Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2021) 29: 1171 – 1186
© TÜBİTAK
doi:10.3906/elk-2005-156

Turkish sign language recognition based on multistream data fusion

Cemil GÜNDÜZ[1*], **Hüseyin POLAT**[2]
[1]Department of Information Systems, Graduate School of Informatics, Gazi University, Ankara, Turkey
[2]Department of Computer Engineering, Faculty of Technology, Gazi University, Ankara, Turkey

Received: 21.05.2020 • **Accepted/Published Online:** 19.01.2021 • **Final Version:** 30.03.2021

Abstract: Sign languages are nonverbal, visual languages that hearing- or speech-impaired people use for communication. Aside from hands, other communication channels such as body posture and facial expressions are also valuable in sign languages. As a result of the fact that the gestures in sign languages vary across countries, the significance of communication channels in each sign language also differs. In this study, representing the communication channels used in Turkish sign language, a total of 8 different data streams—4 RGB, 3 pose, 1 optical flow—were analyzed. Inception 3D was used for RGB and optical flow; and LSTM-RNN was used for pose data streams. Experiments were conducted by merging the data streams in different combinations, and then a sign language recognition system that merged the most suitable streams with the help of a multistream late fusion mechanism was proposed. Considering each data stream individually, the accuracies of the RGB streams were between 28% and 79%; pose stream accuracies were between 9% and 50%; and optical flow data accuracy was 78.5%. When these data streams were used in combination, the sign language recognition performance was higher in comparison to any of the data streams alone. The proposed sign language recognition system uses a multistream data fusion mechanism and gives an accuracy of 89.3% on BosphorusSign General dataset. The multistream data fusion mechanisms have a great potential for improving sign language recognition results.

Key words: Deep learning, sign language recognition, 3D convolutional neural networks, long short-term memory, recurrent neural networks

1. Introduction

Sign languages are the main communication media that hearing-impaired people use in their daily lives. Signs are expressed in more than one communication channel in sign languages. These channels might be body posture and facial expressions in addition to the gestures of fingers, hands, arms, and head [1]. While some of these channels are essential for a sign, some others are used to strengthen the meaning of a sign or to modify it [2].

Sign languages are not universal because they appear and develop naturally in hearing-impaired societies. According to World Federation of the Deaf, there are over 300 sign languages in the world[1]. Each of these languages is different from one another in terms of the signs used and linguistic patterns. Zeshan [3] states that even regional variations can be observed in a country. Apart from the variations among different sign languages, there are linguistic variations between the sign language and the verbal language used in the same country. All these facts make sign languages harder to be understood by hearing people. Sign languages are structurally complex, too. This reduces the number of hearing people who can understand a sign language in a society, which leads to a communication barrier between hearing people and hearing-impaired people [4, 5].

*Correspondence: cemil.gunduz@gazi.edu.tr

[1] World Federation of Deaf (2020). Our Work [online]. Website https: //wfdeaf.org/our-work/ [accessed 13 January 2020].

This work is licensed under a Creative Commons Attribution 4.0 International License.

The field of sign language recognition (SLR) includes topics such as natural language processing, video processing, pattern recognition, and linguistics, and it is a subfield of human–computer interaction. Sign language recognition studies are highly promising for overcoming the communication barrier between hearing and hearing-impaired people. Developing an SLR system, which turns signs into text or speech, will facilitate the communication between hearing and hearing-impaired people. Additionally, SLR systems provide an interactive environment for learning sign languages.

Majority of the studies on SLR focus on the shapes of fingers and movements of hands. However, since signs can be expressed via more than one communication channel, examining solely fingers and hands in an effort to interpret a sign will not be enough. For an effective and successful sign recognition; facial expressions, head and body movements, and posture should be considered, too.

Turkish sign language is the nonverbal language used by hearing- and speech-impaired people in Turkey. In this study, by using Inception 3D and Long Short-Term Memory-Recurrent Neural Network (LSTM-RNN), the impact of communication channels used in Turkish sign language on the sign recognition achievement was evaluated, and a multistream fusion-based SLR system which uses the most suitable streams was proposed. In order to interpret the position of the body and hands through joints, skeletal data extracted through pose estimation were used. To interpret the expressions of body, face, and hands, four different RGB video images were used. In order to interpret the movements while signing, we extracted optical flow from body videos. This methodology was followed so that the maximum number of channels used in sign language communication could be tested.

