IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 8 July 2024, accepted 26 July 2024, date of publication 1 August 2024, date of current version 14 August 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3436671

RESEARCH ARTICLE

Hand and Pose-Based Feature Selection for Zero-Shot Sign Language Recognition

GIRAY SERCAN ÖZCAN[1], YUNUS CAN BILGE[2], AND EMRE SÜMER[1]
[1]Department of Computer Engineering, Başkent University, 06790 Ankara, Türkiye
[2]Image and Video Processing Group, Havelsan, 06510 Ankara, Türkiye

Corresponding author: Giray Sercan Özcan (gozcan@baskent.edu.tr)

ABSTRACT Sign language functions as an indispensable interaction method for a certain portion of people in society, offering a unique way of communication. A significant challenge in advancing towards this objective is the difficulty in obtaining suitable training data for each sign in supervised learning. This challenge comes from the complex process of labeling signs and the limited number of skilled people available to do this job. This work introduces a new approach to the problem of Zero-Shot Sign Language Recognition (ZSSLR). We basically utilize and model hand and landmark data streams extracted from the body of the signer. Based on these extracted and modeled features, we employ a data grading approach to facilitate visual embedding with the self-attention mechanism. We utilize textual sign description features along with visual embedding in the Zero-Shot Learning (ZSL) settings. We assess the efficacy of our methodology in two of the suggested ZSL benchmarks.

INDEX TERMS Sign language recognition, zero-shot sign language recognition, zero-shot learning.

I. INTRODUCTION

Sign language is an essential means of communication for people who have hearing impairments, and the advancement of Sign Language Recognition (SLR) systems significantly aids in facilitating communication within this community, thereby having a wide social impact. Recently, several approaches have emerged for recognizing signs [1], [2] in the sense of Zero-Shot Sign Language Recognition (ZSSLR). Unlike classical SLR approaches, which mainly focus on recognizing samples at test time from the sign categories that are already observed throughout training. In ZSSLR [3], as illustrated in Figure 1, the objective is to identify classes that were not seen throughout the training phase. The process entails transforming examples from the visual feature space into the semantic feature space through multiplication with a compatibility matrix, a key step in Zero-Shot Learning (ZSL) recognition. Subsequently, the system identifies the class that is closest to the transformed example as the best match. In that sense, we propose an approach for ZSSLR and demonstrate the effectiveness of it within its benchmarks.

The associate editor coordinating the review of this manuscript and approving it for publication was Venkata Rajesh Pamula.

Classical supervised SLR systems typically demand substantial quantities of annotated data ([4], [5], [6], [7], [8]) for each sign class. The challenge in extensive datasets lies in the requirement to have numerous examples for every sign word. Given the multitude of sign languages globally, ZSSLR was introduced to tackle the challenges of insufficient data availability and the complexities involved in its annotation [3]. Given that there are over 100 sign languages worldwide, each with numerous dialects [9] and containing more than 3000 words [10], [11], it is evident that annotating each sign language is a challenging task, resulting in an insufficient data issue in this field. We aim to propose a vision-based solution for ZSSLR that recognizes sign categories lacking training examples.

We aim to recognize unseen classes based on their textual descriptions and our proposed visual embedding methodology. The visual component of ours focuses on modeling the temporal and spatial patterns present in sign language. Recognizing spatial and temporal patterns is crucial because spatial dependencies, like the position of the hand, and temporal dependencies, like its movements, can alter the meaning of a sign. To achieve this, we explore Residual Networks Next (ResNeXt) [12], Spatial-Temporal

© 2024 The Authors. This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License.
For more information, see https://creativecommons.org/licenses/by-nc-nd/4.0/

Graph Convolutional Network (ST-GCN) [13], Long Short-Term Memory (LSTM), and the recently proposed Multiscale Vision Transformers (MViTv2) [14] modules. ResNeXt architecture, pre-trained on the Hand Gesture Recognition Image Dataset (HaGRID) [15] dataset, focuses exclusively on recognizing hand patch images, highlighting its importance for spatial pattern recognition. The ST-GCN model operates solely on video landmark data, underscoring its significance for both spatial and temporal modeling. MViTv2, designed as an action recognition module, is particularly crucial for advanced spatial and temporal analysis.

The body and hand regions are especially important for SLR because even the slightest change in movements made in these areas can completely alter the meaning of a sign. Therefore, a part of our framework specifically attends to the body and hand segments extracted from the videos. Features have been extracted from hand and body streams using MViTv2 and ResNeXt. Techniques such as Bidirectional Long Short-Term Memory (Bi-LSTM), 1-Dimensional Convolutional Neural Network (1D-CNN), and self-attention have been utilized to make these features compatible with the system. Landmark data have been processed using Bi-LSTM and ST-GCN. Given the large volume of data from various sources, data grading has been implemented to manage and prioritize related information effectively. Semantic information has been extracted from textual class definitions using the Contrastive Language-Image Pretraining (CLIP) [16] architecture, and the visual embedding has been mapped to its closest semantic class. The reason for this mapping is to show that, based on the extracted information, it most closely resembles that class. Upon that, we aim to map the learned visual representation to its textual description. We have conducted thorough evaluations of this approach using the previously proposed ASL-Text [3] and MS-ZSSLR-W/C [1] ZSSLR benchmark datasets, demonstrating its effectiveness and benefits. The reason for using only these datasets is that, to the best of our knowledge, they are the only publicly accessible datasets in this field available for comparison with other studies.

In brief, the primary novelties are outlined as follows:

- To the best of our knowledge, this study is the first to utilize the powerful feature extraction capabilities of MViTv2 and ResNeXt in addressing the ZSSLR problem domain, thereby establishing a novel approach in this area.
- This research is pioneering in its application of the ST-GCN method within the context of ZSSLR, highlighting the potential of this technique in capturing complex spatial and temporal relationships.
- We are also the first to leverage the CLIP [16] architecture for the extraction of textual vectors that are critical for effective ZSSLR, demonstrating the versatility and effectiveness of this model in understanding and aligning textual information.
- Finally, this study is the first to integrate these advanced methods, combining MViTv2, ResNeXt, ST-GCN, and CLIP to achieve significantly improved results in ZSSLR, thereby setting a new benchmark for future research in this field.

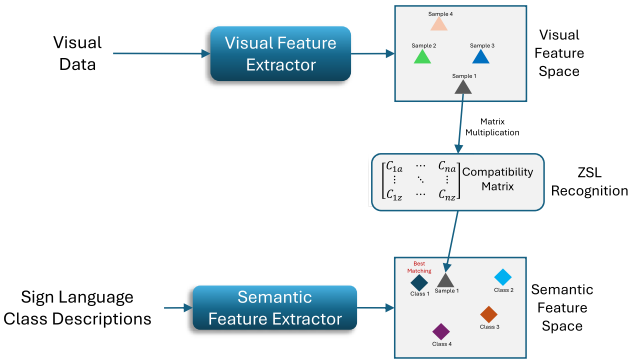


FIGURE 1. General ZSSLR approach consists of three main stages: Visual Embedding, Semantic Embedding, and ZSL Recognition.

The structure of the paper is organized as follows: Section II reviews the relevant literature. Our approach to address the problem of ZSSLR is detailed in Section III. Section IV covers details on the implementation methodologies, a description of the datasets along with their related features, the presentation of the obtained results, and an extensive ablation study to evaluate the contributions of various components. Finally, we conclude our findings in Section V.

II. LITERATURE REVIEW

A. SIGN LANGUAGE RECOGNITION

SLR contains two main categories: Isolated SLR (ISLR) [17] and Continuous SLR (CSLR) [18]. ISLR focuses on recognizing distinct signs or gestures presented separately, which simplifies the recognition process by reducing variability and context dependence. In contrast, CSLR deals with the recognition of sign language in natural, flowing sequences. This involves identifying signs not only in isolation but also considering their context with preceding and succeeding signs, making it more complex due to the variability of sign transitions and the influence of non-manual features.

Studies in ISLR and CSLR can be divided into two categories: those using manually extracted features ([19], [20], [21], [22], [23], [24], [25]) and those employing deep learning architectures ([26], [27], [28], [29]). Research involving manually extracted features typically utilized techniques such as Hidden Markov Model (HMM), and Hidden Conditional Random Fields (HCRF). One of the first real-time studies, conducted by Starner et al. [30], performed CSLR using an HMM-based method with a wearable camera. Among recent studies, there are those that operate under weak supervision [31], [32], [33].

