electronics

MDPI

Article

Enhancing Signer-Independent Recognition of Isolated Sign Language through Advanced Deep Learning Techniques and Feature Fusion

Ali Akdag [1,*] and Omer Kaan Baykan [2]

[1] Department of Computer Engineering, Tokat Gaziosmanpaşa University, Taşlıçiftlik Campus, 60250 Tokat, Türkiye
[2] Department of Computer Engineering, Konya Technical University, 42250 Konya, Türkiye; okbaykan@ktun.edu.tr
* Correspondence: ali.akdag@gop.edu.tr

Abstract: Sign Language Recognition (SLR) systems are crucial bridges facilitating communication between deaf or hard-of-hearing individuals and the hearing world. Existing SLR technologies, while advancing, often grapple with challenges such as accurately capturing the dynamic and complex nature of sign language, which includes both manual and non-manual elements like facial expressions and body movements. These systems sometimes fall short in environments with different backgrounds or lighting conditions, hindering their practical applicability and robustness. This study introduces an innovative approach to isolated sign language word recognition using a novel deep learning model that combines the strengths of both residual three-dimensional (R3D) and temporally separated (R(2+1)D) convolutional blocks. The R3(2+1)D-SLR network model demonstrates a superior ability to capture the intricate spatial and temporal features crucial for accurate sign recognition. Our system combines data from the signer's body, hands, and face, extracted using the R3(2+1)D-SLR model, and employs a Support Vector Machine (SVM) for classification. It demonstrates remarkable improvements in accuracy and robustness across various backgrounds by utilizing pose data over RGB data. With this pose-based approach, our proposed system achieved 94.52% and 98.53% test accuracy in signer-independent evaluations on the BosphorusSign22k-general and LSA64 datasets.

Keywords: sign language recognition; deep learning; feature fusion

Citation: Akdag, A.; Baykan, O.K. Enhancing Signer-Independent Recognition of Isolated Sign Language through Advanced Deep Learning Techniques and Feature Fusion. *Electronics* **2024**, *13*, 1188. https://doi.org/10.3390/electronics13071188

Academic Editors: Pedro Javier Herrera Caro and Jaime Duque-Domingo

Received: 19 January 2024
Revised: 16 March 2024
Accepted: 19 March 2024
Published: 24 March 2024

1. Introduction

People express their thoughts, feelings, and intentions through verbal communication, namely, speaking. However, many deaf or hard-of-hearing people have trouble speaking and understanding what is being said. According to the World Federation of the Deaf, there are over 70 million deaf people worldwide, with more than 80 percent living in developing countries [1]. Many of these deaf and hard-of-hearing people use visual signs known as sign language to communicate with one another, rather than spoken words or auditory phrases [2]. These visual signs comprise a mix of manual features—including hand shape, posture, position, and movement—and non-manual features, such as head and body posture, facial expressions, and lip movements [3]. The primary method of communication for deaf people is sign language. Ordinary people cannot understand most signs without training, and the number of interpreters in the community is insufficient, creating a communication barrier between the deaf and the community [4]. Consequently, many deaf individuals may find it challenging to form social relationships and encounter numerous barriers in education, healthcare, and employment. This results in feelings of social isolation and loneliness. Technological advancements have spurred the development of Sign Language Recognition (SLR) studies aimed at addressing this issue.

The primary objective of SLR studies is to develop algorithms and methods for accurate sign recognition. When deaf people's gestures are translated into spoken or written

language, ordinary hearing people can understand them. Therefore, SLR technologies promote the integration of the deaf minority by eliminating the linguistic barrier between them and the hearing majority [4]. SLR is crucial not only for facilitating communication between deaf and hearing communities but also for increasing the content available to the deaf, through initiatives like developing accessible educational tools, games for the deaf community, and creating visual dictionaries of sign language [5]. Developments in the field of deep learning have significantly impacted SLR studies.

In recent years, deep learning methods have succeeded significantly in image and video processing areas [6–9]. This progress has had a significant impact on SLR work, improving the accuracy and efficiency of these systems. Existing techniques based on deep learning are generally based on Convolutional Neural Networks (CNNs). Thus, 2D-CNNs have been used to recognize 2D static images such as sign alphabets or sign digits [10–12]. However, 2D-CNNs alone cannot extract temporal features of sign language words and sentences composed of video frames. This problem has been overcome by first using 2D-CNNs to create representations of video frames and then using Recurrent Neural Network (RNN) such as Long Short-Term Memory (LSTM) networks [13], Bidirectional LSTM (Bi-LSTM) networks [5], or transformers [14] to extract temporal information. Other structures that capture spatial and temporal information in videos are 3D-CNN [15], formed by adding a third dimension to the 2D convolution process, and (2+1)D-CNNs [16], obtained by combining 2D and 1D convolution. For the SLR task, researchers have often used 3D-CNNs [17–19] and (2+1)D-CNNs [20–22]. Although these models are frequently used in the SLR task, there is rarely any research [23,24] investigating the possibility of building a new deep learning model with 3D-CNN and (2+1)D-CNN. In this context, this study develops an innovative deep learning model based on the fusion of the block structures of both methods to take advantage of the advantages of 3D-CNN and (2+1)D-CNN. The resulting model is expected to exhibit a more effective feature extraction and classification performance.

Despite the success of deep learning methods in the SLR task, significant challenges still need to be addressed. In particular, signer differences between signers complicate the recognition process, so the developed systems must perform with high accuracy while being signer-independent [25]. A complete SLR system should consider facial expressions [26,27], body language, and hand gestures since a holistic evaluation of these features is essential for full SLR [28,29]. The dynamic nature of sign language requires integrating both spatial and temporal dimensions of the data, which necessitates the creation of efficient models that can capture both spatial and temporal information. Background differences, varying lighting conditions, and various visual clutter also pose additional challenges for algorithms to correctly recognize and discriminate sign words [30,31]. The development of real-time applications places high demands on computational power and significant efforts to keep algorithms running fast and accurate [32]. Overcoming these challenges requires in-depth research and innovative developments in the field of SLR. Only in this way can the communication barriers between sign language users and the wider society be overcome, enabling both parties to better understand and interact.

The motivation of our research is to overcome the current challenges in SLR technology and to facilitate the communication of hearing-impaired individuals in society by enabling the conversion of sign language gestures into text or speech with the SLR system we develop. In this context, our research aimed to increase the accuracy and efficiency of the SLR process, thereby expanding social integration and communication opportunities for hearing-impaired individuals. In this way, it aimed to build bridges of interaction between individuals using sign language and the hearing community so that both groups can better understand and interact with each other.

For this purpose, we are developing a robust and highly accurate system for recognizing isolated sign language words. This system is based on a novel R3(2+1)D-SLR network that combines the advantages of 3D and (2+1)D convolutional blocks. Designed to efficiently extract both spatial and temporal features, this network enhances the precision of SLR. Our approach aims to create a comprehensive and accurate recognition system by

fusing the features of the sign components (body, hands, and face) extracted from R3(2+1)D-SLR to classify them with SVM. Moreover, by basing our proposed system on pose data, we enhance its robustness to changing background conditions and performance challenges in real-world scenarios, thereby improving SLR accuracy irrespective of background noise.

In conclusion, this research aims to contribute to the elimination of communication barriers between hearing-impaired individuals and the general public and to make significant advances in SLR technology. We briefly summarize the contributions of this study to the literature as follows:

- We introduce a novel R3(2+1)D-SLR network that innovatively merges R(2+1)D and R3D blocks for enhanced sign language interpretation, offering a deeper understanding of both spatial and temporal dimensions of sign language.
- Utilizing the advanced capabilities of the MediaPipe library, our study leverages high-precision pose data to extract detailed spatial and temporal features. This methodology allows for a more accurate and nuanced understanding of sign language gestures, improving the model's performance.
- We achieve a comprehensive SLR system by integrating and classifying complex features from the body, hands, and face, utilizing SVM for superior accuracy, highlighting our approach's effectiveness in capturing the nuances of sign communication.
- Incorporating real-world backgrounds in our testing datasets underlines our system's adaptability and reliability in diverse environments, addressing a critical challenge in SLR.
- Our strategic use of pose images to enhance system robustness against variable backgrounds showcases our commitment to developing practical, real-world SLR solutions.

The rest of the paper is organized as follows: Section 2 presents the literature review; Section 3 presents the proposed R3(2+1)D-SLR model and feature fusion-based system; Section 4 presents the experiments and results; Section 5 concludes the paper.

2. Related Literature

This section reviews recent advancements in vision-based isolated SLR, focusing on research involving the BosphorusSign22k-general [33,34] and LSA64 [35] datasets, and discusses methodological approaches in the field.

In studies on the BosphorusSign22k-general subset, consisting of 174 isolated Turkish Sign Language words, Kindiroglu et al. [36] developed a feature set named Temporal Accumulative Features (TAF) for the isolated SLR task. This feature set is based on the temporal accumulation of heatmaps obtained from joint movements. This feature set aims to recognize and classify sign language gestures by visually encoding how the movements and hand shapes in sign language videos change over time. As a result of the classification using CNN, a recognition accuracy of 81.58% was achieved. In their study, Gündüz and Polat [19] adopted a multistream data strategy, leveraging information from the face, hands, full body, and optical flow for training their Inception3D model. Additionally, for the LSTM network, they focused on utilizing data related to body and hand pose data. By integrating the feature streams generated from these models, they inputted the combined data into a dual-layer neural network. This innovative approach increased the accuracy from 79.6% to an impressive 89.3% on the BosphorusSign22k-general dataset using full-body data, demonstrating the significant impact of various feature combinations on SLR research.

