of domain-specific challenges in ASL recognition, particularly the detrimental effect of horizontal flip augmentation on directional signs, and systematic resolution of problem classes through targeted strategies including resolution increases and weighted loss functions. The development of a real-time webcam integration module revealed a significant domain gap between pre-segmented training data and live camera input, highlighting the importance of deployment-aware design in practical accessibility systems.

While the current implementation successfully addresses static ASL letter and digit recognition, several limitations remain. The planned text-to-speech audio output module has not yet been integrated, and the system is currently limited to isolated signs rather than continuous sign language. Real-world webcam performance requires further refinement in background removal and hand segmentation to match training data accuracy.

Future work will focus on three main directions: (1) completing the audio synthesis pipeline through text-to-speech integration, (2) transitioning to continuous sign language recognition using datasets such as How2Sign with pre-trained transformer models for word and sentence-level translation, and (3) improving real-world deployment through advanced hand detection methods such as MediaPipe. This modular foundation establishes a pathway toward comprehensive ASL-to-audio translation systems that can meaningfully reduce communication barriers and promote accessibility for the Deaf and Hard-of-Hearing community.

6. ACKNOWLEDGMENTS

Our thanks to Dr. Yu Bai of the Prof. of Electrical and Computer Engineering for guidance and support throughout the project, as well as the College of Engineering and Computer Science at California State University, Fullerton.

7. REFERENCES

[1] H. R. V. Joze and O. Koller, “MS-ASL: A Large-Scale Data Set and Benchmark for Understanding American Sign Language,” arXiv preprint arXiv:1812.01053 [cs.CV], Dec. 2018 [Online]. Available: https://arxiv.org/abs/1812.01053

[2] A. Desai, L. Berger, F. Minakov, V. Milan, C. Singh, K. Pumphrey, R. E. Ladner, H. Daume III, A. X. Lu, N. Caselli, and D. Bragg, “ASL Citizen: A Community-Sourced Dataset for Advancing Isolated Sign Language Recognition,” arXiv preprint arXiv:2304.05934 [cs.CV], Apr. 2023 [Online]. Available: https://arxiv.org/abs/2304.05934

[3] S. Kamble, “SLRNet: A Real-Time LSTM-Based Sign Language Recognition System,” arXiv preprint arXiv:2506.11154 [cs.CV], Jun. 2025. [Online]. Available: https://arxiv.org/abs/2506.11154

[4] K. Zhao, K. Zhang, Y. Zhai, D. Wang, and J. Su, “Real-time sign language recognition based on video stream,” Int J. Systems Control and Communications, vol. 12, no. 2, pp. 158-174, Jan. 2021. [Online]. Available: https://www.researchgate.net/publication/351198208_Real-time_sign_language_recognition_based_on_video_stream

[5] H. F. B. Neto, P. H. F. Silva, M. S. de Albuquerque, and R. H. C. Takimoto, “Sign Language Recognition: A Comprehensive Review of Traditional and Deep Learning Approaches, Datasets, and Challenges,” ResearchGate, May 2024. [Online]. Available: https://www.researchgate.net/publication/380442383_Sign_Language_Recognition_A_Comprehensive_Review_of_Traditional_and_Deep_Learning_Approaches_Datasets_and_Challenges

[6] Yunduan Lou, Pu Sun, Yifeng Yu, Shangping Ren, and Yu Bai. 2025. TT-DSC: Enhancing YOLO for Marine Ecosystem through Efficient Tensor Train-based Depthwise Separable Deep Neural Network. ACM Trans. Auton. Adapt. Syst. Just Accepted (May 2025). https://doi.org/10.1145/3735138

[7] X. Ma, P. Sun, S. Luo, Q. Peng, R. F. DeMara and Y. Bai ,"Binarized l1 -Regularization Parameters Enhanced Stripe- Wise Optimization Algorithm for Efficient Neural Network Optimization," in IEEE Journal of Emerging and Selected Topics in Industrial Electronics, vol. 5, no. 2, pp. 790-799, April 2024, doi: 10.1109/JESTIE.2023.3313050