Inclusive Few-Shot Isolated Sign Language Recognition via Spatio-Temporal SlowFast Prototypes

Anonymous ACL submission

Abstract

Sign language is a vital modality in human communication, yet current AI systems face significant challenges in recognizing it due to limited annotated data and high intra-class variability. In this work, we present a low-resource approach to isolated sign language recognition by framing it as a few-shot learning problem, using a prototypical network trained on a small support set. Our method utilizes a modified SlowFast convolutional architecture to extract rich spatio-temporal embeddings from sign videos, facilitating metric-based comparison between support-set exemplars and query clips. Unlike conventional models that require extensive training data, our approach generalizes to unseen sign classes using only a few labeled examples. We evaluate our model on the LSA64 dataset in a strict few-shot setting, achieving 88% accuracy on held-out classes, substantially outperforming baselines. This study highlights the potential of combining efficient video representations with metric learning to enable scalable, data-efficient sign language understanding. Our results advocate for future human-AI interaction systems that are inclusive and accessible, even in low-resource communication domains.

1 Introduction

Sign languages serve over 70 million deaf and hard-of-hearing individuals worldwide but exhibit considerable regional variation—e.g., American Sign Language and British Sign Language are mutually unintelligible among some 300 distinct languages (Rastgoo et al., 2021; Emmorey, 2023). This diversity hinders communication both between signers of different varieties and between signers and non-signers. Recent deep-learning methods for Sign Language Recognition (SLR) leverage RGB, skeletal, and multimodal inputs (Ahn et al., 2023; Alsulami et al., 2024; Papastratis et al., 2020) and address two sub-tasks: isolated (single-sign) and continuous (sequence) recognition (Zhou et al., 2021). Popular architectures include CNN–RNN hybrids, i.e., Convolutional Neural Networks (CNNs) with Recurrent Neural Networks (RNNs) (Min et al., 2021; Papastratis et al., 2020), Transformer-based models (Camgoz et al., 2020), and graph neural networks (de Amorim et al., 2019). The scarcity of annotated sign-language datasets hampers progress, motivating few-shot learning adaptations such as prototypical networks (Snell et al., 2017; Ferreira et al., 2022; Alsulami et al., 2024). In this work, we integrate the SlowFast network (Feichtenhofer et al., 2019), noted for its success in continuous SLR (Ahn et al., 2023), into a prototypical-learning framework and evaluate its effectiveness on the LSA64 dataset (Ronchetti et al., 2023) for few-shot recognition of novel sign classes under low-sample conditions.

2 Related Works

Sign language recognition (SLR) employs various methodological frameworks. Traditional methods combine CNNs with RNNs for Continuous SLR (CSLR) (Min et al., 2021; Papastratis et al., 2020), while graph-based approaches (de Amorim et al., 2019) are suitable for isolated SLR using skeletal data. Transformer-based methods (Camgoz et al., 2020; Ferreira et al., 2022; Alsulami et al., 2024) excel in both tasks by capturing temporal dependencies.

Feature extraction is essential for effective SLR. The SlowFast network (Feichtenhofer et al., 2019), designed for video recognition, has been adapted for SLR, enhancing Continuous SLR (Ahn et al., 2023) and proving effective for isolated SLR (Hassan et al., 2021). Our work uses SlowFast as the feature extractor in a few-shot learning approach.

Limited annotated data drives few-shot learning in SLR. Prototypical Networks have been applied

to small datasets (Ferreira et al., 2022), while recent studies integrate Transformers with few-shot frameworks (Ferreira et al., 2022; Boháek and Hrúz, 2023; Alsulami et al., 2024). Interestingly, (Alsulami et al., 2024) achieves strong results using skeletal data extracted from RGB videos, while (Sari et al., 2023) employs Shufflenet_V2 to perform few-shot sign language recognition in Indonesian. In contrast, (Bilge et al., 2023) explores zero-shot recognition using TSM, 3D CNN + BiLSTM, and BERT; its focus differs from few-shot learning but highlights related data scarcity solutions.

This work targets isolated SLR, leveraging SlowFast in a few-shot approach. Unlike prior studies (Alsulami et al., 2024; Sari et al., 2023), the use of RGB-based SlowFast features is novel, enhancing performance under data constraints.