The rest of the paper is structured as follows: Section 2 presents the literature review on sign language recognition. Section 3 details the proposed SLR system. Section 4 presents the experiments using the proposed SLR system and the experiment results. In Section 5, the conclusions of the study results are discussed.

2. Related work

Sign language recognition is similar to fields such as action recognition or gesture recognition [6]. However, SLR is stated to be more complicated since sign languages are highly structured [7].

Sign language recognition studies can be categorized into 2 groups: sensor-based and vision-based. Sensor-based sign language refers to the approach where data are collected through wearable devices such as data gloves, accelerometers, and electromyograms. With the help of these sensors, it is possible to directly measure the motion, speed, and position of hands and fingers; and this provides us with more successful and robust recognition. In sensor-based sign language recognition approach, auxiliary equipment on signers makes the signing harder [8]. Thus, it is stated that sensor-based methods are not appropriate for an SLR system to be used in daily life [7].

As for vision-based approach, sign recognition is performed through processing of video camera recordings. These cameras can be depth cameras or other equipment such as Leap Motion, as well as standard RGB cameras. Vision-based sign language recognition can be achieved through continuous or isolated sign videos. Isolated sign language recognition videos have only one sign in each video, whereas continuous sign language videos have multiple signs.

In vision-based sign language recognition, firstly, hands and face were segmented, tracked, and then feature extraction was performed through algorithms such as SIFT and SURF. Later, classification was done through methods such as support vector machines [9], artificial neural networks [10], and hidden Markov models (HMMs) [11]. In this approach, hand-crafted feature extraction could become cumbersome, and classification performance was directly dependent on the extracted features. Due to their characteristics of being able to model temporal information successfully [12], HMMs and modified HMMs were often preferred for the classification [11, 13–16].

Along with the advances including more powerful computer hardware, bigger datasets, and more affordable hardware prices [17], deep learning techniques have come into use as they do in other fields such as image classification and action recognition.

In the early studies of deep learning in SLR, feature extraction from spatial information was done through convolutional layers while temporal information were learned with the help of HMMs [18] or RNNs [4]. Later, the models which use 3D convolutional layers to learn spatiotemporal information in videos have been developed [19–23].

In recent SLR studies, researchers aimed to increase the recognition performance through multimodal data. Kumar et al. [1] used Kinect and Leap Motion sensors for data acquisition and achieved 96.33% accuracy on a 50-sign Indian sign language dataset when they used both of the data modalities. Ferreira et al. [24] developed a multimodal SLR system to recognize 10 motionless signs in the American sign language. In their study, they used RGB, depth, and 3D skeletal data obtained by Leap Motion. Their results suggested that using multiple data types had a positive impact on sign language recognition.

Zhang et al. [25] used RGB and depth images together in their study and gained a 6% of improvement compared to the sole use of RGB. Moreover, they reported that depth images were more robust against the changes in the light and environment; and they were able to capture the signs better.

Camgoz et al. [26] analyzed hand data in another stream in addition to the body video images through Bidirectional-LSTM (BLSTM) layers with the purpose of continuous sign language recognition. They got the highest accuracy when they used two streams for hand and full body videos, then merged and trained them using a third BLSTM layer.

Skeletal data is another data type which can be used to analyze body movements in sign languages. Liu et al. [4] aimed for sign language recognition by using only joint points. Using LSTM-RNN, they analyzed the joint points obtained via a Kinect camera, and they got an accuracy of 86.2% in a dataset of 100 signs in Chinese sign language, and 63.3% in a dataset of 500 classes.

When it is not possible to directly obtain skeletal data through devices like Kinect, pose estimation is used. With the recent developments in pose estimation, SLR researchers have developed pose estimation systems that focus on upper body only. Charles et al. [27] developed a pose estimation system that automatically separates the signer from dynamic backgrounds of TV broadcasts and estimates joint locations using a random forest regressor. Gattupalli et al. [28] proposed a high-performance pose estimation system in American sign language by retraining Deeppose pose estimation model through transfer learning.

