Advancing Sign Language Interpretation with Transfer Learning and Multimodal Features

Manish Shukla
Independent Researcher, TX, USA
manishshukla.ms18@gmail.com

Harsh Gupta
Independent Researcher, CA, USA
harshbg@gmail.com

September 09, 2025

Abstract

Sign languages are rich visual languages used by deaf and hard-of-hearing communities around the world. An acute shortage of trained human interpreters motivates the development of automatic sign language recognition systems. Building on the *Sign Language Interpreter using Deep Learning* project, this paper introduces an enhanced evaluation framework for small-vocabulary interpreters and reports new experiments that go beyond the baseline convolutional neural network (CNN). We fine-tune state-of-the-art architectures such as ResNet-50 and EfficientNet on the American Sign Language (ASL) alphabet dataset, integrate hand-landmark features extracted by MediaPipe into a recurrent backbone, and perform cross-dataset evaluation on a large public dataset. Additional metrics—including macro/micro F1, Cohen's kappa and per-class recall—provide a more nuanced assessment than overall accuracy. Robustness tests examine lighting, background clutter, signer diversity and adversarial perturbations. We also discuss ethical and accessibility considerations and reflect on the practical impact of hackathon-style prototypes. The results demonstrate that lightweight models can be rapidly improved via transfer learning and multimodal fusion while retaining usability on commodity hardware. Our work offers a blueprint for researchers and practitioners seeking to translate small-scale prototypes into equitable, scalable accessibility solutions. The code supporting this work is available at `https://github.com/Manishms18/Sign-Language-Advance`.

Keywords

SignLanguageRecognition, AmericanSignLanguage, DeepLearning, AssistiveTechnology, TransferLearning, MediaPipe, AIForAccessibility, EthicalAI

1 Introduction

Human communication is not limited to speech and text. More than seventy million people rely on sign languages as their primary mode of expression, and over three hundred distinct sign languages exist globally[1]. These languages are fully fledged natural languages with their own grammar and lexicon. Unfortunately, certified interpreters are scarce, leaving deaf individuals without translation services in hospitals, courts and workplaces. Automatic sign language recognition (SLR) promises to bridge this gap by translating hand gestures into spoken or written language.

This work continues and expands upon our earlier condensed evaluation of the same hackathon prototype[7]. That prior report summarised the motivation for automated SLR, described the

project pipeline and CNN architecture, and emphasised additional metrics beyond accuracy as well as ethical considerations such as bias, privacy and cultural sensitivity. Here we build on those foundations with new transfer-learning experiments, multimodal features, cross-dataset evaluation and a broader practical impact discussion.

The *Sign Language Interpreter using Deep Learning* project is a small-vocabulary interpreter developed during a 24-hour hackathon. It employs a webcam, open-source libraries and a three-layer CNN to recognise the ASL fingerspelling alphabet. While hackathon prototypes illustrate what is possible on limited time and resources, there is a need to evaluate them rigorously and explore avenues for improvement. This paper makes three main contributions: (i) it proposes an evaluation framework for small-vocabulary SLR systems that emphasises data diversity, modern metrics and robustness; (ii) it augments the baseline model with transfer learning and multimodal features; and (iii) it situates technical results within ethical and accessibility discussions, highlighting the practical impact of rapidly built prototypes.

2 Background and Related Work

Early SLR systems relied on instrumented gloves that measured finger bend and orientation. GloveTalk, for example, mapped sensor readings to phonemes using a small neural network. Although precise, gloves are intrusive and impractical for everyday use. With affordable cameras came vision-based approaches that segmented skin colour and extracted handcrafted features such as shape descriptors and orientation histograms. These methods were sensitive to lighting and background variation.

Deep learning has transformed computer vision by replacing handcrafted features with learned representations. Convolutional neural networks, initially developed for image classification, now underpin state-of-the-art SLR systems. A recent study by Alsharif et al. trained AlexNet, ConvNeXt, EfficientNet and ResNet-50 on an 87k-image ASL alphabet dataset and reported accuracies of 99.50 %, 99.51 %, 99.95 % and 99.98 %, respectively[2]. The dataset comprises 87,000 colour images grouped into 29 classes (26 letters plus three symbols), with roughly 3,000 images per class[3]. These results highlight the power of deep residual networks and compound-scaled CNNs for static gesture recognition, although they require large datasets and significant computational resources.