Regarding studies on the LSA64 dataset containing 64 Argentine Sign Language words, Ronchetti et al. [37] proposed a probabilistic model that combines sub-classifiers based on different types of features such as position, movement, and hand shape, achieving an accuracy of 97% on the LSA64 dataset, in addition to an average accuracy of 91.7% in their signer-independent evaluation. Rodriguez et al. [38] proposed a model using cumulative shape difference with SVM and achieved 85% accuracy on the LSA64 dataset. Konstantinidis et al. [39] proposed a model based on processing hand and body skeletal features extracted from RGB videos using LSTM layers and late fusion of the processed features. Their proposed model achieved 98.09% accuracy. In later work, the same authors [40]

improved the performance of their method to 99.84% by adding additional streams that process RGB video and optical flow data. Masood et al. [41] proposed a model that extracts spatial features with CNN and then classifies them by extracting temporal features by feeding the pool layer output into RNN before converting it into a prediction. Their model achieved 95.2% accuracy in 46 subcategories of the LSA64 dataset. Zhang et al. [42] proposed a neural network with an alternative fusion of 3D-CNN and Convolutional LSTM, called a Multiple Extraction and Multiple Prediction (MEMP) network, to extract and predict motion videos' temporal and spatial feature information multiple times. On LSA64, the network achieved an identification rate of 99.063%. Imran and Raman [43] used motion history images, dynamic images, and RGB motion image templates, which can represent a video in a single image, to fine-tune three pre-trained CNNs. By fusing the outputs of these three networks with a fusion technique based on their proposed kernel-based extreme learning machine, they achieved a 97.81% accuracy. The model proposed by Elsayed and Fathy [44] using 3D-CNN followed by Convolutional LSTM achieved a 97.4% test accuracy for 40 categories. Marais et al. [45] trained the Pruned VGG network with raw images and achieved a 95.50% test accuracy. In another study by the same authors [46], using the InceptionV3-GRU architecture, they achieved a 97.03% accuracy in signer-dependent testing and 74.22% in signer-independent testing. Alyami et al. [47] classified the key points extracted from the signer's hands and face using a transformer-based model. On the LSA64 dataset, they achieved a 98.25% and 91.09% accuracy in signer-dependent and independent modes, respectively. Furthermore, the combination of hand and face data improved the recognition accuracy by 4% compared to hand data alone, emphasizing the importance of non-manual features in recognition systems.

Rastgoo et al. [48] utilized a CNN-based model to estimate 3D hand landmark points from images detected by a Single Shot Detector (SSD). They applied the singular value decomposition method, a feature extractor, to the coordinates of these 3D hand key points and the angles between the finger segments, fed the obtained features as input to the LSTM, and predicted 100 Persian signs with a 99.5% accuracy. Samaan et al. [49] achieved accuracies of 99.6%, 99.3%, and 100% in LSTM, Bi-LSTM, and GRU models, respectively, on a ten-class dataset, utilizing pose landmark points derived from MediaPipe. Although the test accuracy of the proposed method is high, the number of classes is quite low compared to the number of words used in general sign language dictionaries. Castro et al. [50] introduced a multi-stream approach involving processing summarized RGB frames, segmented regions of the hands and face, joint distances, and artificially generated depth data through a 3D-CNN. In this method, it was shown that the addition of artificial depth maps increased the generalization capacity for different signers. Their method achieved a 91% recognition accuracy on a dataset with 20 classes. Hamza et al. [51] obtained a 93.33% recognition accuracy using the convolutional 3D model and data augmentation techniques of transformation and rotation on a dataset of 80 classes, where each class contained very few examples. This study demonstrated the effectiveness of data augmentation methods on limited datasets. Laines et al. [52] presented an innovative approach to isolated SLR using a Tree Structure Skeleton Image (TSSI) representation that converts pose data into an RGB image, improving the accuracy of skeleton-based models for SLR. In the TSSI representation, columns represent landmark points, rows capture the evolution of these points over time, and RGB channels encode the (x, y, z) coordinates of the points. These data were classified with DenseNet-121, achieving recognition accuracies of 81.47%, 93.13%, and 98% for datasets with 100, 226, and 30 classes, respectively. In particular, taking into account critical components of sign language, such as facial expressions and detailed hand gestures, significantly improved the accuracy of the model. In their study, Podder et al. [25] obtained an 87.69% recognition accuracy on a dataset of 50 classes with their proposed MobileNetV2-LSTM-SelfMLP model using MediaPipe Holistic-based face and hand-segmented data. The method proposed by Jebali et al. [53] is an innovative training approach to SLR that incorporates an integrated approach of manual and non-manual features. Basically, using deep learning models such as CNN and LSTM, a system is developed that can simultaneously

process information from both hand gestures and non-manual components such as facial expressions, and a significant improvement in system performance is observed with the use of non-manual features. It achieved test accuracies of 90.12% and 94.87% on datasets with 450 and 26 classes, respectively.

The real world for SLR recognition may not consist of a flat or fixed background image, but all of the studies mentioned above were evaluated on images with a flat or fixed background. Consequently, this study tested the proposed systems with varied backgrounds added to the test datasets to better simulate real-world conditions. In addition, the proposed R3(2+1)D-SLR network was trained separately for body, hands, and face images obtained by extracting spatial and temporal features from each video with the help of pose data, and the features were fused to classify with SVM for more accurate SLR. Moreover, to make the proposed system more robust to different background conditions, it is proposed to use images derived from pose data instead of standard raw images.

3. Methodology

In this section, the datasets used in the design of the proposed system, the data preprocessing steps, the proposed deep learning model, the fusion technique used, and the classification method are detailed. The diagram of the proposed final system is shown in Figure 1.

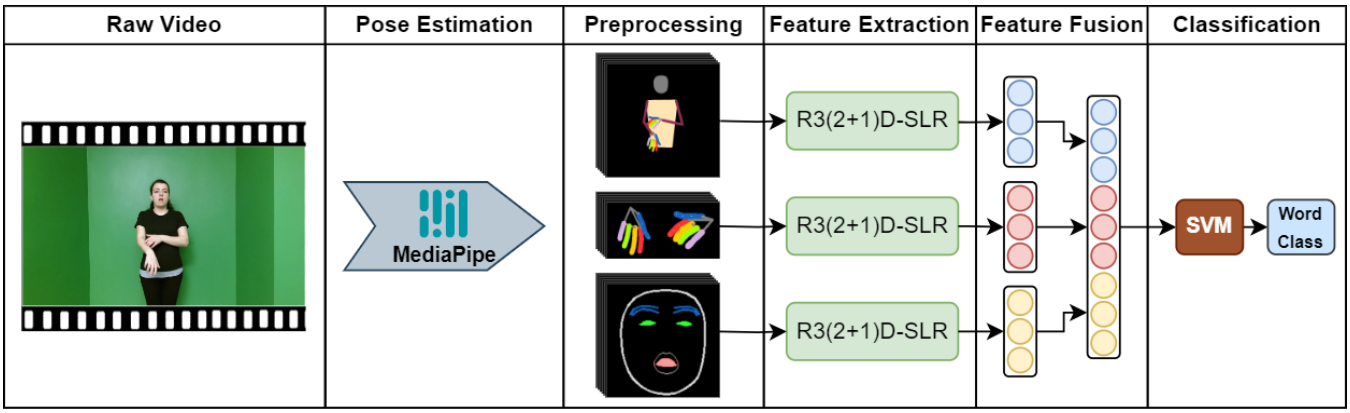

Figure 1. Diagram of the proposed system.

3.1. Datasets

The BosphorusSign22k dataset [33,34] comprises isolated Turkish sign words recorded by six distinct signers. The dataset contains 744 sign words: 428 for health, 163 for finance, and 174 for general use. In this study, the general sub-dataset comprising 174 sign word classes containing a total of 5788 signs was used. Data providers set aside the signs created by "User 4" as test data and all other signs as training data.

The LSA64 dataset [35] consists of 64 Argentine sign language words. The dataset consists of 3200 videos in which ten signers repeat each sign five times. Signers were recorded using colored gloves, with each hand receiving a different color. The dataset is not explicitly separated into training and test sets. Therefore, based on studies in the literature, the dataset was organized for three different experimental studies.

- Experiment 1 (E1): As in the study by Marais et al. [46], the 5th and 10th signers were used as test data and the rest as training data.
- Experiment 2 (E2): Nine out of ten signers were allocated for training, with the remaining one serving as test data. This procedure was iterated ten times, each time selecting a different signer for testing.
- Experiment 3 (E3): The dataset underwent a random split, allocating 80% for training and 20% for testing purposes.

Adding Background Images to Test Datasets

Given that real-world data encompass a variety of backgrounds, analyzing sign language videos against a uniform background may not yield a realistic assessment. In this context, in order to better mimic real-world conditions, 40 different background images selected from indoor and outdoor locations were specially added to the test videos in the datasets. This approach was employed during the testing phase to assess the model's SLR proficiency across diverse backgrounds. Sample frames from the videos with a modified background are presented in Figure 2.

Figure 2. Adding a background to a sample video (from the BosphorusSign22k-general dataset).

This procedure was applied only for the test set; no background changes were made to the videos in the training set. The reason for this approach is to allow the model to learn on a fixed background during the training process and then to objectively test how effectively the model can recognize signs on different backgrounds during the testing process. This methodology allows for a comprehensive evaluation of the model's adaptability and generalization to real-world scenarios.

3.2. Data Preprocessing

3.2.1. MediaPipe Holistic

Our study utilized pose data to extract the spatial and temporal features inherent in sign language videos. To obtain the pose data, we utilized the Holistic solution of the MediaPipe library [54], an open source library based on deep learning and computer vision techniques. MediaPipe Holistic [55] detects up to 543 landmark points: 33 on the body, 468 on the face, and 21 on a single hand. Each landmark point consists of x, y, and z coordinates. The x coordinate is the horizontal position of the landmark; the y coordinate is the vertical position; and the z coordinate is the depth. This detailed pose information allows for a more accurate extraction of spatial and temporal features from sign language word videos.

3.2.2. Spatial Sampling

To obtain spatial information, we first applied the MediaPipe Holistic method to all image frames in each video in the datasets. Utilizing the x and y coordinates of the landmark points, we segmented the images into body, hands, and face regions. After this step, we resized the images to be used in the training process of our deep learning model. We set the body and face images to a 112×112 resolution and the hands images to a 56×112 resolution. As a result of this process, focused data were generated so that our model could analyze each component in detail. The resulting data are shown in Figure 3.

We also aimed to enable our system to recognize sign language signs with a high accuracy and robustness, even under conditions of high background diversity and complexity. For this purpose, we propose to update the traditional RGB video data used in our system with pose data-based images obtained with MediaPipe Holistic. The pose images corresponding to the RGB video data that we created for this purpose are given in Figure 4.

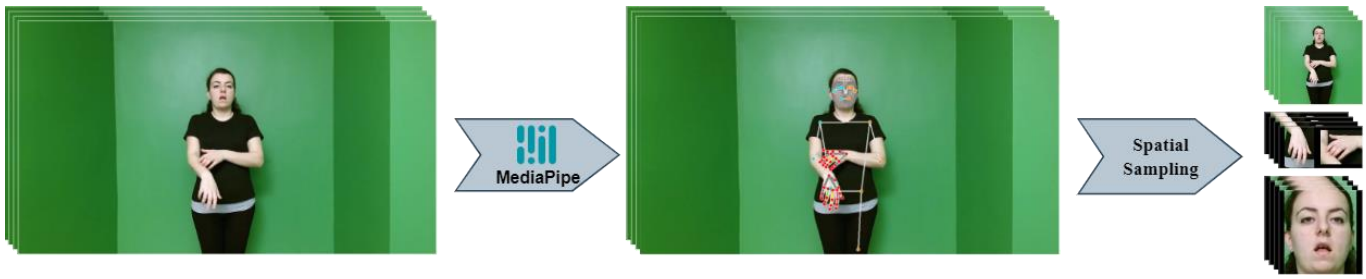

Figure 3. Spatial sampling (from the BosphorusSign22k-general dataset).