Konstantinidis et al. [29] reached an accuracy of 98.09% in sign language recognition in a dataset of 64 Argentinian sign language signs by using stacked LSTM layers on body and hand joint data which was obtained through pose estimation. Escobedo et al. [2] achieved 99.91% accuracy using the same dataset by creating texture maps in addition to the depth and RGB data.

In a study in Turkish sign language [30], 99.25% accuracy was achieved by projecting the skeletal data gathered via a Kinect camera on the video of a 9-class-dataset. Ozdemir et al. [31] proposed an SLR system using features obtained from histogram of oriented gradients, histogram of optical flow, and motion boundary histograms. They achieved 97.87% accuracy on a 10-sign-subset of BosphorusSign dataset. Kindiroglu et al.

[32] achieved up to 81.58% accuracy in BosphorusSign General dataset by using pose data and body RGB images with the temporal accumulative features method.

Tamer et al. [33], making use of pose data extracted from videos, aimed for sign retrieval via dynamic time warping, and they suggested that pose data were more successful than any other channels they had used, and it worked more robustly against signer difference.

3. Materials and methodology

3.1. Dataset

In this study, we used Turkish Sign Language Dataset BosphorusSign [34] since it is the most comprehensive dataset in Turkish Sign Language. BosphorusSign has three subsets for various applications: finance, health, and general. We used ”General” subset of this dataset in this study. There are 154 classes in the subset for general purpose applications; and in each class, there are at least 29 videos, which makes 5829 sign language videos in total. Isolated sign language videos start from a hands down rested position; sign is performed, and then hands go back to the starting position. Videos are recorded in a studio in front of a still background with 1920*1080 resolution and 30 frames per second. Each sign is recorded by 6 signers in various repetitions. All signers in this dataset are right-handed; therefore, we refer the right hand as the dominant hand. In order to split the training and testing data, we followed the instructions given by the data provider. Accordingly, the videos of the 4th signer (User-4) were allocated for testing, and other signers' videos were used as the training data. The histogram of the frame numbers in the dataset is given in Figure 1.

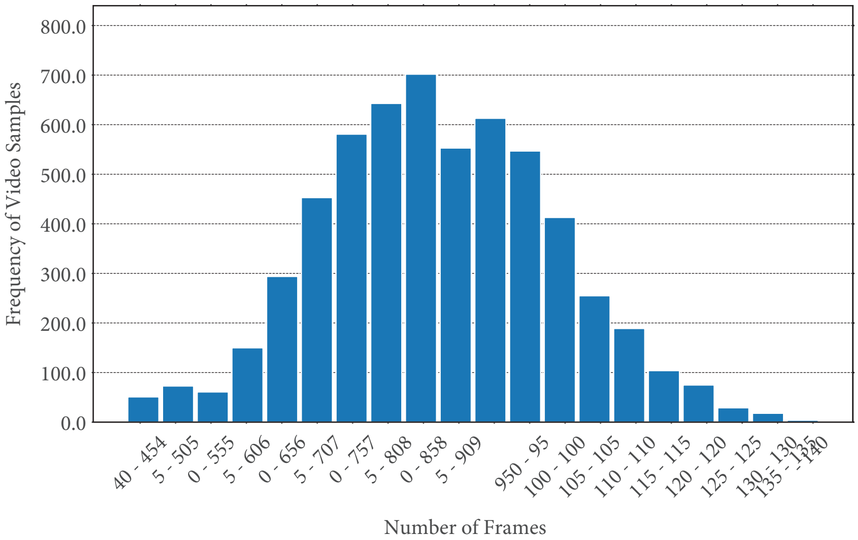

Figure 1. Histogram of frame numbers in the dataset.

3.2. Data preprocessing

Sign language recognition studies have long focused on hands. However, interpreting signs might not be possible only with hands. In order to interpret most of the signs, upper-body needs to be seen [5]. Therefore, the proposed SLR system in this study employs upper-body RGB videos. Data preprocessing is needed to extract data streams from RGB videos for training the deep learning model.

Most deep learning methods require fixed input sizes and large amounts of data in order to work robustly. Thus, data augmentation is often resorted in the studies. Data augmentation outputs the data at a fixed size as required by deep learning model while increasing the number of samples.