More recent works like [34] propose a framework for SLR that combines multiple modalities such as skeletal information, RGB, and depth data. They introduce the Sign Language Graph Convolution Network (SLGCN) for

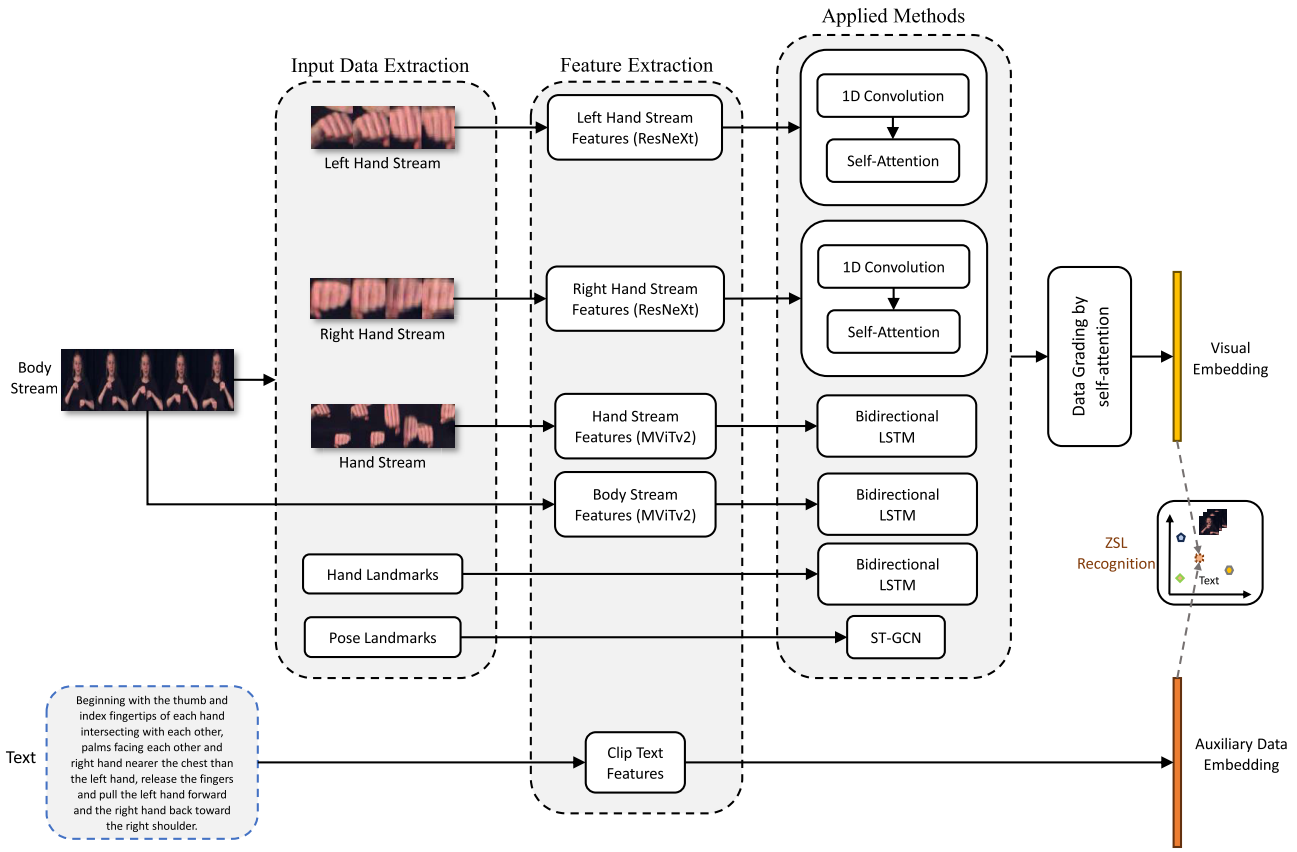


FIGURE 2. The proposed ZSSLR approach is executed in three stages: Visual embedding, Auxiliary embedding, and ZSL recognition approach. Upon acquiring the visual and auxiliary embeddings, we proceeded with the use of the ZSL recognition technique.

focusing on modeling the dynamics involved in sign language and the Separable Spatial-Temporal Convolution Network (SSTCN) for the spatial and temporal components of sign language gestures. Lee et al. [35] present a prototype of a game-based application for learning sign language, integrating a real-time sign recognition system utilizing the Leap Motion Controller. They utilize a combination of Long Short-Term Memory Recurrent Neural Networks (LSTM-RNN) and the k-Nearest Neighbor (kNN) algorithm for SLR. Hu et al. [36] suggest an ISLR framework that is sensitive to hand models by using hand poses and meshes as intermediate representations to enhance SLR accuracy. They use a Graph Convolution Network (GCN) in the inference module to enhance the spatiotemporal representation of the hand-pose sequence. Hu et al. [37] introduce an approach for CSLR by explicitly modeling and emphasizing body trajectories. They utilize a 2-Dimensional Convolutional Neural Network (2D-CNN) feature extractor followed by a CNN-based method for temporal modeling. Zhou et al. [38] introduce a Spatial-Temporal Multi-Cue (STMC) approach for Sign Language Translation (SLT) and SLR that effectively harnesses the complexity and richness of visual cues in sign language. They utilize a Bi-LSTM decoder with a Connectionist Temporal Classification (CTC) decoder for SLR. Min et al. [8] propose a method for CSLR that enhances the feature extractor's training by addressing the issue of overfitting through a Visual Alignment Constraint (VAC). The developed method includes an alignment module and feature extractor, with the addition of an auxiliary classifier for implementing VAC. Tunga et al. [39] introduce an architecture for ISLR that combines GCN and Bidirectional Encoder Representations from Transformers (BERT) [40]. They leverage the strengths of GCN for spatial modeling and BERT for temporal modeling. Another multi-scale work that utilizes GCN [41] captures the dynamic and complex nature of sign language. They introduce a Multi-Scale Spatial-Temporal Graph Convolutional Operator (MS-G3D). Wei and Chen [42] introduce a technique for multilingual CSLR by leveraging cross-lingual sign languages. They initially trained the ISLR model using constructed dictionaries for both the target and auxiliary sign languages. Then, the CSLR model is trained using both language data and mapped labels. Boháček and Hrúz [43] introduce a method for ISLR using a pose-based Transformer model, showcasing the potential of normalization and augmentation strategies in improving SLR. Sreemathy et al. [44] propose using Histogram-Oriented Gradient (HOG) for feature extraction and deep neural networks for classification. Kothadiya et al. [45] present a SLR system using Vision Transformer architecture, showcasing the Transformer model's adaptability

and effectiveness outside its traditional Natural Language Processing (NLP) domain. Shin et al. [46] integrate CNN and Transformer models. This allows for efficient capture of both local and global dependencies in sign language images. Bora et al. [47] integrate MediaPipe's hand-tracking functionality and feed-forward neural network for classifying gestures.

B. ZERO-SHOT LEARNING

After Larochelle et al. [48] laid the groundwork for the ZSL field, the first studies in this area were conducted by Palatucci et al. [49] and Lampert et al. [50]. These foundational works introduced an approach where attributes shared between observed and unobserved classes are used to transfer knowledge to those unseen classes. After these studies, various classification research has been conducted within the ZSL domain, employing methodologies optimized for diverse objectives ([51], [52], [53], [54], [55]).

More recent works such as [56] have utilized GCN to model label correlations, leveraging an attention mechanism for feature extraction. Chen et al. [57] introduce an attribute-guided Transformer, refining visual features and learning attribute localization for discriminative visual embedding representations. Roy et al. [58] present an approach to further enhance ZSL by generating commonsense embeddings of category names through a GCN-based autoencoder trained on a commonsense knowledge graph. Another work [59] utilizes GCN for Generalized ZSL (GZSL), where the goal is to recognize the observed and unobserved classes. They tackled the bias issue, where models often misclassify unobserved samples as seen classes by utilizing weighted kNN classification in both original and latent spaces to separate seen and unseen domains accurately. Gupta et al. [60] suggest a generative technique for addressing the ZSL problem by introducing a model that generates semantically coherent visual multi-label features from their associated class attribute embeddings. Cheng et al. [61] propose a method for ZSL that uses both dynamic and static routing mechanisms within a transformer architecture to minimize the semantic gap between semantic features and visual attributes. Gowda [62] address the challenge of redundant synthetic feature generation in GZSL by introducing a reinforcement learning-based method for strategic feature selection. Guo et al. [63] introduce a new perspective on tackling ZSL by leveraging the inherent graph structure within visual samples to more accurately model the element-semantics correlations. Liu et al. [64] introduce an approach for GZSL to address the challenges of semantic ambiguity and improve knowledge transferability between seen and unseen classes.