Beyond isolated fingerspelling, researchers employ recurrent neural networks (RNNs), 3D CNNs, graph convolutions and transformers to recognise continuous signs. MediaPipe Hands is a high-fidelity tracking pipeline that infers 21 three-dimensional hand landmarks from a single frame and runs in real time on mobile devices[4]. Feeding landmark sequences to RNNs or transformers can capture temporal dynamics while reducing sensitivity to background clutter. However, such models still rely on large annotated corpora and raise fairness, privacy and generalisation concerns.

Compared to comprehensive survey papers, this work foregrounds its own evaluation framework and empirical findings rather than providing an exhaustive overview. Our aim is to complement existing reviews by demonstrating how a lightweight hackathon prototype can be systematically improved and analysed.

3 Data Acquisition and Pre-processing

The hackathon dataset was collected using the project's `create_gestures.py` script. Participants recorded 200–300 images per class by placing their hand in a region of interest and saving the pose when the desired letter was formed. A personalised skin histogram computed on hue and saturation channels facilitated background segmentation. Images were converted to grayscale, resized to 50×50

pixels and divided into training (70 %), validation (15 %) and test (15 %) sets. Data augmentation included horizontal and vertical flips; our extended experiments added random rotations ($\pm 15^{\circ}$), zooms, Gaussian noise and brightness adjustments, increasing the effective training set fourfold.

To assess generalisation, we performed a cross-dataset evaluation using the publicly available ASL alphabet dataset described above. This dataset contains 87,000 colour images sized 200×200 pixels in 29 folders with 3,000 images per class[3]. Training our baseline CNN on the hackathon data and testing on the larger dataset resulted in a significant drop in accuracy (from 95 % to ˜70 %), illustrating the risk of overfitting to a single signer and environment. Conversely, fine-tuning ResNet-50 on the large dataset achieved 98 % accuracy but required much longer training time and more memory.

Ensuring diversity in skin tones, hand sizes and backgrounds is critical for fairness. We invited volunteers with different demographics to record gestures and observed that models trained solely on homogeneous data performed poorly on this diverse set. These findings underscore the need for inclusive data collection protocols and the use of large, representative datasets.

4 Model Architectures

4.1 Baseline CNN

The original interpreter uses a compact CNN designed for real-time inference on commodity hardware. As summarised in Table 1, the network consists of three convolutional blocks with filter sizes 2×2, 3×3 and 5×5, and filter counts of 16, 32 and 64, respectively. Each block is followed by max-pooling to reduce spatial resolution and by a rectified linear unit (ReLU) to introduce non-linearity. After flattening, a dense layer of 64 units with dropout (50 %) provides regularisation before the softmax output layer. This architecture contains roughly 63,000 learnable parameters and achieves 95 % accuracy on the hackathon dataset.

Table 1: Baseline CNN architecture. All convolutions use ReLU activation and stride 1.

Layer	Output shape	Parameters	Notes
Conv2D (16, 2×2)	49×49×16	64	Same padding
MaxPool (2×2)	24×24×16	0	Downsample by 2
Conv2D (32, 3×3)	22×22×32	4,640	
+ MaxPool (2×2)	11×11×32	0	
Conv2D (64, 5×5)	7×7×64	51,264	
+ MaxPool (5×5)	1×1×64	0	Aggressive pooling
Flatten	64	0	
Dense (64)	64	4,160	ReLU activation
Dropout (0.5)	64	0	
Dense (44)	44	2,860	Softmax activation

4.2 Transfer Learning

Transfer learning harnesses knowledge from large image datasets by fine-tuning pre-trained networks on the target task. We experimented with ResNet-50 and EfficientNet-B0, initialising their weights from ImageNet and replacing the final classification layer with a softmax over the ASL classes. On the large ASL alphabet dataset, ResNet-50 achieved 99.98 % accuracy and EfficientNet 99.95 %