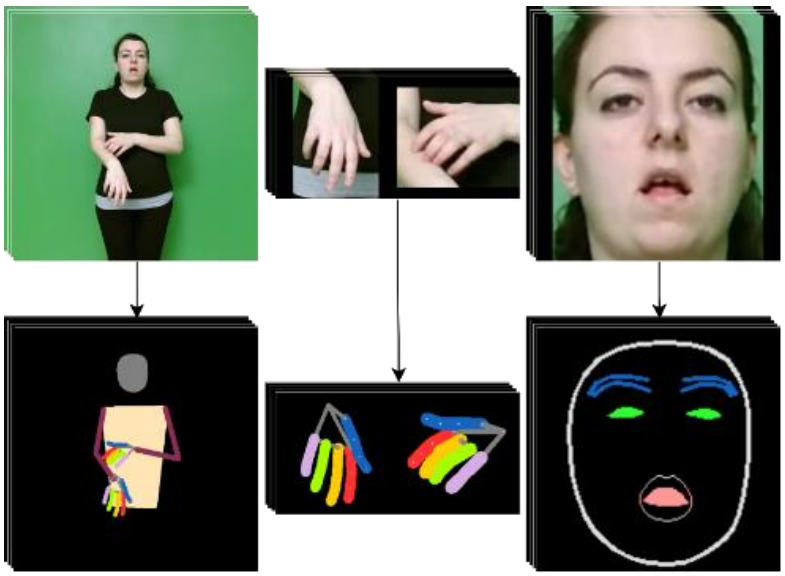

Figure 4. Generation of pose images.

During preprocessing, legs were omitted from body pose images, as they do not influence sign interpretation. In addition, the face was removed from the body pose image as it would be evaluated separately; instead, the head was painted with a solid color to represent the posture position. In the hands pose images, a different color was assigned to each finger in order to analyze the subtle nuances of hand gestures and sign language in more detail. Additionally, we compared the efficacy of colored versus non-colored hands pose images in capturing the nuances of sign language. This approach allows us to study in detail the effect of color coding on the sign recognition capacity of our model. In the face pose images, the eyebrows, eyes, and mouth are painted in different colors to more clearly detect facial expressions and emotional expressions; especially the eyes and the inside of the mouth are completely painted to emphasize opening and closing movements. This detailed approach is expected to increase the robustness of our proposed SLR system to background noise and improve accuracy rates through a comprehensive analysis of sign language components.

3.2.3. Temporal Sampling

Temporal sampling was employed to adapt sign language videos for effective processing by deep learning models. This is because deep learning models usually expect a fixed size of data as input when training. Therefore, converting different lengths of videos into a fixed size is a way to standardize the model's training process and improve its efficiency. In this process, we calculated the Manhattan distance between consecutive frames using the x,y coordinates of the landmark points of each frame obtained using MediaPipe Holistic.

And we repeated this process for all frames in the video. The formulation is given in Equation (1):

$$D^k = \sum_{i=1}^{N} \left(\left| x_i^k - x_i^{k+1} \right| + \left| y_i^k - y_i^{k+1} \right| \right), \quad (1)$$

where D^k is the total Manhattan distance between k and $k+1$ consecutive frames, and N represents the number of landmarks in each frame. This method is calculated for each feature (body, hands, face) using its own landmark points. As a result of this calculation process, we identified the 32 frames that caused the highest variation among the Manhattan distances obtained over the whole video. For videos with fewer than 32 frames, we duplicated the first and last frames as needed to reach a total of 32. This approach captured frames depicting the highest motion intensity, essential for sign language interpretation. Thus, the most informative frames that can represent the sign were identified to be used to train our deep learning model.

3.3. Proposed Model

3.3.1. Convolutional Neural Network (CNN)

Recent years have seen substantial advances in computer vision research, driven by deep learning [56]. At the center of this progress are Convolutional Neural Networks (CNNs), which are particularly prominent in tasks such as image classification [57–60] and object detection [61–63]. CNNs have become a standard in many image processing tasks owing to their ability to extract features from complex image data and make accurate classifications using these features. In particular, Two-Dimensional Convolutional Neural Networks (2D-CNNs) have been widely recognized for their structural simplicity and efficient learning algorithms [64]. The basic convolution function used in a 2D-CNN can be expressed as in Equation (2):

$$f_{2D}(I) = I * K_{2D}, \quad (2)$$

Here, I is an image matrix, K_{2D} is a 2D convolution kernel, and "$*$" represents the convolution process. This process is used to detect features such as patterns, edges, and texture in images. However, they are insufficient for extracting temporal features from video data where the relationship between successive frames is important, such as sign language words. This problem is solved by using 3D convolution kernels, which are obtained by adding a third dimension to 2D convolution to learn both temporal and spatial information from video [65–67].

Deep learning networks called 3D-CNNs [15,68] are networks that process three-dimensional (3D) data, including videos and 3D medical images. These networks use 3D convolution kernels to extract both spatial and temporal features from the data, as opposed to the 2D kernels used by traditional CNNs. This approach makes it possible to effectively detect the motion and dynamics of objects in videos and the spatial relationships between structures in 3D images. The 3D convolution process on a video (V) can be represented in Equation (3):

$$f_{3D}(V) = V * K_{3D}, \quad (3)$$

3D convolution operations are designed to cover all channels of the input data, which means that each convolution kernel (K_{3D}) is of size $C_{in} \times t \times k \times k$. Here, C_{in} represents the number of input channels, t the temporal depth, and k the kernel size (Figure 5a). This structure implies that the kernel can handle the multi-channel nature of the input data and will process all channels simultaneously. As a result of this process, the features extracted by the kernel contain the information of all channels, resulting in C_{out} feature maps. This approach underpins the advanced spatio-temporal feature extraction capability of 3D convolution.

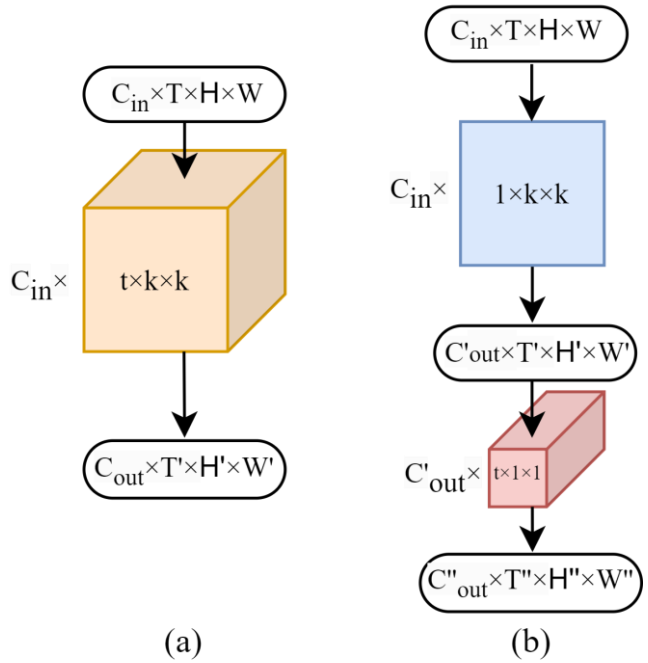

Figure 5. Convolution types: (**a**) 3D convolution; (**b**) (2+1)D convolution.

Another structure used to extract temporal information between consecutive frames is (2+1)D convolution [16]. They decompose the traditional 3D convolution process into spatial and temporal dimensions, first extracting the spatial information in each video frame through 2D convolutions and then processing temporal features through 1D convolutions on the resulting spatial feature maps (Figure 5b). The convolution of a video (V) (2+1)D can be expressed by Equation (4):

$$f_{1D}(f_{2D}(V)) = (V * K_{2D}) * K_{1D}, \quad (4)$$

This decomposed approach allows the model to learn both spatial details and temporal variations more effectively [20]. Moreover, adding activation functions between 2D and 1D convolution operations increases the complexity of the network and its capacity for nonlinear learning. This strategy strengthens the nonlinear properties of the model to better capture the relationships between spatial and temporal dimensions, thus improving the overall performance of the model [16].

3.3.2. R3(2+1)D Convolution Block

This study introduces a fusion of R3D and R(2+1)D convolution blocks, creating the novel R3(2+1)D block structure. A convolution block is the fundamental component of a CNN network, which typically consists of multiple layers, including a convolutional layer, batch normalization, a nonlinear activation function, and a pooling layer. The architecture of a CNN network consists of multiple block structures stacked to form a deeper network. Each block extracts more complex features from the input data, and the final output of the last block is used for classification or other tasks.

The R3D block consists of two 3D convolution layers (Figure 6a), and the R(2+1)D block consists of two (2+1)D convolution layers (Figure 6b). After each layer, 3D Batch Normalization (BN) is applied to make the network more stable and regular, and after each 3D BN, the Rectified Linear Unit (ReLU) activation function is applied. After passing through the convolution layers, the input data are summed with the output value as a residual connection. This value is again passed through a ReLU function and forwarded to the next layers. The output of the i-th residual block can be expressed as follows (Equation (5)):

$$x_i = x_{i-1} + F(x_{i-1}; \theta_i), \quad (5)$$

where x_{i-1} is the input data of layer i; $F(;\theta_i)$ denotes the combination of the two convolutions calculated with weights θ_i and the application of the 3D BN and ReLU functions.