C. ZERO-SHOT SIGN LANGUAGE RECOGNITION

Bilge et al. [3] were pioneers in defining the ZSSLR problem and formulating an approach to address it. They established the ASL-Text dataset by merging videos of sign language with corresponding textual descriptions found in sign language dictionaries, a resource that is also employed in this current study. The dataset includes annotations for both hand and body motions. They evaluated different ZSL techniques such as Embarrassingly Simple Zero-Shot Learning (ESZSL) [53], Semantic Autoencoder (SAE) [65], and Logistic Label Embedding (LLE) [3] on these categorized streams, leading to varying outcomes. In later research [1], the same authors augmented the ASL-Text dataset by incorporating binary feature matrices alongside the textual descriptions in the ZSSLR framework's second segment. Moreover, they developed two benchmark datasets named MS-ZSSLR-W and MS-ZSSLR-C, to which they applied a novel shift-based CNN [66] method in addition to employing 3-Dimensional Convolutional Neural Network (3D-CNN) and LSTM, which were used in their earlier research.

Nihal et al. [67] introduce two techniques for SLR alphabet recognition, leveraging both ZSL and transfer learning to accurately identify seen and unseen sign alphabets. It utilizes pre-trained DenseNet201 for feature extraction and Linear Discriminant Analysis (LDA) for classification in the transfer learning approach. For ZSL, it proposes new semantic descriptors dedicated to sign language. Wu et al. [2] introduce a GZSL system for hand gesture recognition, utilizing a prototype-based detector and a zero-shot label predictor to effectively recognize both seen and unseen hand gestures. It leverages a Leap Motion Controller for capturing gesture sequences and applies Bi-LSTM and a Semantic Autoencoder (SAE) for feature extraction and classification. Yin et al. [68] present a framework for Multilingual Sign Language Translation (MSLT), leveraging dynamic neural network routing to handle multiple sign language and spoken language pairs efficiently. By introducing novel inter-layer and intra-layer language-specific routing mechanisms, the model dynamically adjusts parameter sharing and data flow, significantly enhancing translation performance across diverse language combinations. They also include zero-shot translation results.

D. ZERO-SHOT SIGN LANGUAGE RECOGNITION DATASETS

In addition to the publicly accessible ASL-Text [3] and MS-ZSSLR-W/C [1] datasets introduced by Bilge et al., there are other datasets available; however, these either pertain to different subtopics or are not publicly accessible. For instance, in their study, Nihal et al. [67] aimed to recognize sign language letters, unlike our work, and their datasets contain sign language letters rather than words. Wu et al. [2] attempted to recognize hand gestures using the ZSL method, but their dataset is not publicly available. Yin et al. [68] tried to perform sign language translation in their study, and the dataset they created includes hand movements in multiple sign languages. This task is more challenging and requires different architectures from ours because the movements vary across different sign languages. Rastgoo et al. [69],

[70] utilized SLR datasets in their studies. However, unlike ASL-Text and MS-ZSSLR-W/C, their datasets do not contain definitions of movements extracted from sign language dictionaries for each class; instead, they convert class labels into vectors and perform ZSL. While the field of ZSSLR is relatively new, there are numerous studies, but there are only a few publicly accessible datasets that can be evaluated in comparison with other studies.

III. PROPOSED APPROACH

Our ZSSLR approach, in which we aim to improve performance by enhancing visual and textual representations, is explained in this section. As seen in Figure 2, our main architecture consists of three stages: visual embedding, auxiliary embedding, and the ZSL approach. Detailed explanations corresponding to these three factors are provided in the following paragraphs.

A. HAND AND POSE BASED FEATURE REPRESENTATION

Our objective is to create a sophisticated module designed for acquiring discriminative feature representation. To accomplish this goal, we employ a multi-stream approach that integrates data from various sources: the body stream, left and right hand streams, an aggregated hand stream encompassing both hands, as well as pose and hand landmarks. This comprehensive strategy ensures a rich, nuanced understanding of the data, enabling more effective feature representation.

We employed the MViTv2 [14] method to extract characteristics from the body and the combined hand streams due to its recent *state-of-the-art* (SOTA) performance across multiple benchmarks. We think that extracting features with such a model, which has been successful in multiple benchmarks, will provide a better visual representation. Following this extraction, we applied the Bi-LSTM model to boost the temporal dynamics of these features. For the individual left and right hand streams, we leveraged the ResNeXt [12] architecture, specifically pre-trained on the HaGRID [15], to extract distinct characteristics. The reason for utilizing ResNeXt is that we believe the position and movements of the hand can be better modeled with an architecture that has been pre-trained with hand images only and aims to classify hand gestures. Because ResNeXt architecture was pre-trained on hand images, features have been extracted from each frame of the video and stacked. These stacked images were then processed through 1D-CNN and self-attention layers to reduce computational complexity and select the most relevant features, respectively.

The pose landmarks underwent processing with the ST-GCN [13] model, aimed at capturing spatial-temporal relationships, whereas the hand landmarks were further refined using the Bi-LSTM model to encode temporal information. ST-GCN builds a spatial-temporal graph using skeleton sequences and applies graph convolutional operations to capture dynamic human movements. This approach enables our model to learn both the temporal dynamics of human joints and their spatial configurations.

B. HAND AND POSE BASED FEATURE SELECTION

The applied feature selection incorporates the hand and pose features that are likely to achieve the highest performance in the model. We use a combination of 1D-CNN and self-attention for processing the ResNeXt [12] features. Given λ is the size of the kernel and c_{in} is the input channel id, 1D-CNN output Y at the output channel j for a given position t, $y_j[t]$ can be calculated as:

$$y_j[t] = \sigma(b_j + \sum_{i=0}^{c_{in}-1} \sum_{m=0}^{\lambda-1} W_{j,i,m} \cdot E_i[t+m]) \tag{1}$$

where σ stands for the activation function, b_j is the bias term for output channel j, $W_{j,i,m}$ is the weights of the kernel at output channel j, input channel i and kernel position m. $E_i[t+m]$ is the ResNeXt feature matrix at channel i and position $t+m$.

We calculated self-attention query ϕ, key κ and value ψ from 1D-CNN output Y as:

$$\phi = YW_\phi, \kappa = YW_\kappa, \psi = YW_\psi \tag{2}$$

where W_ϕ, W_κ and W_ψ are the learnable projection matrices for query, key and value, respectively. We calculate the self-attention output O as follows:

$$O = softmax\left(\frac{\phi\kappa^T}{\sqrt{d_\kappa}}\right)\psi \tag{3}$$

where d_κ is the dimension of key vectors and κ^T is the transpose of κ weight matrice. Function *softmax* is applied for converting scores into probabilities that sum up to 1. O_r and O_l output matrices were extracted from E_r right hand, and E_l left hand ResNeXt [12] features, respectively.

Bi-LSTM processes data in both forward and reverse directions. This dual mechanism allows our network to capture context from both past and future states, enhancing its ability to learn spatial and temporal sequence dependencies. Bi-LSTM is applied to MViTv2 [14] body and hand features, resulting in B_b and B_h, respectively, which were extracted with stride 4 and stacked. Additionally, Bi-LSTM is employed to extract meaningful information from hand landmarks, yielding B_{Hl}.

ST-GCN [13] were used for capturing spatial and temporal motifs from landmark data. First, we construct the graph $G = (V, E)$ where V is a set of vertices, E is a set of edges, and we repeat this process for each time frame for temporal data. In ST-GCN, three methods are utilized to determine the order of vertex multiplications, which involves subsets of vertices named layers. For calculating the spatial-temporal graph convolution $\mu^{(l+1)}$ at label $(l+1)$:

$$\mu^{(l+1)} = \sum_{n=1}^{N} \Theta_n^{(l)} \left(D_n^{-\frac{1}{2}} A_n D_n^{-\frac{1}{2}}\right) \mu^{(l)} \tag{4}$$

where $\mu^{(l)}$ is the feature matrix at layer l, A_n is the adjacency matrix between vertices representing different types of relationships in the layer l, D_n is the diagonal degree

matrix corresponding to A_n, $\Theta_n^{(l)}$ is the weights at layer l for the n-th type connection, N is the total number of subgraphs considered.