3 Few-shot Learning

Few-shot learning (FFL) enables models to generalize to new categories from only a few examples per class (Snell et al., 2017). This is especially valuable in sign language recognition (SLR), where collecting large labeled datasets can be difficult or costly (Alsulami et al., 2024). Unlike traditional classifiers that require many samples and can only recognize seen classes, FSL embeds samples into a shared space where unseen classes can be handled by comparing new examples to a small support set.

FSL methods fall into two main groups: transfer-based (non-meta) and meta-learning, the latter including metric-based approaches such as Siamese, Matching, and Prototypical Networks (Parnami and Lee, 2022). These metric-based models learn an embedding so that examples cluster around class "prototypes," allowing for simple nearest-prototype classification for novel samples.

3.1 Prototypical Networks

Prototypical Networks assume each class is represented by the mean ("prototype") of its support embeddings (Snell et al., 2017). Training proceeds episodically: each episode samples an N-way k-shot support set and query set, where N represents the number of classes and k the number of examples per class. The network is optimized to minimize classification error across many such tasks (Parnami and Lee, 2022). At the test stage, a query is assigned to the class with the closest prototype under a chosen distance metric. Classification uses a softmax over these distances, and parameters are learned via negative log-likelihood.

3.2 SlowFast Network

The SlowFast network (Feichtenhofer et al., 2019) is a two-stream architecture designed for video recognition, featuring a Slow pathway for spatial semantics and a Fast pathway for temporal dynamics. Its proven success in sign language recognition, achieving a low Word Error Rate (WER) on benchmark datasets (Ahn et al., 2023), motivated its adoption for this prototypical approach to isolated sign language recognition. In this context, the Slow pathway captures spatial features, such as hand shapes, while the Fast pathway targets dynamic gestures, both of which are critical for accurate recognition (Ahn et al., 2023). Lateral connections fuse Fast features into the Slow stream, enriching spatial representations with temporal cues (Feichtenhofer et al., 2019). Modifications, such as the Bi-directional Feature Fusion module (Ahn et al., 2023), further boost performance by focusing on salient regions.

4 Methodology

4.1 Dataset

The LSA64 dataset (Ronchetti et al., 2023) consists of 3200 videos of 64 common Argentinian Sign Language signs (verbs and nouns), performed by 10 non-expert participants with 5 repetitions each. Recorded with fluorescent gloves to facilitate hand segmentation, it supports the creation of a sign dictionary and the training of an automatic recognition system.

4.2 Data processing

The data were loaded with a fixed seed for reproducibility. All videos were normalized to the range [0, 1], resized to 224 × 224 pixels (retaining all three RGB channels), and zero-padded within each batch to match the longest sequence, as sequences varied in length. The dataset was split 80:20 into 52 training and 12 validation classes, with validation samples strictly held out to avoid embedding adaptation (see Tab.1) to unseen samples. All split parameters and preprocessing code are documented in the accompanying repository[1].

[1]Code available at https://anonymous.4open.science/r/SlowFast_Prototypical_SLR-E54A

Split	# Samples	Sample IDs
Train	52	1, 2, 4, 5, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 19, 20, 21, 22, 23, 24, 26, 28, 29, 30, 31, 32, 33, 34, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 51, 52, 53, 55, 57, 58, 59, 60, 61, 62
Test	12	3, 6, 7, 18, 25, 27, 35, 50, 54, 56, 63, 64

Table 1: Dataset split into train and test sets.

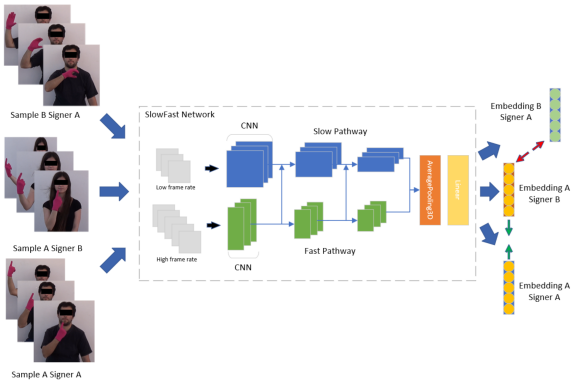

Figure 1: Illustration of the few-shot sign language recognition pipeline using a prototypical network with the SlowFast feature embedder. The graphic shows the processing of video samples and the training process, where embeddings are learned to cluster same-class samples and separate different-class samples, enabling classification based on proximity to class prototypes. For SlowFast architecture details, see (Fan et al., 2020; Feichtenhofer et al., 2019).