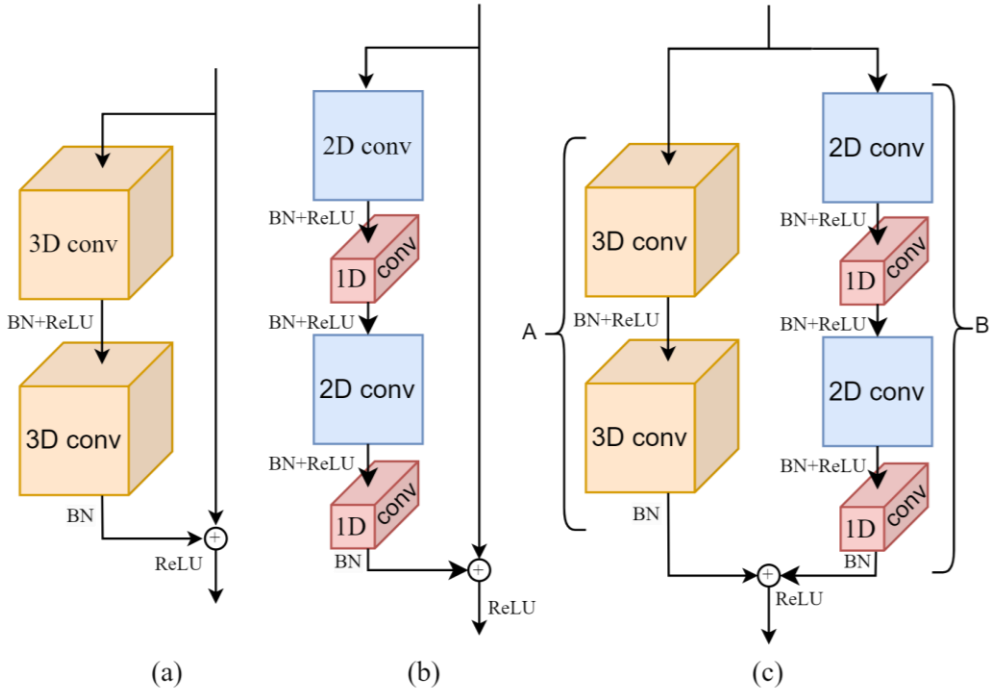

Figure 6. Convolutional block types: (**a**) R3D convolution block; (**b**) R(2+1)D convolution block; (**c**) R3(2+1)D convolution block.

In these structures, 3D BN [69] is applied after each convolution operation. This normalization process accelerates and stabilizes the training of networks by normalizing the inputs in each mini-batch to have a mean of zero and a variance of one. The formula in Equation (6) illustrates the 3D BN process:

$$BN(x) = \gamma \left(\frac{x - \mu_B}{\sqrt{\sigma_B^2 + \epsilon}} \right) + \beta, \tag{6}$$

In this formula, μ_B and σ_B^2 represent the mini-stack mean and variance across depth, height, and width for 3D inputs. γ and β are learnable parameters for scaling and shifting. This approach makes the training of deep 3D models both stable and fast.

After the BN process, the activation function is applied to increase the nonlinearity of the model. In this study, ReLU was preferred as the activation function. The formula of the ReLU activation function is shown in Equation (7):

$$ReLU(x) = \max(0, x), \tag{7}$$

Equation (7) serves as a pivotal component in introducing non-linearity within the network. It operates by retaining any positive input as is, while converting any negative input to zero. This simplicity not only facilitates computational efficiency but also helps in mitigating the vanishing gradient problem, thereby expediting the learning process [70].

In this work, we introduce a novel block structure, R3(2+1)D block, by fusing R3D and R(2+1)D convolution blocks. This fusing is illustrated in Figure 6c, where the input data are processed through both R3D and R(2+1)D blocks, with the outcomes summed at the blocks' outputs. This approach allows us to utilize the strengths of both R3D and

R(2+1)D blocks—A for the 3D convolution block and B for the (2+1)D convolution block. The output for the i-th residual block is formulated as follows (Equation (8)):

$$x_i = \mathrm{ReLU}(A(x_{i-1}) + B(x_{i-1})), \tag{8}$$

To the best of our knowledge, this fusion of block structures in the proposed R3(2+1)D-SLR network represents an innovative step towards improving video data processing. By merging the depth and complexity of R3D blocks with the refined spatial-temporal decomposition offered by R(2+1)D blocks, this architecture aims to achieve a balanced extraction of features from 3D data. In the next section, we introduce the deep learning network we designed for the SLR task using the R3(2+1)D block structure.

3.3.3. Proposed R3(2+1)D-SLR Network

We present R3(2+1)D-SLR network architecture designed for the SLR task, utilizing the R3(2+1)D convolution block, in Table 1 and Figure 7. The architecture comprises a stem layer followed by four layers, each incorporating an R3(2+1)D block. The stem layer, similar in structure to the R3(2+1)D block, comprises a combination of 3D and (2+1)D convolution; however, unlike the R3(2+1)D block, it contains one 3D and one (2+1)D convolution layer each. The convolution blocks in the stem layer have 32 convolution kernels of size 7×7 with a stride of 2, processing the incoming data of dimensions $3 \times 32 \times 112 \times 112$ (where 3 represents the number of RGB channels; 32, the total number of video frames; 112×112, the width and height of the image) and converting them to an output of dimensions $32 \times 32 \times 56 \times 56$, which are then passed to the next layer. The subsequent layers following the stem layer contain 64, 64, 128, and 128 convolution kernels, respectively, with the stride value set to 2 in the first and third layers, performing downsampling.

Table 1. Proposed R3(2+1)D-SLR network architecture.

Layer	Input Size	Output Size	R3(2+1)D-CNN	
Stem layer	$3 \times 32 \times 112 \times 112$	$32 \times 32 \times 56 \times 56$	$3 \times \left[\begin{pmatrix}1\times7\times7 \text{ stride } 1,2,2\\ 3\times1\times1 \text{ stride } 1,1,1\end{pmatrix}\right], 32$	$3 \times [(3 \times 7 \times 7 \text{ stride } 1,2,2)], 32$
Layer 1	$32 \times 32 \times 56 \times 56$	$64 \times 16 \times 28 \times 28$	$32 \times \left[\begin{pmatrix}1\times3\times3 \text{ stride } 1,2,2\\ 3\times1\times1 \text{ stride } 2,1,1\end{pmatrix}\right], 64$ $64 \times \left[\begin{pmatrix}1\times3\times3 \text{ stride } 1,1,1\\ 3\times1\times1 \text{ stride } 1,1,1\end{pmatrix}\right], 64$	$32 \times [(3 \times 3 \times 3 \text{ stride } 2,2,2)], 64$ $64 \times [(3 \times 3 \times 3 \text{ stride } 1,1,1)], 64$
Layer 2	$64 \times 16 \times 28 \times 28$	$64 \times 16 \times 28 \times 28$	$64 \times \left[\begin{pmatrix}1\times3\times3 \text{ stride } 1,1,1\\ 3\times1\times1 \text{ stride } 1,1,1\end{pmatrix}\right], 64$ $64 \times \left[\begin{pmatrix}1\times3\times3 \text{ stride } 1,1,1\\ 3\times1\times1 \text{ stride } 1,1,1\end{pmatrix}\right], 64$	$64 \times [(3 \times 3 \times 3 \text{ stride } 1,1,1)], 64$ $64 \times [(3 \times 3 \times 3 \text{ stride } 1,1,1)], 64$
Layer 3	$64 \times 16 \times 28 \times 28$	$128 \times 8 \times 14 \times 14$	$64 \times \left[\begin{pmatrix}1\times3\times3 \text{ stride } 1,2,2\\ 3\times1\times1 \text{ stride } 2,1,1\end{pmatrix}\right], 128$ $128 \times \left[\begin{pmatrix}1\times3\times3 \text{ stride } 1,1,1\\ 3\times1\times1 \text{ stride } 1,1,1\end{pmatrix}\right], 128$	$64 \times [(3 \times 3 \times 3 \text{ stride } 2,2,2)], 128$ $128 \times [(3 \times 3 \times 3 \text{ stride } 1,1,1)], 128$
Layer 4	$128 \times 8 \times 14 \times 14$	$128 \times 8 \times 14 \times 14$	$128 \times \left[\begin{pmatrix}1\times3\times3 \text{ stride } 1,1,1\\ 3\times1\times1 \text{ stride } 1,1,1\end{pmatrix}\right], 128$ $128 \times \left[\begin{pmatrix}1\times3\times3 \text{ stride } 1,1,1\\ 3\times1\times1 \text{ stride } 1,1,1\end{pmatrix}\right], 128$	$128 \times [(3 \times 3 \times 3 \text{ stride } 1,1,1)], 128$ $128 \times [(3 \times 3 \times 3 \text{ stride } 1,1,1)], 128$
Avgpool	$128 \times 8 \times 14 \times 14$	128	3D adaptive average pooling	
FC	128	Class size	Fully connected layer	
Softmax	Class size	Class size	Softmax classifier	

After data processing through these R3(2+1)D blocks, they are reduced to a 128-dimensional vector by the 3D adaptive average pooling process following the last convolution layer. This process is instrumental in preserving essential features while reducing the data to a more manageable format. The resulting 128-dimensional vector is then fed into a fully connected layer, which serves as a bridge to the classification layer. To finalize the classification process, a softmax function is applied at the output of the fully connected

layer, converting the 128-dimensional vector into probability scores corresponding to the targeted number of classes. The class with the highest probability score is then selected as the model's final prediction, providing a clear and interpretable output for the SLR task. Overall, this architecture allows for a balanced extraction of spatial and temporal information from sign language video data, offering a rich set of features in terms of both depth and complexity.

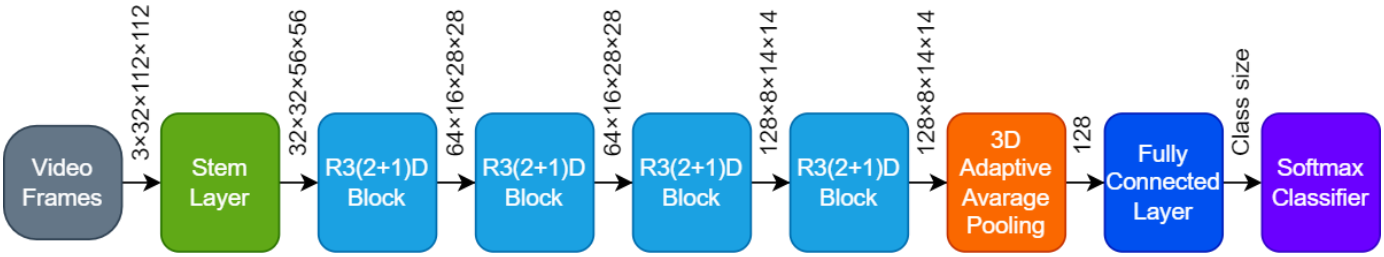

Figure 7. Proposed R3(2+1)D-SLR network structure.