C. DATA GRADING METHODOLOGY

Data grading has been used because features have been extracted using different methods from a large number of various data types, such as landmark and video data. This enables the model to select the most effective features. Similarly to Equation 3, we use self-attention with slight difference. First, the features coming from various methods have been concatenated to form Z:

$$Z = \text{Combine}(O_r, O_l, B_b, B_h, B_{Hl}, \mu) \tag{5}$$

where O_r and O_l stands for 1D-CNN, self-attention output from right and left hand ResNeXt [12] features, respectively. B_b and B_h denote the Bi-LSTM output from the body and hand MViTv2 [14] features, while B_{Hl} and μ represent the Bi-LSTM and ST-GCN [13] output from the hand and pose landmarks, respectively. Consequently, $Z \in \mathbb{R}^{p\times d}$, where p signifies the number of methods and d denotes the number of features.

We introduced learnable weight parameters W_ϕ, W_κ and W_ψ to transport input features into query ϕ, key κ, and value ψ, which is required for self-attention. These matrices map d-dimensional feature vectors into d'-dimensional and can be calculated as in Equation 2 with Y replaced by concatenated feature matrix Z, i.e. as $\phi = ZW_\phi, \kappa = ZW_\kappa, \psi = ZW_\psi$. Here, $\phi, \kappa, \psi \in \mathbb{R}^{p\times d'}$. Attention weights were computed using the softmax function to normalize the dot products of ϕ and κ, indicating the importance of each feature:

$$A = \text{softmax}\left(\frac{\phi\kappa^T}{\sqrt{d'}}\right) \tag{6}$$

where $A \in \mathbb{R}^{p\times p}$ is the attention matrix, with each element a_{ij} indicating the influence of feature j on feature i. Then, the value matrices are multiplied with attention weights:

$$\nu = A\psi \tag{7}$$

where $\nu \in \mathbb{R}^{p\times d'}$ is the matrix of weighted features after attention has been applied and, consequently, our visual embedding matrix. This is our visual embedding learning methodology.

D. SIGN DESCRIPTION MODELING

Language embedding vector matrix for all classes, denoted as L, that capture the context are derived from written descriptions of visual signs using the sophisticated language model CLIP [16], which is detailed in Figure 3. At its core, CLIP functions as a model capable of ZSL, leveraging natural language to link with visual concepts it has learned previously or to describe new ones. This capability facilitates its application to a variety of tasks without prior explicit training on them, marking a significant advancement over traditional models like word2vec [71], GloVe [72], and BERT [40]. Unlike these models, CLIP's distinct advantage lies in its context-sensitive embeddings and its flexibility. It is also worth mentioning that the training of these embeddings, along with the function of compatibility, is conducted in a unified end-to-end process.

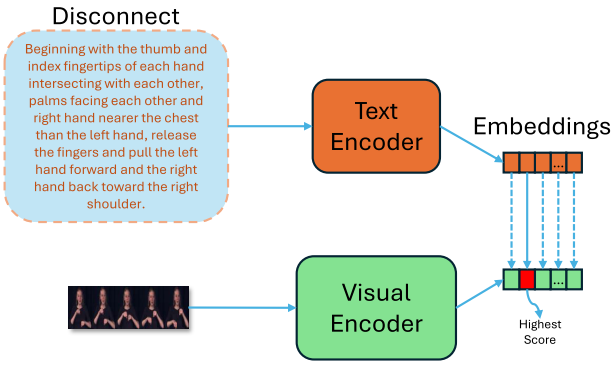


FIGURE 3. CLIP learns visual concepts from textual descriptions, empowering it to execute a range of visual tasks without explicit training.

CLIP uses a dual-encoder architecture: visual and text encoder. These encoders transform their respective inputs into the same embedding space where the compatibility of image-text pairs is measured. For the visual encoder, modified versions of standard architectures such as Residual Networks (ResNet) [73] and Vision Transformer (ViT) [74] were utilized. For the text encoder, the Transformer [75] architecture was employed.

E. ZERO-SHOT SIGN RECOGNITION

We employ a label embedding-based approach [76], [77] as a ZSL component seen in Figure 2. Namely, LLE [1] focuses on learning a compatibility matrix that aligns video feature embeddings with semantic class embeddings, such as textual descriptions. It utilizes a logistic model to predict the likelihood of a particular video belonging to a class, applying a cross-entropy loss function for optimization. This approach is further enhanced with regularization to prevent overfitting, promoting the acquisition of generalized representations that help to bridge the disparity between observed and unobserved sign classes.

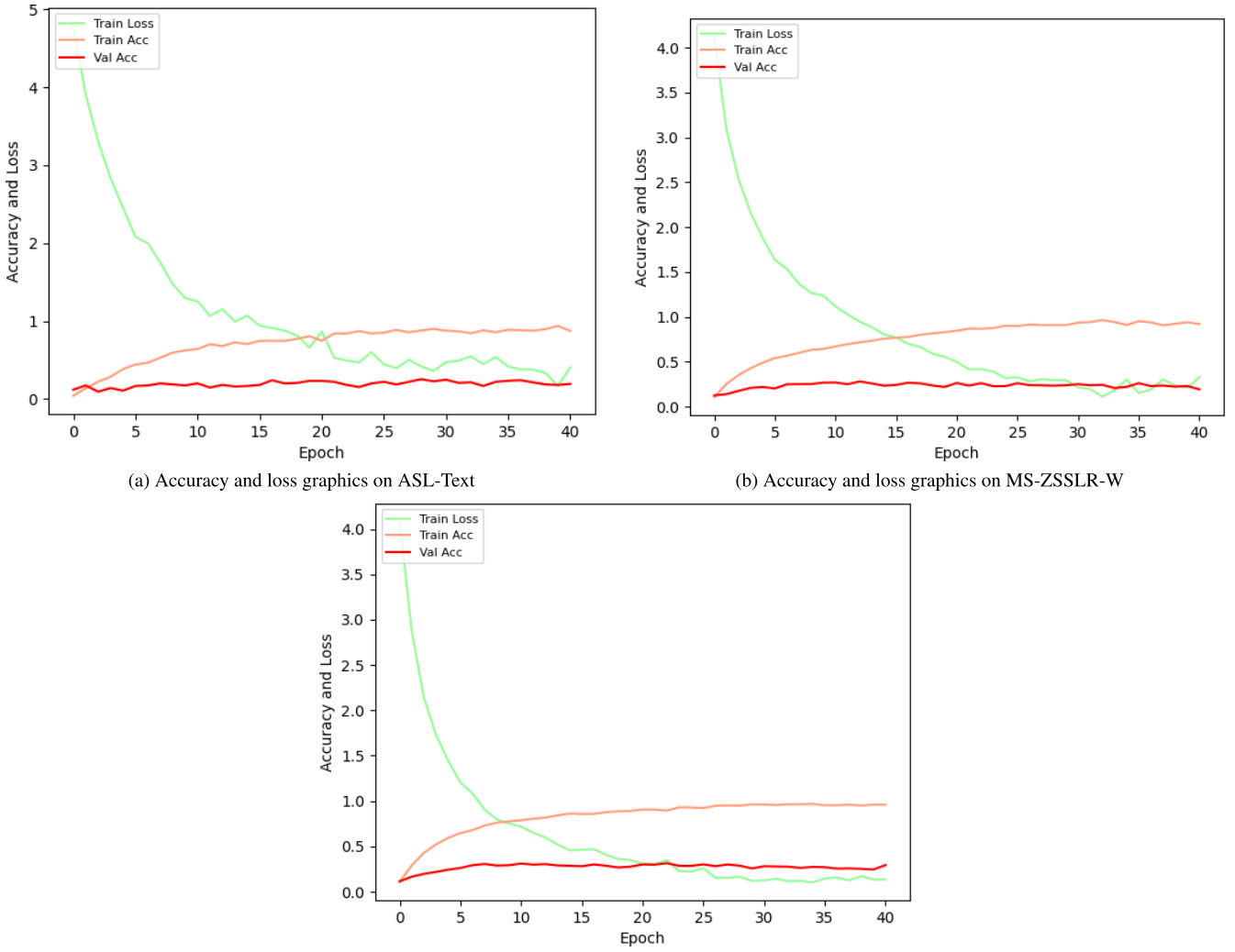
We first need to calculate the compatibility score s, between visual embedding matrix ν and learned language embedding of label l:

$$s(\nu, l) = \alpha(W\nu + \rho)^T y_l \tag{8}$$

where α is the non-linear function of activation, W is learnable matrix of compatibility, ρ is bias term and y_l is label embedding at layer l. Then, we calculate the probability of each label by:

$$P(l|\nu) = \frac{\exp(s(\nu, l))}{\sum_{l'\in L}\exp(s(\nu, l'))} \tag{9}$$

where L includes both seen and unseen labels. The highest probability is then chosen as the class of that sample.