4.3 Training details

Training was performed for 30 epochs, each consisting of 100 episodes, using a custom episodic sampler (see 1) to enforce class balance. Due to GPU memory limitations, we fixed n-way=4, with three support and two query samples per class, all sampled at random within each episode. Reproducibility was ensured by seeding both PyTorch and Python's random libraries with 123 across training, testing, and evaluation. Optimization used the Euclidean-distance-based loss from (Snell et al., 2017), modified to accommodate video inputs.

5 Experiments

The experiments were conducted on a split dataset comprising not only unseen samples but also entire classes withheld during training (see Tab. 1). The model's performance was evaluated using n-way classification with n=5 and n=10, support sizes ranging from 1 to 10, and a fixed query size of 15, following similar methodology in (Ferreira et al., 2022). Classification accuracy, defined as the proportion of correctly classified query samples per episode, was averaged over 1000 episodes to ensure robust results across diverse signer samples. The results and settings are presented in Table 2.

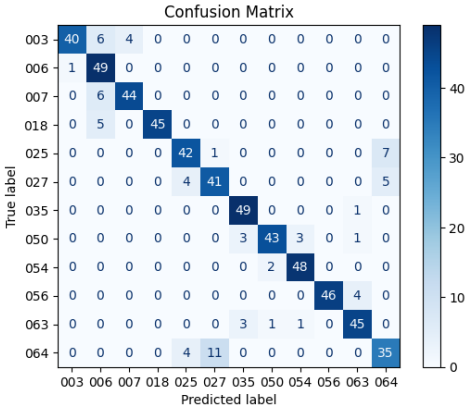

Figure 2: Confusion Matrix of Test Set Predictions.

n_way	Accuracy (k_support)		
	1-shot	5-shot	10-shot
5-way	80.0	90.7	92.0
10-way	66.0	81.9	84.9

Table 2: Average Classification Accuracy for Different Few-Shot Configurations ($q_{\text{query}} = 15$, $n_{\text{episodes}} = 1000$; values in [%])

To further assess generalization, we extracted embeddings for all test samples and computed class prototypes as the mean embedding of each class training examples. Each test embedding was then assigned to its nearest prototype via Euclidean distance. Of 600 test samples, 73 were misclassified (12% error; 88% accuracy). A confusion matrix (Fig. 2) provides a detailed breakdown of classification results, illustrating both correct predictions and errors across all test classes.

5.1 UMAP visualization

To visualize the embeddings, we employed the UMAP algorithm (McInnes et al., 2020), a dimension reduction technique that constructs a fuzzy topological model of high-dimensional data using local manifold approximations and fuzzy simplicial sets. It assumes data points are uniformly distributed on a locally connected Riemannian manifold with a constant metric, optimizing a low-dimensional representation by minimizing cross-entropy between high- and low-dimensional models. Unlike t-SNE, UMAP preserves both local

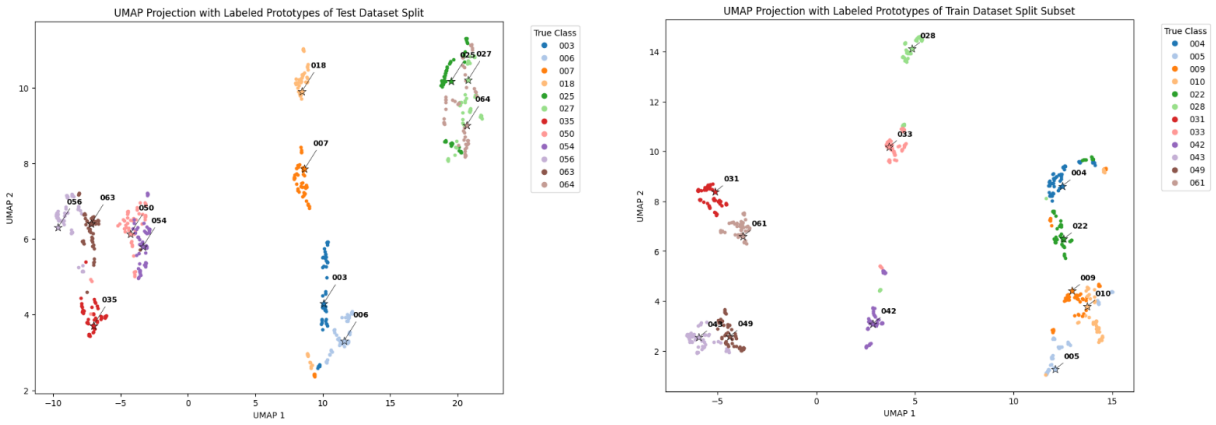

Figure 3: UMAP visualization of embeddings from the test (left) and training (right) dataset. The x-axis (UMAP 1) and y-axis (UMAP 2) are the first and second components of the 128→2 UMAP projection. Embeddings from the test and 12 randomly selected training classes are shown, with class prototypes indicated by star-shaped markers to illustrate clustering behavior in the low-dimensional space.