3.4. Feature Fusion

An effective SLR system must consider all sign components' features for completeness. For this reason, we first trained separate models for body, hands, and face data and extracted 128 dimensional feature vectors from each model. Feature fusion merges these diverse feature sets into a unified representation, enabling a comprehensive evaluation. Our proposed SLR system employs simple concatenation for feature fusion. This method creates a single long vector by adding different feature vectors end-to-end. The feature vectors obtained from the body, hands, and face models are combined as in Equation (9):

$$F_{concatenated} = \left[F_{body} \,||\, F_{hands} \,||\, F_{face}\right], \tag{9}$$

Here, the operator $||$ indicates the merging process. This simple yet effective method allows the model to integrate information from different types of features and make a more comprehensive inference of meaning.

3.5. Support Vector Machine

This study utilized a Support Vector Machine (SVM) to classify the fused feature vectors from the models. SVM, a supervised learning algorithm, classifies and regresses by identifying the optimal hyperplane that separates the feature space [71]. SVM creates hyperplanes in a high-dimensional space to classify the training dataset. If the training set cannot be linearly separated, a kernel function is used to pass the data to a new vector space. In this work, we used Radial Basis Function (RBF) kernels [72], where a distance measure is smoothed by a radial (exponential) function. The RBF kernel effectively manages nonlinear relationships between class labels and attributes, unlike linear kernels. The choice of SVM in our study is due to its capacity to work with high-dimensional data [73], its generalization capability [74,75], and its ability to effectively solve nonlinear classification problems [76,77]. Particularly in tasks with complex spatial and temporal features, such as SLR [78,79], these features of SVM significantly increase the success of the model.

4. Experiments and Results

For the implementation of the proposed SLR system, we utilized a PC equipped with Ubuntu 18.04, an Intel Core i5-8400 processor, 16 GB RAM, and a 12 GB GeForce GTX 1080 Ti GPU. In these experiments, the PyTorch library was used. All models were trained with the SGD optimizer for 35 epochs. The batch size was set to 8 and the learning rate was initially set to 1×10^{-2} and then reduced by 1/10 every 15 epochs.

4.1. Training Models

In order to evaluate the performance of our proposed R3(2+1)D-SLR model, we developed the R(2+1)D-10 model with only R(2+1)D blocks and the R3D-10 model with only R3D blocks, both with the same depth. We also used non-pretrained versions of these models in order to benchmark them against the R3D-18 and R(2+1)D-18 models [16] with more layers. For the evaluation of the model performances, we used RGB body images from both BosphorusSign22k-general and LSA64 datasets as training material, as detailed in Section 3.2.2. RGB body data have dimensions of $3 \times 32 \times 112 \times 112$. Post-training test performances of these models are compared in Table 2.

Table 2. Comparison of models.

Models	Test Accuracy (%)		Training Time (s/batch)	Inference Time (ms)	Number of Parameters
	BosphorusSign22k-General	LSA64 (E1)			
R3D-18	71.33	90.93	0.58	372	33 M
R(2+1)D-18	73.12	94.99	1.01	617	31 M
R3D-10	54.79	97.81	0.12	54	1.9 M
R(2+1)D-10	72.18	97.81	0.14	79	1.8 M
R3(2+1)D-SLR	79.66	98.90	0.24	116	3.8 M

Our R3(2+1)D-SLR model demonstrated a significant performance advantage over existing models on the BosphorusSign22k-general and LSA64 datasets. The R3(2+1)D-SLR model presented a balanced efficiency on the SLR task, both in terms of test accuracy (79.66% and 98.90%) and inference time (116 ms). In particular, the model's test accuracy was significantly higher than that of models containing only R(2+1)D or R3D blocks of the same depth, providing important evidence for how a block-level fusion approach can improve the generalization ability of models.

The training time and inference time metrics showed that the R3(2+1)D-SLR model provides an appropriate balance between complexity and performance. This balance underlines that the model can be a practical solution for real-time applications and increases the applicability of deep learning models. Moreover, the number of parameters of the model (3.8 M) is able to achieve a high test accuracy while keeping the computational load at reasonable levels, which further enhances the efficiency and applicability of the model.

Consequently, the balanced performance profile provided by the R3(2+1)D-SLR model positions it as a preferable choice for SLR tasks. Therefore, only the proposed R3(2+1)D-SLR network is used for SLR system design in the following sections of the paper.

4.2. Feature Fusion with Raw Data

Although body images capture all structures of a sign, reducing the resolution to 112×112 results in significant detail loss. This challenge is addressed in our study by separately analyzing each structure that represents the sign, including face and hands images, in addition to body images. Distinct models were trained for each of these features to harness their unique representational capabilities fully.

Our sign prediction strategy merges 128-dimensional feature vectors extracted from each model before the fully connected layer, creating a comprehensive 384-dimensional feature vector. This composite vector is subsequently utilized as the input for the SVM classifier, with the objective of capturing a holistic representation of sign language words. Figure 8 outlines our proposed system.

The test results of the models trained for each feature and the results of the test performed by classifying the feature fusion with SVM are shown in Tables 3 and 4.

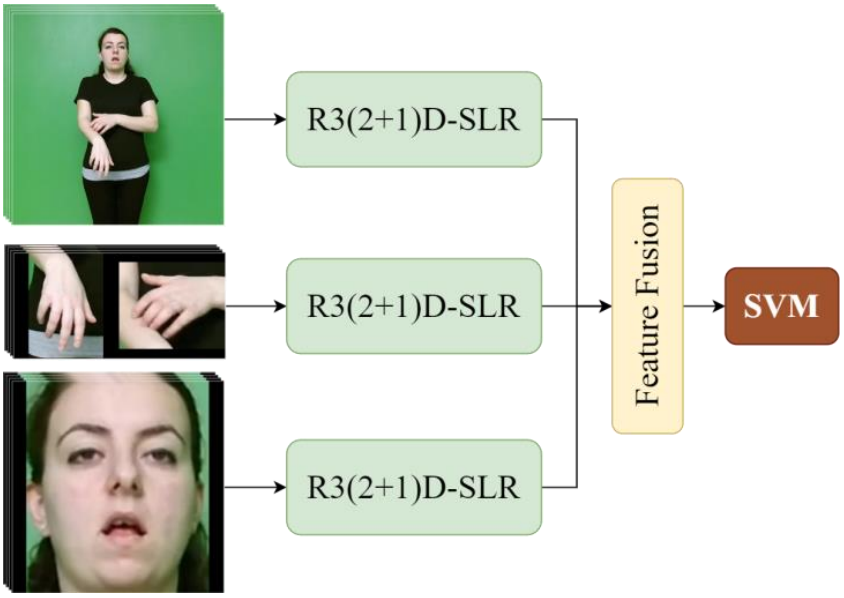

Figure 8. System diagram with raw data as input.

Table 3. Test accuracy of a system with raw data as input (BosphorusSign22k-general).

Feature	Test Accuracy (%)	
	No Background	With Background
Body	79.66	16.12
Hands	83.24	56.69
Face	22.23	19.81
Fusion + SVM	91.78	72.39

Table 4. Test accuracy of a system with raw data as input (LSA64).

Feature	Background Status	Test Accuracy (%)												
		E1	E2										E3	
			1	2	3	4	5	6	7	8	9	10	Mean	
Body	no background	98.90	97.50	94.06	97.50	96.87	99.37	99.37	94.37	96.24	96.24	99.37	97.09	99.53
	with background	72.65	58.43	31.25	44.68	59.37	57.49	72.18	55.62	51.87	64.99	72.50	56.84	66.09
Hands	no background	91.25	98.12	95.31	91.87	96.24	95.62	92.18	97.18	96.56	93.75	91.25	94.81	99.21
	with background	77.65	86.87	81.87	78.75	82.81	77.18	80.31	86.56	81.56	71.56	79.06	80.65	86.56
Face	no background	60.15	60.93	56.87	60.00	64.37	65.31	58.12	66.56	59.06	56.56	60.93	60.87	80.78
	with background	61.25	59.06	55.31	59.06	63.74	64.68	59.37	62.18	59.06	55.31	60.00	59.78	76.09
Fusion+ SVM	no background	99.37	99.68	99.37	99.06	99.37	99.37	99.68	99.68	100	99.37	99.68	99.53	99.84
	with background	95.31	93.43	84.68	93.43	94.68	95.62	97.81	95.62	95.93	91.87	95.93	93.90	98.90

Our experimental evaluation reveals the effectiveness of this feature fusion approach. As indicated in Table 3, hands images consistently outperformed full-body images in the BosphorusSign22k-general dataset, capturing distinctive features with a test accuracy of 83.24%. Although face images achieved a lower test accuracy of 22.23%, they were instrumental in identifying unique characteristics across 174 word classes. The integration of body, hands, and face images through the SVM classification resulted in a significant increase in test accuracy to 91.78%.

Further analysis was conducted with test sets modified by the addition of backgrounds, aiming to assess model robustness under varying conditions. The introduction of backgrounds led to a noticeable decline in test accuracy, underscoring the impact of

background complexity on model performance. Specifically, the accuracy for body images plummeted from 79.66% to 16.12%, and for hands images from 83.24% to 56.69% within the BosphorusSign22k-general dataset. Despite this, the fused features' classification via SVM demonstrated resilience, with the test accuracy reducing less dramatically from 91.78% to 72.39%, highlighting the robustness of our feature fusion approach against background variations.

Table 4 illuminates the efficacy of our system across signer-independent (E1 and E2) and signer-dependent (E3) evaluations within the LSA64 dataset. Specifically, the table shows the body image's superior performance compared to the hands image. Remarkably, the face image demonstrated significant representation for the 64 word classes, underscoring its substantial role in sign word recognition. The synthesized feature fusion, classified via SVM, achieved an impressive signer-independent test accuracy of 99.37% in E1, an average of 99.53% in E2, and 99.84% in the signer-dependent assessment of E3. The introduction of backgrounds to the LSA64 dataset test data precipitated a notable decline in accuracy, albeit the adverse effect was mitigated by the feature fusion strategy. This mitigation underscores the robustness of the fused feature approach, particularly when contrasted with the outcomes from the BosphorusSign22k-general dataset.

Our analysis underlines the need to consider both manual and non-manual features for a correct evaluation in SLR tasks. Our empirical evidence, derived from both datasets, confirms that a fusion of features markedly surpasses the performance of individual features. Moreover, the addition of a background to the test images led to a degradation in accuracy, attributed to the introduction of noise and extraneous information into the classification model. These results show that the background is important in the recognition of sign language words; therefore, the background should be taken into account when designing an SLR model.

4.3. Feature Fusion with Pose-Based Data

To build a more robust SLR system resilient to varying background images, we repeated the model training with pose images as outlined in Section 3.2.2 and illustrated in Figure 9. The test accuracy performance of the models is shown in Tables 5–7.