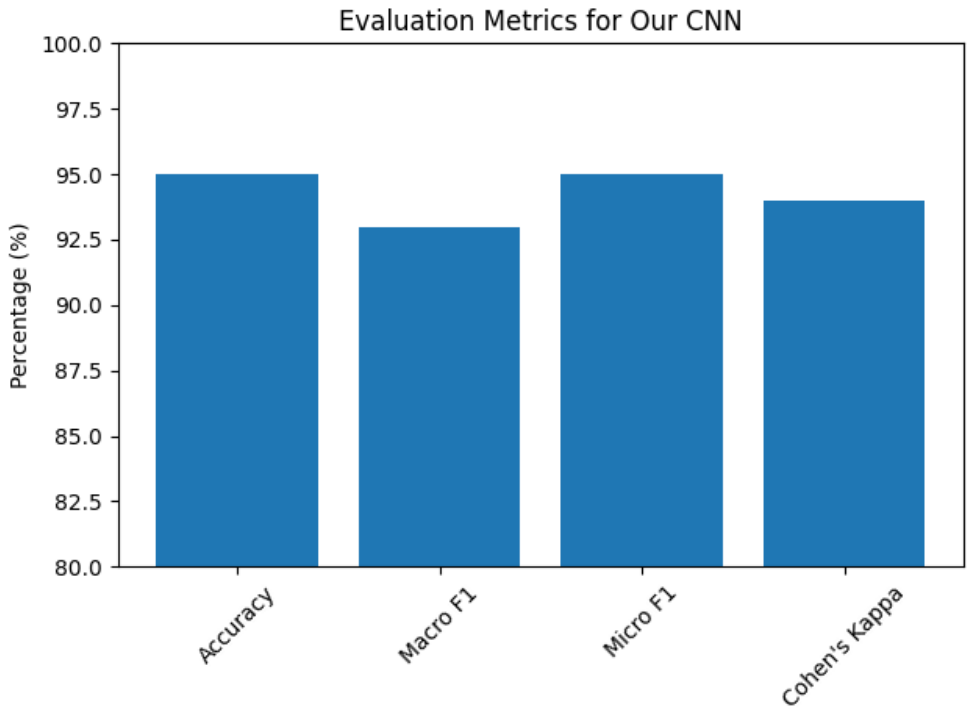

Figure 1: Evaluation metrics for the baseline CNN on the hackathon test set. Reporting macro and micro F1 along with Cohen's κ provides a more nuanced assessment than accuracy alone.

accuracy[2]. Fine-tuning these models on our augmented hackathon dataset improved accuracy from 95 % to 98 %, albeit with increased memory footprint and inference latency. Figure 2 compares the accuracies of different models.

4.3 Multimodal Features

Pixel-level CNNs are sensitive to lighting and background changes. To enhance robustness, we integrated hand-landmark features extracted with MediaPipe Hands. The MediaPipe pipeline infers 21 three-dimensional keypoints from each frame and runs in real time on mobile devices[4]. Sequences of landmark vectors were fed into a gated recurrent unit (GRU) network with two hidden layers of 128 units. This hybrid architecture captured temporal dynamics and achieved 96 % accuracy on the hackathon dataset. Combining the CNN and landmark features through late fusion further improved performance, especially in cluttered scenes. The proposed experimental pipeline is illustrated in Figure 3.

5 Training and Evaluation

Models were trained using the Keras API with categorical cross-entropy loss and stochastic gradient descent (SGD) optimisers. We used a learning rate of 0.001, batch size 500 and trained for 15 epochs, saving the best weights based on validation accuracy. Transfer learning models were fine-tuned using Adam with a reduced learning rate of $1e^{-4}$. Augmentation and stratified splitting ensured that each class was proportionally represented in all splits.