neighborhoods and global structures efficiently (McInnes et al., 2020), making it ideal for few-shot learning where sign language samples of the same class must cluster closely. We applied UMAP to test embeddings to assess generalization and to compare embeddings from 12 randomly selected training classes, with results shown in Fig. 3.

6 Results and Discussion

The proposed slow-fast meta-based network achieved an accuracy of 88% on the test split dataset, effectively handling unseen data. Predictions were made by calculating the distance between a sample's embedding and class prototypes, formed by averaging embeddings within each class. This 88% accuracy, with a 12% error rate, aligns with prototypical learning goals, highlighting the network's (Feichtenhofer et al., 2019) reliability as a feature extractor for isolated SLR.

Visualization of test dataset embeddings using UMAP (McInnes et al., 2020) revealed that some classes formed distinct clusters, while others, such as "025", "027", and "064", overlapped (see Fig. 3). These overlaps are reflected in the confusion matrix, where 11 of 50 samples from class "064" are misclassified as "027" (Fig. 2). While this agreement between UMAP and the confusion matrix reinforces the validity of our embedding space, we note that the 2-D projection may oversimplify the original 128-dimensional relationships and thus could either magnify or obscure certain class proximities. On the selected training subset, UMAP visualization shows good class separation, yet some samples are near other class prototypes, suggesting potential misclassification or similarities the model struggles to disentangle, possibly due to unseen training samples.

Overall, the model demonstrates strong performance in few-shot sign language recognition, with effective feature extraction, though challenges remain in separating certain classes.

7 Conclusion

This study demonstrated that the SlowFast network, combined with prototypical learning, achieves 88% accuracy in isolated sign language recognition (SLR), effectively extracting features from unseen data. While the model excels in clustering most classes, it struggles with overlapping ones. Future research should focus on validating the model on larger, glove-free datasets and exploring alternative modalities, such as skeletal landmarks, to enhance robustness and reduce reliance on artificial markers. Additionally, investigating margin-based loss functions (e.g. multi-way contrastive loss (Parnami and Lee, 2022)) and hyperparameter optimization could improve class separability and address sampling biases.

Limitations

Despite its demonstrated efficacy, the proposed prototypical network exhibits notable difficulties in discriminating among certain sign language gestures. UMAP projections reveal overlapping clusters for classes such as "025," "027," and "064," and the confusion matrix confirms systematic misclassifications—for example, instances of "064" are frequently labeled as "027." These observations

suggest that, while the prototypical loss effectively draws semantically similar embeddings into close proximity, it does not always enforce sufficient inter-class separation. Moreover, the few-shot sampling strategy employed during training may intensify this issue: by randomly selecting only a subset of classes per episode, some samples may never contribute to prototype refinement, and increasing episodes or epochs risks overfitting to the limited training set.

Further limitation arises from the dataset itself. Although the collection encompasses 64 gesture classes, our experiments utilize only 12 randomly selected categories (see Tab. 1), which constrains the breadth of the evaluation. In addition, signers wore fluorescent gloves to facilitate hand segmentation (Ronchetti et al., 2023), enhancing recognition accuracy but undermining real-world applicability, as everyday users are unlikely to wear such equipment.

Generalization to out-of-distribution visual data represents a further concern. By focusing exclusively on raw RGB inputs, the network may rely on background or context cues that do not transfer across environments.

Finally, our reliance on two-dimensional UMAP visualizations to assess embedding separability introduces a methodological constraint. Reducing 128-dimensional feature vectors to a planar representation inevitably distorts inter-point distances and may obscure the true structure of the embedding space, suggesting a need for more robust evaluation methods in future studies.

References

Junseok Ahn, Youngjoon Jang, and Joon Son Chung. 2023. Slowfast network for continuous sign language recognition. *Preprint*, arXiv:2309.12304.

Amjad Alsulami, KHAWLAH BAJBAA, Issam H. Laradji, and Hamzah Luqman. 2024. Few-shot Learning for Sign Language Recognition with Embedding Propagation. In *ArXiv*.