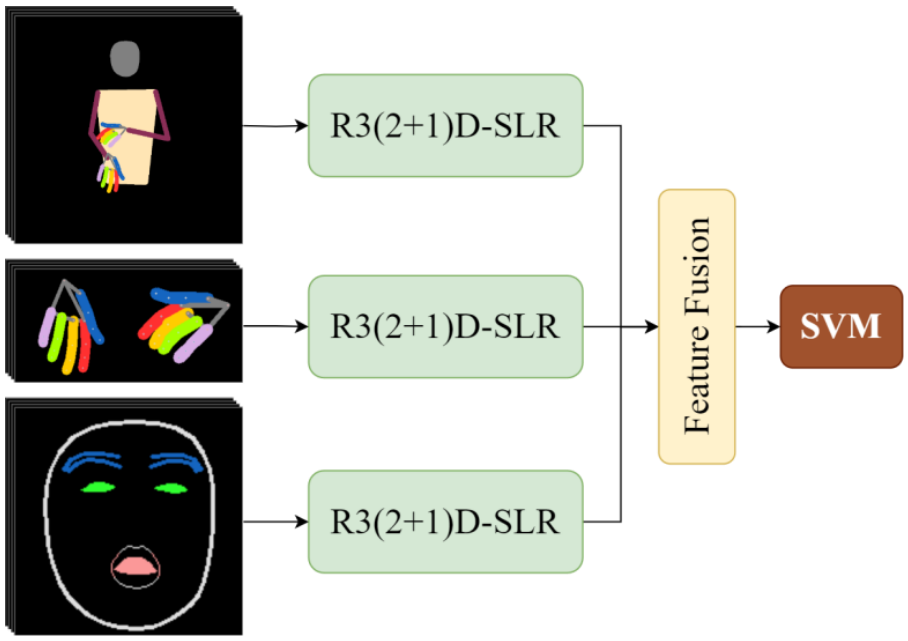

Figure 9. System diagram with pose data as input.

Table 5. Test accuracy of a system with pose data as input (BosphorusSign22k-general).

Feature	Test Accuracy (%)	
	No Background	With Background
Body	79.97	76.29
Hands	88.40	86.51
Face	10.64	9.79
Fusion + SVM	94.52	93.25

Table 6. Comparison of hands features.

Feature	Test Accuracy (%)
Raw image	83.24
Uncolored pose image	83.14
Colored pose image	88.40

Table 7. Test accuracy of a system with pose data as input (LSA64).

Feature	Background Status	Test Accuracy (%)											
		E1	E2										E3
			1	2	3	4	5	6	7	8	9	10	Mean	
Body	no background	94.99	93.75	96.24	89.06	94.99	98.43	95.62	92.81	96.56	93.12	97.81	94.84	99.37
	with background	93.59	92.50	94.99	84.06	93.75	97.18	94.37	89.06	94.06	91.25	94.99	92.62	98.75
Hands	no background	92.81	97.50	98.12	92.18	94.37	95.93	93.12	96.24	93.12	94.37	93.43	94.84	99.06
	with background	90.93	95.31	97.81	89.68	93.43	92.18	90.62	93.75	92.18	91.25	90.93	92.71	97.65
Face	no background	20.78	23.74	25.00	24.06	20.00	25.31	20.00	18.75	18.43	15.31	25.31	21.59	54.37
	with background	19.21	21.56	20.62	22.49	17.49	26.24	15.00	15.62	18.43	16.24	23.43	19.71	50.93
Fusion + SVM	no background	96.56	100	99.68	97.18	98.43	99.68	98.75	98.12	98.12	98.43	96.87	98.53	100
	with background	95.46	99.68	99.68	96.25	97.18	99.68	97.81	97.18	98.43	97.81	95.62	97.93	99.68

Table 5 demonstrates that, for the BosphorusSign22k-general dataset, pose images without backgrounds significantly improved performance for both body and hands features. In particular, there was a significant increase in accuracy in the hands pose image. Building on these insights, to explore the impact of finger color differentiation, we compared the accuracy of using colored versus uncolored hands pose images from the BosphorusSign22k-general dataset. Table 6, alongside Figure 10, presents the results for all hands data versions, showcasing the trained model's outcomes.

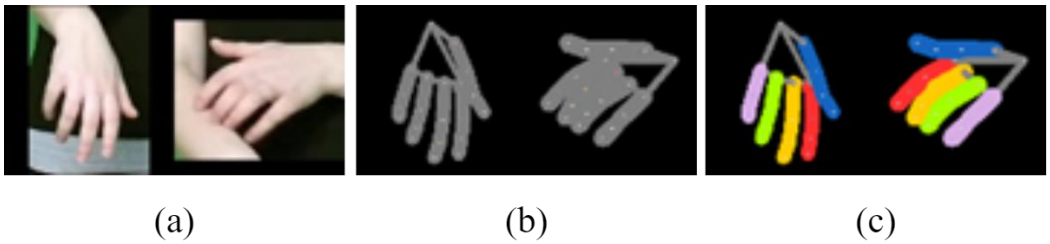

Figure 10. All versions of hands data: (**a**) raw data; (**b**) uncolored pose image; (**c**) colored pose image.

The color differentiation of fingers enhanced test accuracy by 5.26% over the uncolored version and 5.16% over the raw image version. This is because the finger's orientation, position, and shape are better distinguished as each finger is colored with a different color. Therefore, the hands pose image with colored fingers performs better than both the raw hands image and the uncolored hands pose image.

There was a decrease in accuracy for the face pose images obtained from the BosphorusSign22k-general dataset compared to the raw data. This is because although eyebrows, eyes, and lips are prominent in the face pose image, many details that could represent the sign are lost. Despite this challenge with face pose images, in the test with SVM classification of the fused features, there was an increase from 91.78% to 94.52%.

When the evaluations were performed using test images with a background, the test accuracy increased for all features except the face images. The accuracy of the proposed final system increased from 72.39% to 93.25%.

The experimental studies using pose images for the BosphorusSign22k-general dataset were repeated using the LSA64 dataset, and the results are reported in Table 7. For the LSA64 dataset, the use of pose data generally resulted in a decrease in accuracy in the evaluations with un-backgrounded test images. However, it was observed that using colored hands pose images generally provided a higher test accuracy than the raw images. There was a significant drop in accuracy for the face feature, as important details were lost in the pose data. In the tests performed with backgrounded test images, there was an increase in accuracy for all data types except for the face feature. For the proposed feature fusion-based system, although the accuracy decreased in the tests without background compared to the raw images, in the tests with background the signer-independent recognition accuracy increased from 95.31% to 95.46% for E1, from 93.90% to 97.93% for E2, and from 98.90% to 99.68% for signer-dependent (E3) recognition.

In summary, using pose images instead of normal images improved the accuracy of the final merged system in both datasets when evaluated with test images with backgrounds. Owing to the pose images, the system is robust to different backgrounds, making it more suitable for use in a real-world scenario. Furthermore, using pose data can eliminate disadvantageous factors for recognition accuracy due to different background and lighting conditions and shadows, as well as personal differences such as beards, mustaches, long hair, and different clothes.

4.4. Comparison with Other Studies

This section compares studies in the literature using BosphorusSign22k-general and LSA64 datasets and the proposed final SLR system's performances. Only the results obtained with the original test sets are included for comparison. Studies using the BosphorusSign22k-general dataset and test results of the proposed system are shown in Table 8.

Table 8. Comparison of the proposed system with results in the literature (BosphorusSign22k-general).

Studies	Test Accuracy (%)
Kindiroglu et al. (2019) [36]	81.58
Gündüz and Polat (2021) [19]	89.35
Our method (raw image-based input)	91.78
Our method (pose image-based input)	94.52

As can be seen from Table 8, while the study presented by Kindiroglu et al. [36] is based on pose data with an accuracy of 81.58%, Gündüz and Polat [19] achieved a higher accuracy rate of 89.35% by using multimodal data (RGB, pose, optical flow) and considering body, hands, and face data together. Our method surpassed existing studies, achieving a 91.78% accuracy with raw images and 94.52% with pose-based inputs. Moreover, the proposed R3(2+1)D-SLR network achieved an 88.40% test accuracy with the colored hands pose image. These results show that our model can effectively extract rich spatial and temporal features from both raw images and pose data. By fusing these features extracted from the body, hands, and face, the accuracy of the recognition was significantly improved.

For the LSA64 dataset, signer-dependent and signer-independent evaluations were performed to make comparisons with the studies in the literature. The results are shown in Table 9.

Table 9. Comparison of the proposed system with results in the literature (LSA64).

	Studies	Test Accuracy (%)
E1	Marais et al. (2022) [46]	74.22
	Our method (raw image-based input)	99.37
	Our method (pose image-based input)	96.56
E2	Ronchetti et al. (2016) [37]	91.7
	Rodriguez et al. (2018) [38]	85
	Alyami et al. (2023) [47]	91.09
	Our method (raw image-based input)	99.53
	Our method (pose image-based input)	98.53
E3	Ronchetti et al. (2016) [37]	97
	Konstantinidis et al. (2018) [39]	98.09
	Konstantinidis et al. (2018) [40]	99.84
	Zhang et al. (2019) [42]	99.063
	Imran and Raman (2020) [43]	97.81
	Marais et al. (2022) [45]	95.50
	Alyami et al. (2023) [47]	98.25
	Our method (raw image-based input)	99.84
	Our method (pose image-based input)	100

As can be seen from Table 9, the proposed feature fusion-based method outperformed the existing works in the literature using the LSA64 dataset both in E1 and E2, which are signer-independent evaluations, and in E3, which is a signer-dependent evaluation. Compared to the method proposed by Marais et al. [46] based on the InceptionV3-GRU method using the entire video image with signers 5 and 10 separated as a test, our R3(2+1)D-SLR model achieved a recognition accuracy of 94.99% with an increase of 5.44% for the same input modality. This shows the superiority of our proposed deep learning model over InceptionV3-GRU in spatial and temporal feature extraction. This result increased up to 99.37% with the inclusion of hands and face data. While the 91.09% accuracy rate obtained by Alyami et al. [47] in E2, another signer-independent experimental environment, shows the strengths of transformer models, our model achieved a 99.53% accuracy for the raw-image input and a 98.53% accuracy for the pose-based input on the LSA64 dataset, which provides a significant advantage, especially for in-depth processing of spatio-temporal features. In E3, a signer-dependent evaluation, the 99.84% accuracy rate presented by Konstantinidis et al. [40] is a remarkable achievement in SLR. This paper proposes a model based on VGG-16 and LSTM that thoroughly examines body, hands, and face data using a multimodal approach covering RGB, pose, and optical flow data. However, our proposed method achieved a 99.84% accuracy for raw image-based input and 100% accuracy for pose-based input, which emphasizes the superiority of our method. In addition, the recognition accuracy of 99.53% achieved by our proposed R3(2+1)D-SLR model with RGB body data emphasizes the importance of the effective use of deep learning models.