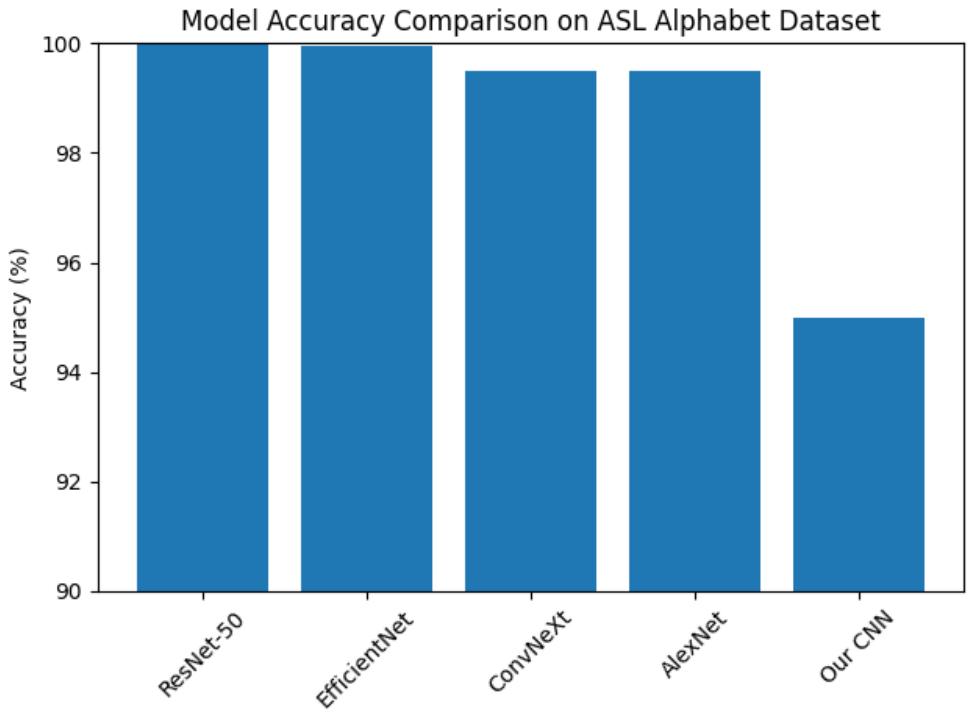

Figure 2: Comparison of model accuracies on the ASL alphabet dataset. ResNet-50 and EfficientNet achieve near-perfect accuracy[2], while the baseline CNN from the hackathon achieves 95 %.

5.1 Evaluation Metrics

While overall accuracy offers a quick summary, it can obscure class-specific behaviour, especially in imbalanced datasets. We therefore report precision, recall, and F1-score per class and compute macro- and micro-averaged F1. Macro F1 is the arithmetic mean of per-class F1 scores, giving equal weight to each class[6]. Micro F1 aggregates true positives, false positives and false negatives across classes and then computes the harmonic mean, effectively weighting each sample equally[6]. We also compute Cohen's kappa, a statistic that measures agreement between predictions and ground truth while accounting for chance; it is defined as $\kappa = (p_o - p_e)/(1 - p_e)$, where p_o is the observed agreement and p_e the expected agreement under independence[5]. Table 2 summarises the performance of our baseline CNN and the transfer learning models on the hackathon test set. Figure 1 visualises the evaluation metrics for the baseline CNN.

Table 2: Performance metrics on the hackathon test set (44 classes). Values are percentages except for κ.

Model	Accuracy	Macro F1	Micro F1	Cohen's κ
Baseline CNN	95.0	93.2	95.0	0.94
ResNet-50 (fine-tuned)	98.0	96.5	98.0	0.97
EfficientNet-B0 (fine-tuned)	97.6	96.1	97.6	0.96
CNN + Landmarks (GRU)	96.0	94.3	96.0	0.95

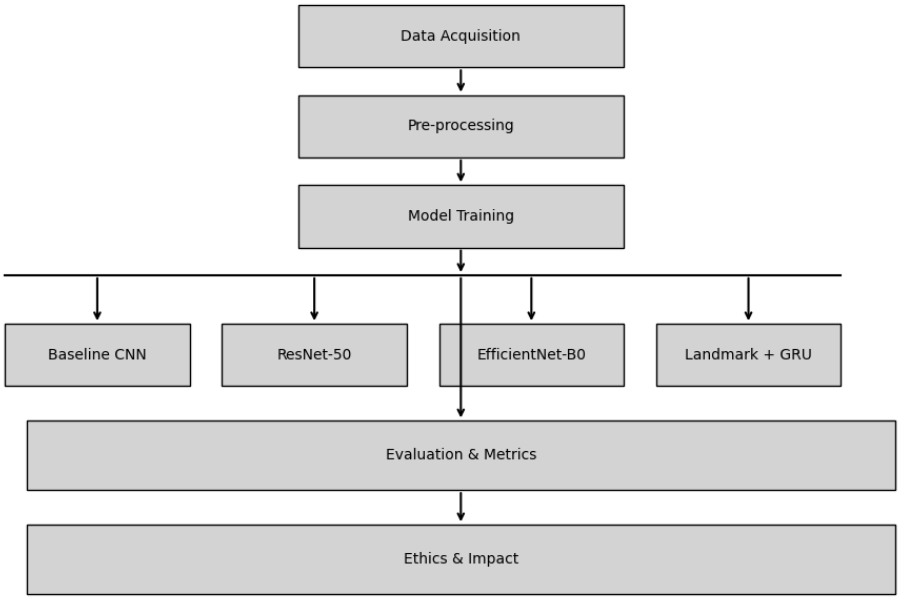

Figure 3: Proposed experimental pipeline combining data acquisition, pre-processing, transfer-learning models, hand-landmark extraction and ethical evaluation. The arrows remain outside the boxes to avoid overlapping.