In static images, data augmentation can be done in various ways such as cropping, zooming, and rotating. As for videos, temporal data augmentation is also possible in addition to the spatial data augmentation. For temporal data augmentation in videos, various techniques can be used. One of those techniques is taking sequential frames of the videos at a certain number. However, it is not suitable in isolated sign language recognition as this does not make use of whole video information. Another technique is doing sampling by taking a certain number of frames at equal intervals from videos. When the length of the video is not enough for the fixed number of frames to be taken, the first and/or the last frames can be repeated until the video becomes long enough [22, 35].

In this study, a temporal data augmentation was preferred by sampling 32 frames at equal intervals regardless of the number of video frames. No spatial data augmentation was done, but a square-shaped area in the middle of the videos was cropped and resized so that the frames were all at equal width and height.

RGB sign language videos contain numerous information channels which are also important in sign language recognition. In this study, pose and optical flow data required for the proposed SLR system were extracted from the RGB videos through data preprocessing. Data augmentation and data preprocessing procedures are presented in Figure 2.

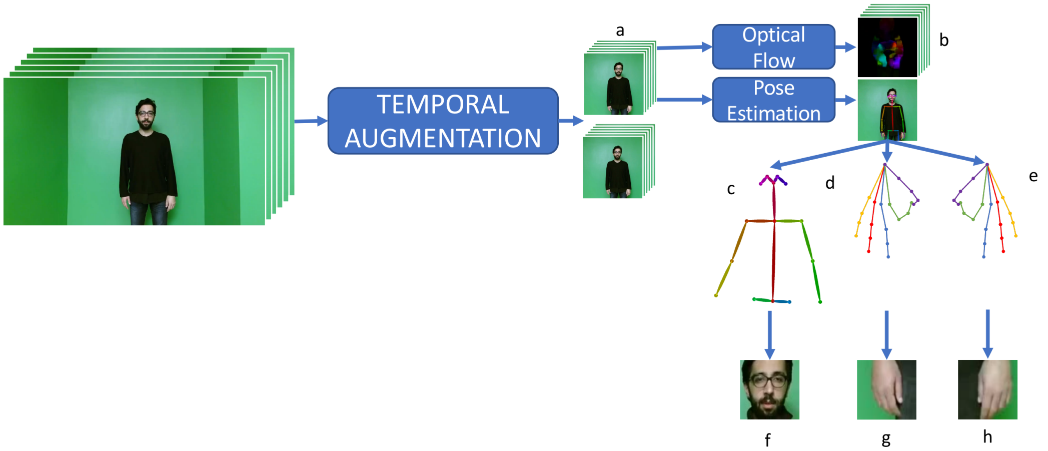

Figure 2. Data augmentation and preprocessing outputs: a) cropped body RGB videos, b) optical flow RGB video, c) body pose data, d) right hand pose data, e) left hand pose data, f) cropped face video, g) cropped right hand video, h) cropped left hand video

Primarily, for data preparation, using the videos resulted from data augmentation, we extracted 2D pose data of body and hands using Openpose [36] pose estimation library. Openpose is able to make 2D or 3D pose estimations for body, feet, hands, and face with the help of convolutional layers. Employing a top-down approach, it is able to extract pose data, primarily, for body; then for hands and face making use of the points found for body. Openpose is able to estimate 25 joint points for body, 70 for face, and 21 for each hand [36]. In this study, of all the extracted points, we used 15 upper-body joints and all hand joints. Each joint point was normalized depending on width and height of the video in order not to be affected by the video size. To evaluate pose estimation performance of Openpose we used visual inspection. We projected the body pose estimations onto RGB videos. When we watched the projected videos, we did not observe significant inconsistencies in the estimated joint locations. Therefore, we found the pose estimation results useful for our purpose.

Optical flow is another method that researchers use in sign language and action recognition studies[6, 8, 16, 35]. With optical flow, temporal information in videos can be modeled using another data modality. In order to analyze the motions better and to emphasize the importance of temporal information in sign language, we created optical flow videos. We used Farneback [37] algorithm for the dense optical flow extraction. Optical flow outputs were processed and turned into RGB videos in such a way that they were colored according to the motion direction and size.