(a) Accuracy and loss graphics on ASL-Text

(b) Accuracy and loss graphics on MS-ZSSLR-W

(c) Accuracy and loss graphics on MS-ZSSLR-C

FIGURE 4. Accuracy and loss graphics on datasets.

IV. EXPERIMENTS

A. IMPLEMENTATION DETAILS

The architecture we propose as a model, which was applied in the latest experiment, can be seen in Figure 2, and the specifics of experiments will be addressed in subsequent paragraphs.

The initial step of the experiments on the MS-ZSSLR-W/C [1] and ASL-Text [3] datasets involved extracting input data. As far as we know, these datasets are the only publicly accessible datasets in this field. We used MediaPipe [78] to separate the streams of the left and right hands, both hands combined, and hand and pose landmarks from the body stream. Features from each frame of the right and left hand streams were obtained using a ResNeXt [12] model trained on the HaGRID [15] dataset, with the frame count fixed at 32 and a stride of 4. Similarly, features from the body and hand streams were extracted using MViTv2 [14]. We utilized the pre-trained ResNeXt50 model, which achieved a 98.3 F1 score on the HaGRID dataset [15], and the MViTv2-S pre-trained model, sized at 16 × 4, which achieved an 81.0% accuracy on the Kinetics-400 [79] dataset.

1D-CNN and self-attention were applied to ResNeXt features to extract spatial and temporal relationships, selecting the best features to improve performance. Bi-LSTM has been applied to body and hand stream features to extract long-term relationships and adapt them to the existing model. Bi-LSTM was also applied to the hand landmarks, while ST-GCN [13] was applied to the pose landmarks. For ST-GCN, only shoulder and hand landmarks from MediaPipe's pose landmarks were taken. The neighborhood matrix of landmarks obtained from MediaPipe was integrated into the model. Finally, due to the large amount of input data, data grading was performed, which was applied with self-attention.

For textual feature extraction, CLIP [16] architecture was utilized, focusing on sentence-based features with a

TABLE 1. Landmarks and features used in experiments.

Experiment #	MViTv2 Body Stream Features	MViTv2 Hand Stream Features	Pose Landmarks	Hand Landmarks	Right Hand ResNeXt Features	Left Hand ResNeXt Features
#1		✓				
#2	✓					
#3	✓	✓				
#4	✓	✓	✓	✓		
#5	✓				✓	✓
#6	✓		✓		✓	✓
#7	✓	✓	✓	✓	✓	✓

length of 768. Due to the CLIP implementation's limitation without fine tuning of 77 words per class, we restricted the maximum number of words in sign class descriptions to 77. We employed the CLIP-ViT-L/14-336px model, which achieved superior performance on many datasets [16].

Top-n accuracy metric was used for measuring the accuracy of the model predictions, where a prediction is considered correct, providing that the ground truth is included among the top-n model's guesses sorted by confidence. We use the Top-1 measure for the model's performance based on whether its top prediction exactly matches the correct answer. Top-2 and Top-5 measure metrics were used to detect whether the true label is among the model's top two or five guesses. This is required because we are working on a large number of classes, which is 50 for the test split of datasets used. The AUC (Area Under Curve) metric was used to better measure the differences between positive and negative classes. Since this study involves multi-class classification, the One-vs-Rest (OvR) approach was followed when calculating the AUC score, with the class to which the example belongs considered as the positive class and the others as negative classes. The average AUC score across all classes is provided. Additionally, to better understand the effectiveness of the developed method, the changes in classifier loss and performance over epochs have been graphically presented.

B. DATASETS

In this study, three existing datasets have been used. These are ASL-Text [3], MS-ZSSLR-W(ild)/C(lean) [1]. ASL-Text was created from the ASLLVD [80] dataset. From the ASLLVD dataset, 250 classes containing the most signers and samples were selected, and while the classes were divided into train, validation, and test, textual descriptions necessary for ZSL were also extracted for each class. The MS-ZSSLR-W/C datasets were created from the MS-ASL [81] dataset. As with the creation of the ASL-Text dataset, 200 classes with the most signers and examples from the MS-ASL dataset were selected, and textual descriptions for each class were extracted while being divided into train, validation, and test. MS-ZSSLR-W includes lingual variation; thus, it can model real life more effectively. MS-ZSSLR-C, on the other hand, does not include lingual variation, offering a benchmark in a more controlled and pristine environment.

While the ASL-Text dataset comprises 1188 videos in the training set, the validation set has 151, and the test set includes 259 videos. The MS-ZSSLR-W/C datasets are larger than the ASL-Text dataset. Specifically, the MS-ZSSLR-W dataset contains 5862, 1153, and 1846 videos in training, validation, and test sets, respectively. On the other hand, the MS-ZSSLR-C dataset includes 7303, 1368, and 1779 videos in training, validation, and test sets, respectively. In total, there are 1598 samples in the ASL-Text dataset, 8861 in the MS-ZSSLR-W dataset, and 10450 in the MS-ZSSLR-C dataset.

C. RESULTS

Numerous experiments have been regulated using different data sources to establish the success of the introduced model. These sources can be seen in Table 1. In the first experiment, only MViTv2 hand features were used, while in the second, only MViTv2 body features were utilized. The third experiment combined both MViTv2 hand and body features. In the fourth, alongside MViTv2 features, pose and hand landmark data were used; in the fifth, landmark data were removed, and ResNeXt features for the right and left hands were added. The sixth experiment included pose landmark data in addition to the data used in the fifth, and the seventh used all data sources.

The outcomes from the ASL-Text dataset are detailed in Table 2. The experiments demonstrated that Experiment#7, which utilized all available data and features, accomplished the highest performance on the test dataset. Regarding the validation dataset, it attained performances of 24.31, 34.8, and 59.64 for Top-1, Top-2, and Top-5, respectively. Meanwhile, the test dataset saw performances of 22.76, 32.83, and 50.15 in the same categories with the average AUC score of 0.71.

Table 3 illustrates the results on the MS-ZSSLR-W dataset. In the conducted experiments, it can be seen that Experiment#7, which used all data and features, yielded the best results for the test dataset. For the validation dataset, performances of 28.75, 42.63, and 67.67 in Top-1, Top-2, and Top-5, respectively, were achieved, while for the test dataset, performances of 14.86, 24.6, and 43.07 were obtained with the average AUC score of 0.74.

The results from the experiments conducted for MS-ZSSLR-C can be observed in Table 4. The same situation

TABLE 2. Results obtained from ASL-Text dataset.

Experiment #	Val (30 classes)			Test (50 classes)			Average
	Top-1	Top-2	Top-5	Top-1	Top-2	Top-5	AUC Score
#1	17.62	26.78	54.67	22.46	30.77	47.18	0.68
#2	23.87	31.2	50.84	18.12	26.8	45.33	0.68
#3	23.58	34.96	**60.91**	21.89	30.53	**50.71**	0.74
#4	**26.09**	**35.51**	57.95	21.22	30.38	45.48	0.72
#5	21.98	32.36	56.49	17.89	27.55	43.11	0.73
#6	23.98	34.18	56.11	18.52	29.56	48.15	0.71
#7	24.31	34.8	59.64	**22.76**	**32.83**	50.15	0.71

TABLE 3. Outcomes derived from MS-ZSSLR-W dataset.