5. Conclusions

In summary, our novel R3(2+1)D-SLR network marks a significant advancement in SLR by effectively merging R3D and R(2+1)D convolution blocks. This innovative approach facilitates a deeper understanding and a more accurate capture of sign language's complex spatial and temporal dynamics. Together with this proposed network, our comprehensive methodology integrating the fusion of body, hands, and face features based on pose data consistently outperformed and surpasses the work in the existing literature on various datasets and under different background conditions.

Our future work will focus on exploring alternative classifiers to the SVM used in our proposed SLR system and optimizing their hyperparameters. In particular, going beyond direct feature fusion, we aim to explore the potential of alternative ensemble techniques such as boosting, bagging, and stacking to more effectively integrate and evaluate different feature sets. We will also focus on the robustness and adaptability of the SLR system, examining the integration of additional data augmentation techniques in addition to background variations to increase the model's adaptability to various environmental conditions. Together, these efforts aim to propel SLR technology forward, improving communication access and inclusivity for the deaf and hard-of-hearing.

Author Contributions: Conceptualization, A.A. and O.K.B.; methodology, A.A. and O.K.B.; software, A.A. and O.K.B.; validation, A.A. and O.K.B.; formal analysis, A.A. and O.K.B.; investigation, A.A. and O.K.B.; writing—original draft preparation, A.A. and O.K.B.; writing—review and editing, A.A. and O.K.B.; visualization, A.A.; supervision, O.K.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The BosphorusSign22k-general dataset [33,34] used in this study can be obtained from the dataset creators upon reasonable request. The dataset creators can be contacted for access through the following link: https://ogulcanozdemir.github.io/bosphorussign22k/ (accessed on 15 September 2023). The LSA64 dataset [35] is available through this link: https://facundoq.github.io/datasets/lsa64/ (accessed on 13 October 2023).

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

References

1. International Day of Sign Languages. Available online: https://www.un.org/en/observances/sign-languages-day (accessed on 10 January 2024).
2. Sreemathy, R.; Turuk, M.; Kulkarni, I.; Khurana, S. Sign Language Recognition Using Artificial Intelligence. *Educ. Inf. Technol.* **2023**, *28*, 5259–5278. [CrossRef]
3. Mukushev, M.; Sabyrov, A.; Imashev, A.; Koishybay, K.; Kimmelman, V.; Sandygulova, A. Evaluation of Manual and Non-Manual Components for Sign Language Recognition. In Proceedings of the LREC 2020—12th International Conference on Language Resources and Evaluation, Marseille, France, 11–16 May 2020.
4. Rastgoo, R.; Kiani, K.; Escalera, S. Sign Language Recognition: A Deep Survey. *Expert. Syst. Appl.* **2021**, *164*, 113794. [CrossRef]
5. Das, S.; Biswas, S.K.; Purkayastha, B. A Deep Sign Language Recognition System for Indian Sign Language. *Neural Comput. Appl.* **2022**, *35*, 1469–1481. [CrossRef]
6. Munsif, M.; Khan, S.U.; Khan, N.; Baik, S.W. Attention-Based Deep Learning Framework for Action Recognition in a Dark Environment. *Hum. Centric Comput. Inf. Sci.* **2024**, *14*, 1–22.
7. Zhang, Y.; Deng, L.; Zhu, H.; Wang, W.; Ren, Z.; Zhou, Q.; Lu, S.; Sun, S.; Zhu, Z.; Gorriz, J.M.; et al. Deep Learning in Food Category Recognition. *Inf. Fusion* **2023**, *98*, 101859. [CrossRef]
8. Nogales, R.E.; Benalcázar, M.E. Hand Gesture Recognition Using Automatic Feature Extraction and Deep Learning Algorithms with Memory. *Big Data Cogn. Comput.* **2023**, *7*, 102. [CrossRef]
9. Aslani, S.; Jacob, J. Utilisation of Deep Learning for COVID-19 Diagnosis. *Clin. Radiol.* **2023**, *78*, 150–157. [CrossRef] [PubMed]
10. Tolentino, L.K.S.; Serfa Juan, R.O.; Thio-ac, A.C.; Pamahoy, M.A.B.; Forteza, J.R.R.; Garcia, X.J.O. Static Sign Language Recognition Using Deep Learning. *Int. J. Mach. Learn. Comput.* **2019**, *9*, 821–827. [CrossRef]
11. Wadhawan, A.; Kumar, P. Deep Learning-Based Sign Language Recognition System for Static Signs. *Neural Comput. Appl.* **2020**, *32*, 7957–7968. [CrossRef]
12. Damaneh, M.M.; Mohanna, F.; Jafari, P. Static Hand Gesture Recognition in Sign Language Based on Convolutional Neural Network with Feature Extraction Method Using ORB Descriptor and Gabor Filter. *Expert. Syst. Appl.* **2023**, *211*, 118559. [CrossRef]
13. Yang, S.; Zhu, Q. Continuous Chinese Sign Language Recognition with CNN-LSTM. In Proceedings of the Ninth International Conference on Digital Image Processing (ICDIP 2017), Hong Kong, China, 19–22 May 2017; Volume 10420.
14. Camgoz, N.C.; Koller, O.; Hadfield, S.; Bowden, R. Multi-Channel Transformers for Multi-Articulatory Sign Language Translation. In Proceedings of the Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), Glasgow, UK, 23–28 August 2020; Volume 12538.

15. Ji, S.; Xu, W.; Yang, M.; Yu, K. 3D Convolutional Neural Networks for Human Action Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *35*, 221–231. [CrossRef] [PubMed]
16. Tran, D.; Wang, H.; Torresani, L.; Ray, J.; Lecun, Y.; Paluri, M. A Closer Look at Spatiotemporal Convolutions for Action Recognition. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018.
17. Sarhan, N.; Frintrop, S. Transfer Learning for Videos: From Action Recognition to Sign Language Recognition. In Proceedings of the 2020 IEEE International Conference on Image Processing (ICIP), Abu Dhabi, United Arab Emirates, 25–28 October 2020.
18. Gökçe, Ç.; Özdemir, O.; Kındıroğlu, A.A.; Akarun, L. Score-Level Multi Cue Fusion for Sign Language Recognition. In Proceedings of the Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), Glasgow, UK, 23–28 August 2020; Volume 12536.
19. Gündüz, C.; Polat, H. Turkish Sign Language Recognition Based on Multistream Data Fusion. *Turk. J. Electr. Eng. Comput. Sci.* **2021**, *29*, 1171–1186. [CrossRef]
20. Huang, M.; Qian, H.; Han, Y.; Xiang, W. R(2+1)D-Based Two-Stream CNN for Human Activities Recognition in Videos. In Proceedings of the 2021 40th Chinese Control Conference (CCC), Shanghai, China, 26–28 July 2021; Volume 2021.
21. Han, X.; Lu, F.; Yin, J.; Tian, G.; Liu, J. Sign Language Recognition Based on R(2+1)D with Spatial-Temporal-Channel Attention. *IEEE Trans. Hum. Mach. Syst.* **2022**, *52*, 687–698. [CrossRef]
22. Wang, F.; Du, Y.; Wang, G.; Zeng, Z.; Zhao, L. (2+1)D-SLR: An Efficient Network for Video Sign Language Recognition. *Neural Comput. Appl.* **2021**, *34*, 2413–2423. [CrossRef]
23. Yang, B.; Zhou, P. Mixed 3D-(2+1)D Convolution for Action Recognition. In Proceedings of the Eleventh International Conference on Digital Image Processing (ICDIP 2019), Guangzhou, China, 14 August 2019.
24. Zhou, Z.; Lui, K.S.; Tam, V.W.L.; Lam, E.Y. Applying (3+2+1)D Residual Neural Network with Frame Selection for Hong Kong Sign Language Recognition. In Proceedings of the 2020 25th International Conference on Pattern Recognition (ICPR), Milan, Italy, 10–15 January 2020.
25. Podder, K.K.; Ezeddin, M.; Chowdhury, M.E.H.; Sumon, M.S.I.; Tahir, A.M.; Ayari, M.A.; Dutta, P.; Khandakar, A.; Mahbub, Z.B.; Kadir, M.A. Signer-Independent Arabic Sign Language Recognition System Using Deep Learning Model. *Sensors* **2023**, *23*, 7156. [CrossRef] [PubMed]
26. Kumar, P.; Roy, P.P.; Dogra, D.P. Independent Bayesian Classifier Combination Based Sign Language Recognition Using Facial Expression. *Inf. Sci.* **2018**, *428*, 30–48. [CrossRef]
27. Irasiak, A.; Kozak, J.; Piasecki, A.; Stęclik, T. Processing Real-Life Recordings of Facial Expressions of Polish Sign Language Using Action Units. *Entropy* **2023**, *25*, 120. [CrossRef]
28. Özdemir, O.; Baytaş, İ.M.; Akarun, L. Multi-Cue Temporal Modeling for Skeleton-Based Sign Language Recognition. *Front. Neurosci.* **2023**, *17*, 8191. [CrossRef]
29. Javaid, S.; Rizvi, S. Manual and Non-Manual Sign Language Recognition Framework Using Hybrid Deep Learning Techniques. *J. Intell. Fuzzy Syst.* **2023**, *45*, 3823–3833. [CrossRef]
30. Tian, Y.; Han, F.; Zhu, M.; Xu, X.; Li, Y. Research on Sign Language Gesture Division and Gesture Extraction in Complex Background. In *International Conference on Computer Vision, Application, and Algorithm (CVAA 2022)*; SPIE: Bellingham, WA, USA, 2023.
31. Hamada, Y.; Shimada, N.; Shirai, Y. Hand Shape Estimation under Complex Backgrounds for Sign Language Recognition. In Proceedings of the Sixth IEEE International Conference on Automatic Face and Gesture Recognition, Seoul, Republic of Korea, 19 May 2004.
32. Noriaki, H.; Yamamoto, M. Real-Time Isolated Sign Language Recognition. In *International Conference on Frontiers of Artificial Intelligence, Ethics, and Multidisciplinary Applications*; Farmanbar, M., Tzamtzi, M., Verma, A.K., Chakravorty, A., Eds.; Springer Nature: Singapore, 2024; pp. 445–458.
33. Camgoz, N.C.; Kindiroglu, A.A.; Karabüklü, S.; Kelepir, M.; Sumru Ozsoy, A.; Akarun, L. BosphorusSign: A Turkish Sign Language Recognition Corpus in Health and Finance Domains. In Proceedings of the 10th International Conference on Language Resources and Evaluation, Portorož, Slovenia, 23–28 May 2016.
34. Özdemir, O.; Kindiroglu, A.A.; Camgöz, N.C.; Akarun, L. BosphorusSign22k Sign Language Recognition Dataset. *arXiv* **2020**, arXiv:2004.01283.
35. Ronchetti, F.; Quiroga, F.; Lanzarini, L. LSA64: An Argentinian Sign Language Dataset. *arXiv* **2016**, arXiv:2310.17429.
36. Kindiroglu, A.A.; Ozdemir, O.; Akarun, L. Temporal Accumulative Features for Sign Language Recognition. In Proceedings of the 2019 International Conference on Computer Vision Workshop, ICCVW 2019, Seoul, Republic of Korea, 27–28 October 2019.
37. Ronchetti, F.; Quiroga, F.; Estrebou, C.; Lanzarini, L.; Rosete, A. Sign Languague Recognition without Frame-Sequencing Constraints: A Proof of Concept on the Argentinian Sign Language. In Proceedings of the Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), San José, Costa Rica, 23–25 November 2016; Volume 10022.
38. Rodríguez, J.; Martínez, F. Towards On-Line Sign Language Recognition Using Cumulative SD-VLAD Descriptors. In Proceedings of the Communications in Computer and Information Science, Cartagena, Colombia, 26–28 September 2018; Volume 885.
39. Konstantinidis, D.; Dimitropoulos, K.; Daras, P. Sign Language Recognition Based on Hand and Body Skeletal Data. In Proceedings of the 3DTV-Conference, Helsinki, Finland, 3–5 June 2018; Volume 2018.