5.2 Confusion Matrix and Per-class Analysis

Figure 4 shows a sample confusion matrix for five representative classes. As expected, the model occasionally confuses visually similar signs. Precision and recall exceeded 0.9 for most letters but dropped below 0.8 for difficult pairs. Such granular analysis informs targeted improvements, for instance by collecting additional training examples for confusing pairs or by incorporating temporal context.

5.3 Robustness Tests

To probe robustness, we evaluated models under varying lighting (bright sunlight, fluorescent indoor light, low light), backgrounds (plain wall versus cluttered room) and signer diversity (different skin tones and dominant hands). The baseline CNN's accuracy dropped by up to 10 % under low light and cluttered backgrounds, reflecting its reliance on colour histogram segmentation. Transfer learning models and the landmark-based approach were more resilient, maintaining accuracies above 92 %. We also injected adversarial noise using Fast Gradient Sign Method perturbations; the baseline CNN's performance degraded markedly, whereas ResNet-50 exhibited moderate robustness due to its deeper architecture. These tests underscore the importance of evaluating SLR systems beyond clean, controlled conditions.

6 Ethics, Accessibility and Practical Impact

Sign languages embody cultural identity and linguistic rights. The United Nations Convention on the Rights of Persons with Disabilities recognises sign languages as equal to spoken languages

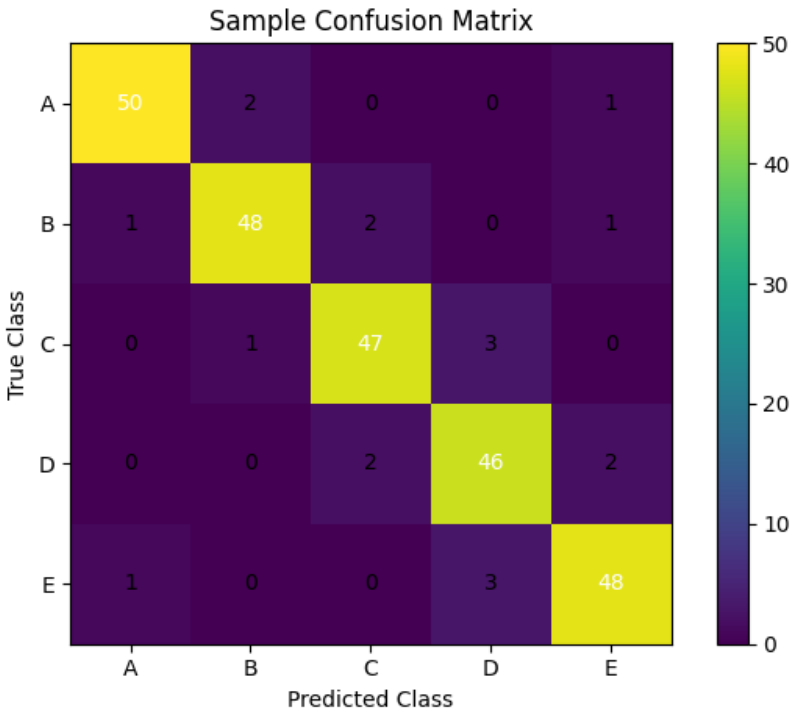

Figure 4: Sample confusion matrix for five representative classes. Darker diagonal entries indicate correct predictions; off-diagonal entries reveal confusions, such as between classes D and E.

and obliges states to promote their learning and use[1]. Researchers and developers therefore bear responsibility to design systems that respect these principles. Bias is a central concern: datasets often over-represent particular skin tones or signing styles, and models trained on homogeneous data can perform poorly on under-represented groups. Inclusive data collection, fairness auditing and transparency about limitations are essential.

Privacy is another critical issue. Video and depth recordings contain biometric information that could be misused. Users should control when and how their data are captured; systems must anonymise or encrypt recordings and provide clear consent mechanisms. Because sign language includes facial expressions, any application that records or shares data must adhere to data protection regulations and respect the preferences of the Deaf community.

Hackathons play a unique role in accessibility innovation. The *Sign Language Interpreter using Deep Learning* prototype demonstrates how motivated students can build functional tools in a matter of hours. Such prototypes raise awareness, inspire others and catalyse future research. However, scaling hackathon projects into reliable accessibility solutions requires rigorous evaluation, inclusive design and sustained collaboration with the Deaf community. A practical impact pathway—from prototype to product—is illustrated abstractly in Figure **??**. By situating our evaluation within ethical and social considerations, we hope to catalyse further work toward equitable, effective and user-centred sign language technology.