Experiment #	Val (30 classes)			Test (50 classes)			Average
	Top-1	Top-2	Top-5	Top-1	Top-2	Top-5	AUC Score
#1	21.01	32.31	58.09	10.71	18.4	35.37	0.71
#2	24.13	34.81	59.02	13.02	21.26	38.21	0.72
#3	25.09	37.44	63.69	14.11	23.74	42.6	0.74
#4	28.15	**42.83**	**67.89**	14.37	23.26	41.3	0.74
#5	26.68	39.03	62.65	14.43	22.56	40.33	0.71
#6	25.39	38.57	65.96	14.37	22.38	39.15	0.72
#7	**28.7**	42.63	67.67	**14.86**	**24.6**	**43.07**	0.74

observed in the MS-ZSSLR-W experiment is also valid for the experiments conducted on the MS-ZSSLR-C dataset. Experiment#7, where all features and data were used, showed the highest performance. For the validation dataset, performances of 31.73, 47.23, and 72.6 in Top-1, Top-2, and Top-5, respectively, were obtained, and for the test dataset, performances of 18.67, 30.29, and 50.92 were achieved with the average AUC score of 0.76.

The accuracy and loss graphs of Experiment#7 on ASL-Text, MS-ZSSLR-W and MS-ZSSLR-C can be seen on Figure 4a, Figure 4b,and Figure 4c, respectively. As can be seen in all the figures, as the epochs progress, the performance of the training and validation sets increases while the loss decreases. This indicates that the models are learning successfully.

The comparison of the results received from the experiments conducted for all datasets with the SOTA baseline approaches and improvements (differences between SOTA and our approach) can be revealed in Table 5. The results from the test datasets of Experiment#7 have been taken. This observation reflects the improved outcomes drawn from the studies in the article introducing the MViTv2 model [14]. Moreover, the extraction of text features using the CLIP architecture, trained via a zero-shot method as opposed to the BERT architecture trained with supervised methods used in the baseline study [1], enhances the adaptability of this architecture to other zero-shot frameworks. This enhancement is identified as one of the contributors to the increased success depicted in this table. The integration of the ST-GCN method applied for the first time to the pose landmarks in this study and the Bi-LSTM method applied to hand landmarks has resulted in a greater increase in performance compared to the method where ResNeXt features are extracted from separated right and left hand videos, previously trained on the HaGRID dataset. The reason for this is attributed to the HaGRID dataset consisting exclusively of high-resolution videos, whereas the utilized

TABLE 4. Results on the MS-ZSSLR-C dataset.

Experiment #	Val (30 classes)			Test (50 classes)			Average
	Top-1	Top-2	Top-5	Top-1	Top-2	Top-5	AUC Score
#1	22.84	35.68	60.77	14.27	23.95	43.74	0.73
#2	23.97	36.57	59.98	18.16	27.69	45.48	0.77
#3	26.62	40.43	66.28	17.22	28.77	49.4	0.76
#4	30.64	44.72	70.07	18.38	**30.33**	50.6	0.75
#5	28.14	42.01	67.52	17.88	27.73	46.62	0.73
#6	29.19	43.74	68.18	18.48	28.24	45.96	0.77
#7	**31.73**	**47.23**	**72.6**	**18.67**	30.29	**50.92**	0.76

TABLE 5. Contributions on datasets.

Study		Top-1	Top-2	Top-5
Baseline	MS-ZSSLR-W [1]	8.7	15.7	30.3
	MS-ZSSLR-C [1]	7.01	11.5	22.1
	ASL-Text [3]	20.9	32.5	**51.4**
Ours	MS-ZSSLR-W	**14.86**	**24.6**	**43.07**
	MS-ZSSLR-C	**18.67**	**30.29**	**50.92**
	ASL-Text	**22.76**	**32.83**	50.15
Improvement	MS-ZSSLR-W	+6.16	+8.9	+12.77
	MS-ZSSLR-C	+11.66	+18.79	+28.82
	ASL-Text	+1.89	+0.33	-1.25

datasets in the present study are low-resolution. Looking at the table, it is clearly visible that our study shows better performance. In the ASL-Text dataset, the performance gap between the baseline and our approach is minimal; we think this primarily because the dataset has fewer training samples per class compared to others. However, for the MS-ZSSLR-W dataset, almost twice the performance has been achieved, and for the MS-ZSSLR-C dataset, more than double the performance has been obtained.

D. ABLATION STUDY

We perform ablation studies to meticulously analyze the introduced model by isolating and assessing the effects of its individual components. These studies are essential to confirm the usefulness and efficiency of each part, especially in a domain where the complexity of models might conceal the elements critical for performance. By methodically removing or altering components of our model, we seek to identify their distinct roles and contributions to the model's overall effectiveness. This method not only validates our design decisions but also deepens our understanding of how components interact within complex systems, resulting in models that are both more robust and interpretable.

The test results of Experiment#1 in the ASL-Text, which utilized only hand features, are close to the test results of Experiment#7, where we employed all data and features, as shown in Figure 2. However, Experiment#1 also holds the lowest validation scores among tests conducted on the ASL-Text. This is attributed to the smaller number of training samples in ASL-Text compared to other datasets utilized in this study, as well as the limitations of using only hand features, pointing to issues of data scarcity and inadequate representation. This conclusion becomes more apparent when considering that Experiment#1 test results

in the MS-ZSSLR-W/C datasets fail to surpass validation outcomes. It can be said that the test results in Experiment#1 of ASL-Text were due to random assignments since they surpassed the validation results. Among the Experiment#1 applied to the MS-ZSSLR-W/C datasets, which is shown in Table 3 and 4 respectively, the lowest test results have been obtained within the dataset. However, since there are many samples included in the training split of these datasets, this low performance can be solely ascribed to the inadequate representation of the data.

Contradictory results have been obtained in experiments conducted across all datasets, but the best validation and test results in MS-ZSSLR-W and MS-ZSSLR-C were obtained from Experiment#7, where all data sources were used. When looking at Experiment#5 and Experiment#6, where only the pose landmark data changed, it is observed that the use of ST-GCN either improves performance or yields almost identical results in the MS-ZSSLR-W dataset case. Upon examining the first four experiments, the best test results are seen in Experiment#4, where we applied ST-GCN to pose landmarks.

V. CONCLUSION

This study aims to perform ZSSLR using a person's hand, pose, and body data. The application of ST-GCN to pose landmark data, the use of MViTv2 and ResNeXt architectures pre-trained on the HaGRID dataset for feature extraction, the implemented Bi-LSTM and self-attention methods, and the use of the CLIP architecture for extracting textual vectors are factors that enhance performance. These methods have been tested on the readily available MS-ZSSLR-W, MS-ZSSLR-C, and ASL-Text datasets, resulting in improved performance. The results from these datasets show that the field of ZSSLR requires further exploration. The primary challenge in ZSSLR revolves around effectively creating visual and textual representations while establishing robust connections between them. This challenge revolves around the inherent complexity of bridging the semantic disparity between textual descriptions and visual modality. Moreover, the absence of direct correspondence between seen and unseen classes complicates this challenge, requiring sophisticated techniques for knowledge transfer and generalization. Different methods can be applied to improve the representations of visual and textual data to achieve greater results. Different architectures designed to classify hand gestures can be tried to enhance visual representations. Additionally, landmark data for the shoulder and hand can be combined to test various GCN architectures. Characteristics found in sign languages can be added with binary features to improve textual representations.

REFERENCES

[1] Y. C. Bilge, R. G. Cinbis, and N. Ikizler-Cinbis, "Towards zero-shot sign language recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 1, pp. 1217–1232, Jan. 2023.

[2] J. Wu, Y. Zhang, and X. Zhao, "A prototype-based generalized zero-shot learning framework for hand gesture recognition," in *Proc. 25th Int. Conf. Pattern Recognit. (ICPR)*, Jan. 2021, pp. 3435–3442.

[3] Y. C. Bilge, N. Ikizler-Cinbis, and R. G. Cinbis, "Zero-shot sign language recognition: Can textual data uncover sign languages?" in *Proc. Brit. Mach. Vis. Conf. (BMVC)*, 2019, pp. 1–14.

[4] U. Nandi, A. Ghorai, M. M. Singh, C. Changdar, S. Bhakta, and R. K. Pal, "Indian sign language alphabet recognition system using CNN with diffGrad optimizer and stochastic pooling," *Multimedia Tools Appl.*, vol. 82, no. 7, pp. 9627–9648, Mar. 2023.