Yunus Can Bilge, Ramazan Gokberk Cinbis, and Nazli Ikizler-Cinbis. 2023. Towards zero-shot sign language recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(1):1217–1232.

Matyás Boháek and M. Hrúz. 2023. Learning from what is already out there: Few-shot sign language recognition with online dictionaries. *2023 IEEE 17th International Conference on Automatic Face and Gesture Recognition (FG)*, pages 1–6.

Necati Cihan Camgoz, Oscar Koller, Simon Hadfield, and Richard Bowden. 2020. Multi-channel transformers for multi-articulatory sign language translation. *Preprint*, arXiv:2009.00299.

Cleison Correia de Amorim, David Macêdo, and Cleber Zanchettin. 2019. *Spatial-Temporal Graph Convolutional Networks for Sign Language Recognition*, pages 646–657. Springer International Publishing.

Karen Emmorey. 2023. Ten things you should know about sign languages. *Current Directions in Psychological Science*, 32:096372142311730.

Haoqi Fan, Yanghao Li, Bo Xiong, Wan-Yen Lo, and Christoph Feichtenhofer. 2020. Pyslowfast. `https://github.com/facebookresearch/slowfast`.

Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. 2019. Slowfast networks for video recognition. *Preprint*, arXiv:1812.03982.

Silvan Ferreira, Esdras Costa, Márcio Dahia, and Jampierre Rocha. 2022. A transformer-based contrastive learning approach for few-shot sign language recognition. *Preprint*, arXiv:2204.02803.

Ahmed Hassan, Ahmed Elgabry, and Elsayed Hemayed. 2021. Enhanced dynamic sign language recognition using slowfast networks. In *2021 17th International Computer Engineering Conference (ICENCO)*, pages 124–128.

Leland McInnes, John Healy, and James Melville. 2020. Umap: Uniform manifold approximation and projection for dimension reduction. *Preprint*, arXiv:1802.03426.

Yuecong Min, Aiming Hao, Xiujuan Chai, and Xilin Chen. 2021. Visual alignment constraint for continuous sign language recognition. *Preprint*, arXiv:2104.02330.

Ilias Papastratis, Kosmas Dimitropoulos, Dimitrios Konstantinidis, and Petros Daras. 2020. Continuous sign language recognition through cross-modal alignment of video and text embeddings in a joint-latent space. *IEEE Access*, 8:91170–91180.

Archit Parnami and Minwoo Lee. 2022. Learning from few examples: A summary of approaches to few-shot learning. *Preprint*, arXiv:2203.04291.

Razieh Rastgoo, Kourosh Kiani, and Sergio Escalera. 2021. Sign language recognition: A deep survey. *Expert Systems with Applications*, 164:113794.

Franco Ronchetti, Facundo Manuel Quiroga, César Estrebou, Laura Lanzarini, and Alejandro Rosete. 2023. Lsa64: An argentinian sign language dataset. *Preprint*, arXiv:2310.17429.

Irma Permata Sari, Fuad Mumtas, Z.E. Ferdi Fauzan Putra, Ressy Dwitias Sari, Ati Zaidiah, and Mayanda Mega Snatoni. 2023. Enhanced few-shot learning for indonesian sign language with prototypical networks approach. In *2023 International*

Conference on Informatics, Multimedia, Cyber and Informations System (ICIMCIS), pages 278–283.

Jake Snell, Kevin Swersky, and Richard S. Zemel. 2017. Prototypical networks for few-shot learning. *Preprint*, arXiv:1703.05175.

Hao Zhou, Wengang Zhou, Weizhen Qi, Junfu Pu, and Houqiang Li. 2021. Improving sign language translation with monolingual data by sign back-translation. *Preprint*, arXiv:2105.12397.

A Implementation details

The SlowFast model was implemented following (Feichtenhofer et al., 2019) using the official SlowFast Meta codebase (Fan et al., 2020). To match the Slow Pathway's lower frame rate, we selected every eighth frame from each video sample. The architecture was modified by reducing the number of residual stages to three and the number of FuseFastToSlow blocks to two, with the embedding dimension (WIDTH_PER_GROUP (Fan et al., 2020)) set to 32. The prediction head applies 3D average pooling and a linear layer, producing a 128-dimensional normalized embedding for each video. Training was conducted using PyTorch with the Adam optimizer and a learning rate of 0.001. The SlowFast pipeline is illustrated in Fig. 1. For further details on the implementation and additional experiments, see the code repository1.