As hand shapes and facial expressions are important in sign languages, we created 3 other video streams so that face and each hand could be analyzed within their individual streams. For creating those videos, we used the joint points obtained from Openpose. Pose estimation results of Openpose were very accurate for body and face. In order to create the face videos, a square area around nose tip joint point was cropped from the body RGB video frames. As for hands, the self occlusion of fingers caused less accurate estimation for hand joints. Less points were found and the found joints were not very accurate because of it. Therefore, to estimate hand locations, we tried a few strategies such as taking the most accurate hand joint or the joints in the center of hand. Then we opted for calculating the mean of all found joint points since this presented a better estimation for hand location. Taking the calculated point as the center of the hand, we cropped a square area from the RGB frame. As a result of the preprocessing; videos of body, face, and hands were created as RGB data streams. Each video consists of 32 frames. Body videos were resized to 224 x 224 pixels, face videos to 80 x 80 pixels, and hands videos to 60 x 60 pixels.

As the result of the data augmentation and preprocessing, 8 streams in different data types were obtained (4 RGB, 1 optical flow, and 3 pose data). They were RGB videos of body, face, left hand, and right hand; an optical flow video; joint points of body, left hand, and right hand. The streams are shown in Figure 3.

Upon data augmentation and preprocessing, the number of videos to be used as the training data was almost doubled. No data augmentation was applied to the testing dataset. After data preprocessing and augmentation, there were 9081 samples to be used for training and 948 samples to be used for testing.

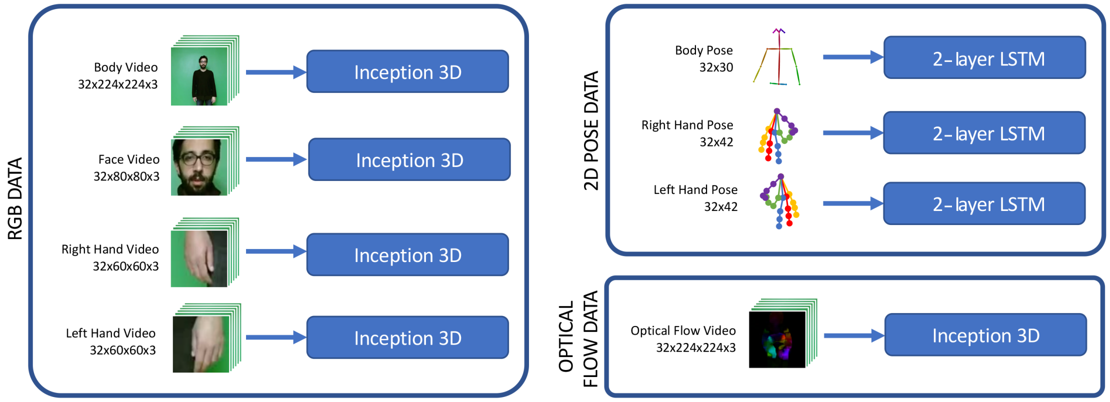

Figure 3. Streams created after data preprocessing.

3.3. Feature extraction

After the data preprocessing, there were two types of data modalities. They were the videos coded in RGB and the sequential pose data extracted from each video frame. Each data stream was trained through a deep learning model separately. Inception3D (I3D) model [35] was used to extract RGB and optical flow features; and LSTM-RNN was used to learn from skeletal data.

2D convolution is enough for extracting feature maps from 2D data like images that only have spatial information. Videos have spatiotemporal information, and temporal information is as important as spatial information in many applications such as action recognition and sign language recognition. It is not possible to analyze the differences between video frames with 2D convolution. Therefore, researchers proposed models that make 3 dimensional convolution [38]. Thus, I3D model, which can effectively learn the spatiotemporal information in videos, was used in this study.

I3D model is the inflated form of Inception v1 image classification model in order that it can process temporal data, as well. 2 dimensional convolutions and pooling filters of the original model were turned into 3 dimensional ones through the inflation. I3D has often been used in recent action recognition and sign language recognition studies for benchmarking [22, 23]. The details of I3D model are given in Figure 4. Inception units in the model do multiple convolutions with various filter sizes and then the convolution results are transferred to the next layer after concatenation.