[5] J. Gangrade and J. Bharti, "Vision-based hand gesture recognition for Indian sign language using convolution neural network," *IETE J. Res.*, vol. 69, no. 2, pp. 723–732, Feb. 2023.

[6] B. Subramanian, B. Olimov, S. M. Naik, S. Kim, K.-H. Park, and J. Kim, "An integrated mediapipe-optimized GRU model for Indian sign language recognition," *Sci. Rep.*, vol. 12, no. 1, p. 11964, Jul. 2022.

[7] S. Katoch, V. Singh, and U. S. Tiwary, "Indian sign language recognition system using SURF with SVM and CNN," *Array*, vol. 14, Jul. 2022, Art. no. 100141.

[8] Y. Min, A. Hao, X. Chai, and X. Chen, "Visual alignment constraint for continuous sign language recognition," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 11522–11531.

[9] M. P. Lewis and F. Gary, *Ethnologue: Languages of the World*, C. D. Fennig, Ed., Dallas, TX, USA: SIL Int., 2015, pp. 62–233.

[10] R. Hills, D. Renner, P. S. Lott, and C. Valli, *The Gallaudet Dictionary of American Sign Language*. Washington, DC, USA: Gallaudet Univ. Press, 2021.

[11] D. Brien, *Dictionary of British Sign Language: English*. London, U.K.: Faber & Faber, 1992.

[12] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5987–5995.

[13] S. Yan, Y. Xiong, and D. Lin, "Spatial temporal graph convolutional networks for skeleton-based action recognition," in *Proc. AAAI Conf. Artif. Intell.*, vol. 32, 2018, pp. 1–9.

[14] Y. Li, C.-Y. Wu, H. Fan, K. Mangalam, B. Xiong, J. Malik, and C. Feichtenhofer, "MViTv2: Improved multiscale vision transformers for classification and detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2022, pp. 4804–4814.

[15] A. Kapitanov, K. Kvanchiani, A. Nagaev, R. Kraynov, and A. Makhliarchuk, "HaGRID–HAnd gesture recognition image dataset," 2022, *arXiv:2206.08219*.

[16] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, and J. Clark, "Learning transferable visual models from natural language supervision," in *Proc. Int. Conf. Mach. Learn.*, vol. 139, 2021, pp. 8748–8763.

[17] J. Duan, S. Zhou, J. Wan, X. Guo, and S. Z. Li, "Multi-modality fusion based on consensus-voting and 3D convolution for isolated gesture recognition," 2016, *arXiv:1611.06689*.

[18] J. Pu, W. Zhou, and H. Li, "Dilated convolutional network with iterative optimization for continuous sign language recognition," in *Proc. 27th Int. Joint Conf. Artif. Intell.*, Jul. 2018, p. 7.

[19] T. G. Zimmerman, J. Lanier, C. Blanchard, S. Bryson, and Y. Harvill, "A hand gesture interface device," *ACM SIGCHI Bull.*, vol. 18, no. 4, pp. 189–192, Apr. 1987.

[20] S. Tamura and S. Kawasaki, "Recognition of sign language motion images," *Pattern Recognit.*, vol. 21, no. 4, pp. 343–353, Jan. 1988.

[21] M. B. Waldron and S. Kim, "Isolated ASL sign recognition system for deaf persons," *IEEE Trans. Rehabil. Eng.*, vol. 3, no. 3, pp. 261–271, Sep. 1993.

[22] M. W. Kadous, "Machine recognition of Auslan signs using PowerGloves: Towards large-lexicon recognition of sign language," in *Proc. Workshop Integr. Gesture Lang. Speech*, 1996, pp. 165–174.

[23] M. Zahedi, D. Keysers, T. Deselaers, and H. Ney, "Combination of tangent distance and an image distortion model for appearance-based sign language recognition," in *Proc. 27th DAGM Symp.*, Vienna, Austria. Berlin, Germany: Springer, Sep. 2005, pp. 401–408.

[24] H. Cooper and R. Bowden, "Sign language recognition using boosted volumetric features," in *Proc. IAPR Conf. Mach. Vis. Appl.*, 2007, pp. 359–362.

[25] O. Koller, J. Forster, and H. Ney, "Continuous sign language recognition: Towards large vocabulary statistical recognition systems handling multiple signers," *Comput. Vis. Image Understand.*, vol. 141, pp. 108–125, Dec. 2015.

[26] D. Li, C. R. Opazo, X. Yu, and H. Li, "Word-level deep sign language recognition from video: A new large-scale dataset and methods comparison," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2020, pp. 1448–1458.

[27] O. Koller, S. Zargaran, H. Ney, and R. Bowden, "Deep sign: Hybrid CNN-HMM for continuous sign language recognition," in *Proc. Brit. Mach. Vis. Conf.*, 2016, pp. 1–136.

[28] O. Koller, S. Zargaran, and H. Ney, "Re-sign: Re-aligned end-to-end sequence modelling with deep recurrent CNN-HMMs," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3416–3424.

[29] O. Koller, N. C. Camgoz, H. Ney, and R. Bowden, "Weakly supervised learning with multi-stream CNN-LSTM-HMMs to discover sequential parallelism in sign language videos," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 9, pp. 2306–2320, Sep. 2020.

[30] T. Starner, J. Weaver, and A. Pentland, "Real-time American sign language recognition using desk and wearable computer based video," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 12, pp. 1371–1375, Dec. 1998.

[31] S. Albanie, G. Varol, L. Momeni, T. Afouras, J. S. Chung, N. Fox, and A. Zisserman, "BSL-1K: Scaling up co-articulated sign language recognition using mouthing cues," in *Proc. 16th Eur. Conf.*, Glasgow, U.K. Cham, Switzerland: Springer, Aug. 2020, pp. 35–53.

[32] L. Momeni, G. Varol, S. Albanie, T. Afouras, and A. Zisserman, "Watch, read and lookup: Learning to spot signs from multiple supervisors," in *Proc. Asian Conf. Comput. Vis.*, 2020, pp. 1–18.

[33] N. Cihan Camgöz, O. Koller, S. Hadfield, and R. Bowden, "Sign language transformers: Joint end-to-end sign language recognition and translation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 10020–10030.

[34] S. Jiang, B. Sun, L. Wang, Y. Bai, K. Li, and Y. Fu, "Skeleton aware multi-modal sign language recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2021, pp. 3408–3418.

[35] C. K. M. Lee, K. K. H. Ng, C.-H. Chen, H. C. W. Lau, S. Y. Chung, and T. Tsoi, "American sign language recognition and training method with recurrent neural network," *Exp. Syst. Appl.*, vol. 167, Apr. 2021, Art. no. 114403.

[36] H. Hu, W. Zhou, and H. Li, "Hand-model-aware sign language recognition," in *Proc. AAAI Conf. Artif. Intell.*, vol. 35, 2021, pp. 1558–1566.

[37] L. Hu, L. Gao, Z. Liu, and W. Feng, "Continuous sign language recognition with correlation network," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 2529–2539.

[38] H. Zhou, W. Zhou, Y. Zhou, and H. Li, "Spatial–temporal multi-cue network for sign language recognition and translation," *IEEE Trans. Multimedia*, vol. 24, pp. 768–779, 2022.

[39] A. Tunga, S. V. Nuthalapati, and J. Wachs, "Pose-based sign language recognition using GCN and BERT," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. Workshops (WACVW)*, Jan. 2021, pp. 31–40.

[40] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," 2018, *arXiv:1810.04805*.

[41] M. Vázquez-Enríquez, J. L. Alba-Castro, L. Docío-Fernández, and E. Rodríguez-Banga, "Isolated sign language recognition with multi-scale spatial–temporal graph convolutional networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2021, pp. 3457–3466.

[42] F. Wei and Y. Chen, "Improving continuous sign language recognition with cross-lingual signs," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2023, pp. 23612–23621.

[43] M. Bohácek and M. Hrúz, "Sign pose-based transformer for word-level sign language recognition," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. Workshops (WACVW)*, Jan. 2022, pp. 182–191.

[44] R. Sreemathy, M. Turuk, I. Kulkarni, and S. Khurana, "Sign language recognition using artificial intelligence," *Edu. Inf. Technol.*, vol. 28, no. 5, pp. 5259–5278, May 2023.