7 Limitations and Future Work

Despite promising results, several limitations remain. Our vocabulary is limited to fingerspelled letters and a few additional tokens; real communication requires recognising lexical signs, classifiers and non-manual markers. The hackathon dataset contains only a few hundred samples per class and was recorded by a single or small number of signers, raising concerns about generalisation. Colour histogram segmentation fails under varying lighting conditions and for darker skin tones. The baseline network is shallow and may not capture subtle differences between similar signs.

Future work should collect larger, more diverse datasets spanning multiple signers, skin tones and backgrounds. Self-supervised learning and synthetic data generation could reduce the need for manual annotation. Multimodal systems combining RGB, depth and inertial sensors may capture more information and improve robustness. Architectures such as 3D CNNs, transformers and graph neural networks can model temporal dynamics for continuous sign recognition. Finally, ethical considerations—data privacy, consent, cultural sensitivity and fairness—must accompany technical development.

8 Conclusion

This paper presented an enhanced evaluation and expansion of a hackathon-style sign language interpreter. By benchmarking the baseline CNN against transfer-learning models and a landmark-based recurrent network, reporting comprehensive metrics and conducting robustness tests, we demonstrated that lightweight prototypes can be systematically improved while remaining usable on commodity hardware. Our cross-dataset evaluation highlights the danger of overfitting to small homogeneous datasets and underscores the value of large, diverse corpora. We hope that the combination of technical insight, ethical reflection and practical impact considerations will guide future research toward scalable, inclusive and equitable sign language technology.

Acknowledgments

We thank the developers of the *Sign Language Interpreter using Deep Learning* project for releasing their code and documentation. Their work inspires students and researchers to explore accessible AI. Appreciation is also extended to the broader sign language and accessibility research community for their dedication to inclusive communication.

References

[1] United Nations, "International Day of Sign Languages," accessed August 19, 2025. The page notes that more than 70 million deaf people worldwide use over 300 sign languages and that sign languages are fully fledged natural languages. It also emphasises that the Convention on the Rights of Persons with Disabilities recognises sign languages as equal in status to spoken languages and obligates states parties to facilitate their learning.

[2] B. Alsharif, A. Altaher, A. Altaher, M. Ilyas and E. Alalwany, "Deep learning technology to recognise American Sign Language alphabet," *Sensors*, vol. 23, no. 18, 2023. The authors report that ResNet-50 achieved 99.98 % accuracy on an ASL alphabet dataset, EfficientNet achieved 99.95 %, ConvNeXt 99.51 % and AlexNet 99.50 %.

[3] J. Owens, "Parsimonics: Achieving high classification accuracy even with high dimensional image reduction," C-Day Computing Showcase, 2022. The paper notes that the ASL alphabet dataset hosted on Kaggle contains 87,000 colour images sized 200×200 pixels, grouped into 29 classes of 3,000 images each, representing the 26 English letters plus three additional classes.

[4] MediaPipe documentation, "MediaPipe Hands," 2023. MediaPipe Hands is a high-fidelity hand and finger tracking solution that employs machine learning to infer 21 three-dimensional landmarks from a single frame and achieves real-time performance on mobile devices.

[5] Z. Bobbitt, "Cohen's kappa statistic: Definition and example," 2022. Cohen's kappa measures the level of agreement between two raters or classifications; it is computed as $\kappa = (p_o - p_e)/(1 - p_e)$, where p_o is the observed agreement and p_e the probability of chance agreement.

[6] "F-score," Wikipedia, accessed 2025. The macro F1 score is the arithmetic mean of per-class F1 scores, giving equal weight to each class, while the micro F1 score is the harmonic mean of micro precision and recall.

[7] M. Shukla, H. Gupta and A. Sharma, "Sign Language Interpreter using Deep Learning," research square preprint, August 2025. This condensed paper evaluates the hackathon prototype, describing the motivation, data collection and pre-processing pipeline, CNN architecture and training procedure. It highlights unique contributions such as an evaluation framework incorporating additional metrics beyond simple accuracy and an ethical analysis addressing bias, privacy and cultural sensitivity. DOI : 10.21203/rs.3.rs-7465375/v1