40. Konstantinidis, D.; Dimitropoulos, K.; Daras, P. A Deep Learning Approach for Analyzing Video and Skeletal Features in Sign Language Recognition. In Proceedings of the 2018 IEEE International Conference on Imaging Systems and Techniques (IST), Krakow, Poland, 16–18 October 2018.
41. Masood, S.; Srivastava, A.; Thuwal, H.C.; Ahmad, M. Real-Time Sign Language Gesture (Word) Recognition from Video Sequences Using CNN and RNN. In *Advances in Intelligent Systems and Computing*; Springer: Singapore, 2018; Volume 695.
42. Zhang, X.; Li, X. Dynamic Gesture Recognition Based on MEMP Network. *Future Internet* **2019**, *11*, 91. [CrossRef]
43. Imran, J.; Raman, B. Deep Motion Templates and Extreme Learning Machine for Sign Language Recognition. *Vis. Comput.* **2020**, *36*, 1233–1246. [CrossRef]
44. Elsayed, E.K.; Fathy, D.R. Semantic Deep Learning to Translate Dynamic Sign Language. *Int. J. Intell. Eng. Syst.* **2020**, *14*, 316–325. [CrossRef]
45. Marais, M.; Brown, D.; Connan, J.; Boby, A. An Evaluation of Hand-Based Algorithms for Sign Language Recognition. In Proceedings of the 2022 International Conference on Artificial Intelligence, Big Data, Computing and Data Communication Systems (icABCD), Durban, South Africa, 4–5 August 2022; pp. 1–6.
46. Marais, M.; Brown, D.; Connan, J.; Boby, A.; Kuhlane, L. Investigating Signer-Independent Sign Language Recognition on the LSA64 Dataset. In *Southern Africa Telecommunication Networks and Applications Conference (SA TNAC)*; Rhodes University: Grahamstown, South Africa, 2022.
47. Alyami, S.; Luqman, H.; Hammoudeh, M. Isolated Arabic Sign Language Recognition Using A Transformer-Based Model and Landmark Keypoints. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.* **2023**, *23*, 1–19. [CrossRef]
48. Rastgoo, R.; Kiani, K.; Escalera, S. Real-Time Isolated Hand Sign Language Recognition Using Deep Networks and SVD. *J. Ambient. Intell. Humaniz. Comput.* **2021**, *13*, 591–611. [CrossRef]
49. Samaan, G.H.; Wadie, A.R.; Attia, A.K.; Asaad, A.M.; Kamel, A.E.; Slim, S.O.; Abdallah, M.S.; Cho, Y.I. MediaPipe's Landmarks with RNN for Dynamic Sign Language Recognition. *Electronics* **2022**, *11*, 3228. [CrossRef]
50. de Castro, G.Z.; Guerra, R.R.; Guimarães, F.G. Automatic Translation of Sign Language with Multi-Stream 3D CNN and Generation of Artificial Depth Maps. *Expert. Syst. Appl.* **2023**, *215*, 119394. [CrossRef]
51. Hamza, H.M.; Wali, A. Pakistan Sign Language Recognition: Leveraging Deep Learning Models with Limited Dataset. *Mach. Vis. Appl.* **2023**, *34*, 71. [CrossRef]
52. Laines, D.; Gonzalez-Mendoza, M.; Ochoa-Ruiz, G.; Bejarano, G. Isolated Sign Language Recognition Based on Tree Structure Skeleton Images. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 17–24 June 2023; pp. 276–284.
53. Jebali, M.; Dakhli, A.; Bakari, W. Deep Learning-Based Sign Language Recognition System Using Both Manual and Non-Manual Components Fusion. *AIMS Math.* **2024**, *9*, 2105–2122. [CrossRef]
54. Lugaresi, C.; Tang, J.; Nash, H.; McClanahan, C.; Uboweja, E.; Hays, M.; Zhang, F.; Chang, C.-L.; Yong, M.G.; Lee, J.; et al. MediaPipe: A Framework for Building Perception Pipelines. *arXiv* **2019**, arXiv:1906.08172.
55. Grishchenko, I.; Bazarevsky, V. MediaPipe Holistic—Simultaneous Face, Hand and Pose Prediction, on Device. Available online: https://ai.googleblog.com/2020/12/mediapipe-holistic-simultaneous-face.html (accessed on 11 January 2022).
56. Chai, J.; Zeng, H.; Li, A.; Ngai, E.W.T. Deep learning in computer vision: A critical review of emerging techniques and application scenarios. *Mach. Learn. Appl.* **2021**, *6*, 100134. [CrossRef]
57. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for Image Recognition. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; Volume 2016.
58. Huang, G.; Liu, Z.; Weinberger, K.Q. Densely Connected Convolutional Networks. *arXiv* **2016**, arXiv:1608.06993.
59. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L.C. MobileNetV2: Inverted Residuals and Linear Bottlenecks. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018.
60. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. In Proceedings of the 3rd International Conference on Learning Representations, ICLR 2015—Conference Track Proceedings, San Diego, CA, USA, 7–9 May 2015.
61. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. SSD: Single Shot Multibox Detector. In Proceedings of the Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), Amsterdam, The Netherlands, 11–14 October 2016; Volume 9905.
62. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1137–1149. [CrossRef] [PubMed]
63. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; Volume 2016.
64. LeCun, Y.; Bengio, Y.; Hinton, G. Deep Learning. *Nature* **2015**, *521*, 436–444. [CrossRef]
65. Tran, D.; Bourdev, L.; Fergus, R.; Torresani, L.; Paluri, M. Learning Spatiotemporal Features with 3D Convolutional Networks. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015; pp. 4489–4497.

66. Carreira, J.; Zisserman, A. Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset. In Proceedings of the 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, 21–26 July 2017; Volume 2017.
67. Köpüklü, O.; Kose, N.; Gunduz, A.; Rigoll, G. Resource Efficient 3D Convolutional Neural Networks. In Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops, Montreal, BC, Canada, 11–17 October 2021.
68. Taylor, G.W.; Fergus, R.; LeCun, Y.; Bregler, C. Convolutional Learning of Spatio-Temporal Features. In Proceedings of the Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), Heraklion, Crete, Greece, 5–11 September 2010; Volume 6316.
69. Ioffe, S.; Szegedy, C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6–11 July 2015; Volume 1.
70. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet Classification with Deep Convolutional Neural Networks. In *Advances in Neural Information Processing Systems*; Pereira, F., Burges, C.J., Bottou, L., Weinberger, K.Q., Eds.; Curran Associates, Inc.: New York, NY, USA, 2012; Volume 25.
71. Cortes, C.; Vapnik, V. Support-Vector Networks. *Mach. Learn.* **1995**, *20*, 273–297. [CrossRef]
72. Amari, S.; Wu, S. Improving Support Vector Machine Classifiers by Modifying Kernel Functions. *Neural Netw.* **1999**, *12*, 783–789. [CrossRef] [PubMed]
73. Hussain, S.F. A Novel Robust Kernel for Classifying High-Dimensional Data Using Support Vector Machines. *Expert. Syst. Appl.* **2019**, *131*, 116–131. [CrossRef]
74. Barbiero, P.; Squillero, G.; Tonda, A. Modeling Generalization in Machine Learning: A Methodological and Computational Study. *arXiv* **2020**, arXiv:2006.15680.
75. Behzad, M.; Asghari, K.; Eazi, M.; Palhang, M. Generalization Performance of Support Vector Machines and Neural Networks in Runoff Modeling. *Expert. Syst. Appl.* **2009**, *36*, 7624–7629. [CrossRef]
76. Hannan, S.A.; Pushparaj Ashfaque, M.W.; Lamba, A.; Kumar, A. Analysis of Detection and Recognition of Human Face Using Support Vector Machine. In *Artificial Intelligence of Things*; Challa, R.K., Aujla, G.S., Mathew, L., Kumar, A., Kalra, M., Shimi, S.L., Saini, G., Sharma, K., Eds.; Springer Nature: Cham, Switzerland, 2024; pp. 86–98.
77. Park, J.; Choi, Y.; Byun, J.; Lee, J.; Park, S. Efficient Differentially Private Kernel Support Vector Classifier for Multi-Class Classification. *Inf. Sci.* **2023**, *619*, 889–907. [CrossRef]
78. Zhang, L.; Zhu, G.; Shen, P.; Song, J.; Shah, S.A.; Bennamoun, M. Learning Spatiotemporal Features Using 3DCNN and Convolutional LSTM for Gesture Recognition. In Proceedings of the 2017 IEEE International Conference on Computer Vision Workshops, ICCVW 2017, Venice, Italy, 22–29 October 2017; Volume 2018.
79. Myagila, K.; Kilavo, H. A Comparative Study on Performance of SVM and CNN in Tanzania Sign Language Translation Using Image Recognition. *Appl. Artif. Intell.* **2022**, *36*, e2005297. [CrossRef]