[45] D. R. Kothadiya, C. M. Bhatt, T. Saba, A. Rehman, and S. A. Bahaj, "SIGNFORMER: DeepVision transformer for sign language recognition," *IEEE Access*, vol. 11, pp. 4730–4739, 2023.

[46] J. Shin, A. S. Musa Miah, M. A. M. Hasan, K. Hirooka, K. Suzuki, H.-S. Lee, and S.-W. Jang, "Korean sign language recognition using transformer-based deep neural network," *Appl. Sci.*, vol. 13, no. 5, p. 3029, Feb. 2023.

[47] J. Bora, S. Dehingia, A. Boruah, A. A. Chetia, and D. Gogoi, "Real-time assamese sign language recognition using MediaPipe and deep learning," *Proc. Comput. Sci.*, vol. 218, pp. 1384–1393, Jan. 2023.

[48] H. Larochelle, D. Erhan, and Y. Bengio, "Zero-data learning of new tasks," in *Proc. AAAI*, vol. 1, 2008, p. 3.

[49] M. Palatucci, D. Pomerleau, G. E. Hinton, and T. M. Mitchell, "Zero-shot learning with semantic output codes," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 22, 2009, pp. 1–9.

[50] C. H. Lampert, H. Nickisch, and S. Harmeling, "Learning to detect unseen object classes by between-class attribute transfer," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 951–958.

[51] T. Mensink, E. Gavves, and C. G. M. Snoek, "COSTA: Co-occurrence statistics for zero-shot classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 2441–2448.

[52] Y. Fu, T. M. Hospedales, T. Xiang, Z. Fu, and S. Gong, "Transductive multi-view embedding for zero-shot recognition and annotation," in *Proc. 13th Eur. Conf.*, Zurich, Switzerland. Cham, Switzerland: Springer, Sep. 2014, pp. 584–599.

[53] B. Romera-Paredes and P. Torr, "An embarrassingly simple approach to zero-shot learning," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 2152–2161.

[54] X. Xu, T. Hospedales, and S. Gong, "Semantic embedding space for zero-shot action recognition," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2015, pp. 63–67.

[55] J. L. Ba, K. Swersky, S. Fidler, and R. Salakhutdinov, "Predicting deep zero-shot convolutional neural networks using textual descriptions," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 4247–4255.

[56] G. Ou, G. Yu, C. Domeniconi, X. Lu, and X. Zhang, "Multi-label zero-shot learning with graph convolutional networks," *Neural Netw.*, vol. 132, pp. 333–341, Dec. 2020.

[57] S. Chen, Z. Hong, Y. Liu, G. S. Xie, B. Sun, H. Li, Q. Peng, K. Lu, and X. You, "Transzero: Attribute-guided transformer for zero-shot learning," in *Proc. AAAI Conf. Artif. Intell.*, vol. 36, Jun. 2022, pp. 330–338.

[58] A. Roy, D. Ghosal, E. Cambria, N. Majumder, R. Mihalcea, and S. Poria, "Improving zero-shot learning baselines with commonsense knowledge," *Cognit. Comput.*, vol. 14, no. 6, pp. 2212–2222, Nov. 2022.

[59] J. Wang, X. Wang, and H. Zhang, "Domain-aware multi-modality fusion network for generalized zero-shot learning," *Neurocomputing*, vol. 488, pp. 23–35, Jun. 2022.

[60] A. Gupta, S. Narayan, S. Khan, F. S. Khan, L. Shao, and J. van de Weijer, "Generative multi-label zero-shot learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 12, pp. 14611–14624, Dec. 2023.

[61] D. Cheng, G. Wang, B. Wang, Q. Zhang, J. Han, and D. Zhang, "Hybrid routing transformer for zero-shot learning," *Pattern Recognit.*, vol. 137, May 2023, Art. no. 109270.

[62] S. N. Gowda, "Synthetic sample selection for generalized zero-shot learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2023, pp. 58–67.

[63] J. Guo, S. Guo, Q. Zhou, Z. Liu, X. Lu, and F. Huo, "Graph knows unknowns: Reformulate zero-shot learning as sample-level graph recognition," in *Proc. AAAI Conf. Artif. Intell.*, vol. 37, 2023, pp. 7775–7783.

[64] M. Liu, F. Li, C. Zhang, Y. Wei, H. Bai, and Y. Zhao, "Progressive semantic-visual mutual adaption for generalized zero-shot learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 15337–15346.

[65] E. Kodirov, T. Xiang, and S. Gong, "Semantic autoencoder for zero-shot learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4447–4456.

[66] J. Lin, C. Gan, and S. Han, "TSM: Temporal shift module for efficient video understanding," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, Oct. 2019, pp. 7083–7093.

[67] R. A. Nihal, S. Rahman, N. M. Broti, and S. Ahmed Deowan, "Bangla sign alphabet recognition with zero-shot and transfer learning," *Pattern Recognit. Lett.*, vol. 150, pp. 84–93, Oct. 2021.

[68] A. Yin, Z. Zhao, W. Jin, M. Zhang, X. Zeng, and X. He, "MLSLT: Towards multilingual sign language translation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 5099–5109.

[69] R. Rastgoo, K. Kiani, S. Escalera, and M. Sabokrou, "Multi-modal zero-shot dynamic hand gesture recognition," *Exp. Syst. Appl.*, vol. 247, Aug. 2024, Art. no. 123349.

[70] R. Rastgoo, K. Kiani, S. Escalera, and M. Sabokrou, "Multi-modal zero-shot sign language recognition," 2021, *arXiv:2109.00796*.

[71] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 26, 2013, pp. 1–9.

[72] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2014, pp. 1532–1543.

[73] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 770–778.
[74] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16×16 words: Transformers for image recognition at scale," 2020, *arXiv:2010.11929*.
[75] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–11.
[76] Z. Akata, F. Perronnin, Z. Harchaoui, and C. Schmid, "Label-embedding for attribute-based classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 819–826.
[77] J. Weston, S. Bengio, and N. Usunier, "Large scale image annotation: Learning to rank with joint word-image embeddings," *Mach. Learn.*, vol. 81, no. 1, pp. 21–35, Oct. 2010.
[78] C. Lugaresi, J. Tang, H. Nash, C. McClanahan, E. Uboweja, M. Hays, F. Zhang, C.-L. Chang, M. Guang Yong, J. Lee, W.-T. Chang, W. Hua, M. Georg, and M. Grundmann, "MediaPipe: A framework for building perception pipelines," 2019, *arXiv:1906.08172*.
[79] W. Kay, J. Carreira, K. Simonyan, B. Zhang, C. Hillier, S. Vijayanarasimhan, F. Viola, T. Green, T. Back, P. Natsev, M. Suleyman, and A. Zisserman, "The kinetics human action video dataset," 2017, *arXiv:1705.06950*.
[80] C. Neidle, A. Thangali, and S. Sclaroff, "Challenges in development of the American sign language lexicon video dataset (ASLLVD) corpus," in *Proc. 5th Workshop Represent. Process. Sign Lang., Interact. Between Corpus Lexicon*, 2012, pp. 1–8.
[81] H. R. V. Joze and O. Koller, "MS-ASL: A large-scale data set and benchmark for understanding American sign language," 2018, *arXiv:1812.01053*.

GIRAY SERCAN ÖZCAN received the B.Sc. and M.Sc. degrees in computer engineering from Başkent University, Türkiye, in 2010 and 2014, respectively. He is currently a Research Assistant in computer engineering with Başkent University. His main research interests include computer vision, machine learning, and bioinformatics.

YUNUS CAN BILGE received the B.Sc. degree from Izmir University of Economics, Türkiye, the master's degree from the University of New South Wales, Australia, and the Ph.D. degree from Hacettepe University, Türkiye, in 2022. He is currently a Machine Learning Engineer with Havelsan, Türkiye. His research interests include computer vision and machine learning with minimal supervision.

EMRE SÜMER received the B.Sc. degree in computer engineering from Başkent University, Ankara, Türkiye, in 2001, and the M.Sc. and Ph.D. degrees in geodetic and geographic information technologies from Middle East Technical University, Ankara, in 2004 and 2011, respectively. He is currently an Associate Professor with the Department of Computer Engineering, Başkent University. His primary research interests include image analysis, computer vision, and pattern recognition.