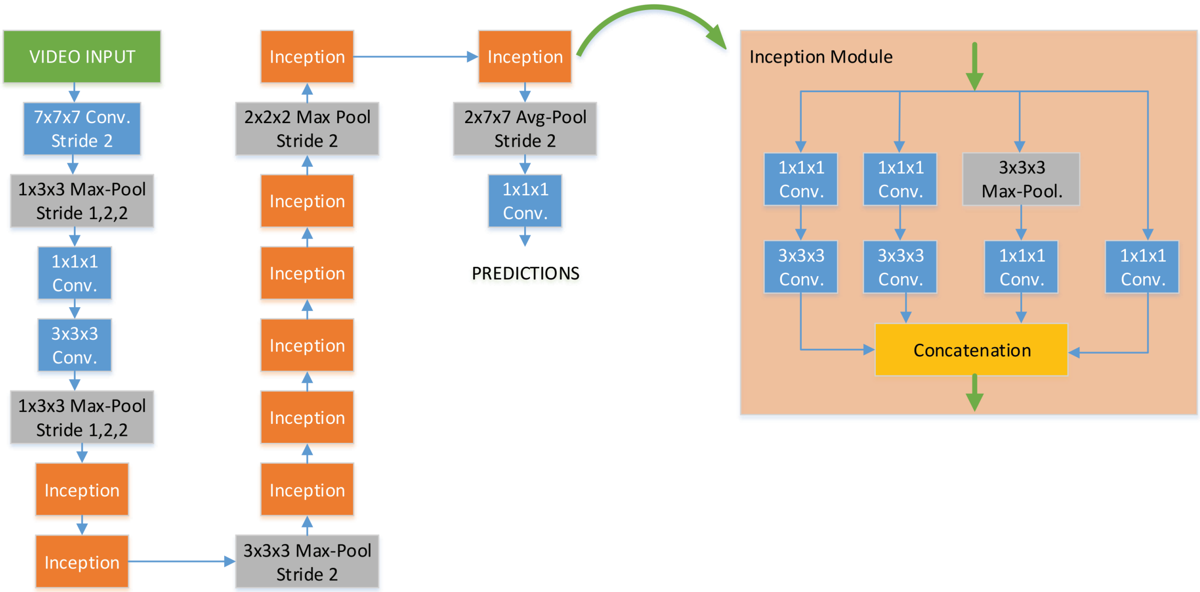

Figure 4. Overview of Inception3D architecture.

The pose data used in the model were made up of body and hand joints' location in the 2D space, which were extracted from each frame of the video samples. We normalized joint coordinates according to the video width and height in order not to be affected by the video resolution. To train this data, we used LSTM-RNNs, which is the improved version of RNNs that can work with sequential data effectively.

An RNN is a special type of neural network that uses previous inputs to calculate the output. For a sequential data of length T, such as $X = (x_1, ..., x_T)$, the hidden states $H = (h_1, ..., h_T)$ of the RNN can be calculated iteratively using Equation 1, where θ represents a nonlinear activation function and β represents the RNN parameter.

$$h_t = \theta(x_t, h_{t-1}; \beta) \tag{1}$$

For long sequences, RNNs have problems like exploding or vanishing gradient. With memory units, LSTM solves the problem with RNNs by forgetting or remembering previous network status, and updating the current state. An LSTM unit contains a self-connected memory cell (c), an input gate (i), an output gate (o), and a forget gate (f). For a given input, the below calculations are done in LSTM.

$$i_t = \theta_i(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1}), \tag{2}$$

$$f_t = \theta_f(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1}), \tag{3}$$

$$c_t = f_tc_{t-1} + i_t\theta_c(W_{xc}x_t + W_{hc}h_{t-1}), \tag{4}$$

$$o_t = \theta_o(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_t), \tag{5}$$

$$h_t = o_t\theta_t(c_t), \tag{6}$$

In the equations 2–6, W represents the weights between two LTSM units, θ represents activation functions, i_t, f_t, c_t, o_t, and h_t represent the outputs of input gate, forget gate, cell, output gate, and LSTM unit at time t, respectively.

In this study, we assessed the impact of each stream on sign language recognition; and then we proposed a holistic SLR system utilizing the most suitable streams. For stream fusion, we used late fusion technique in this study. Late fusion means that each stream's feature extraction is done in their own streams, and they are combined with other streams at a point close to the output. After fusion, combined features are transferred to a fully connected layer, and sign recognition is accomplished. The multistream data fusion model proposed in this study is given in Figure 5.

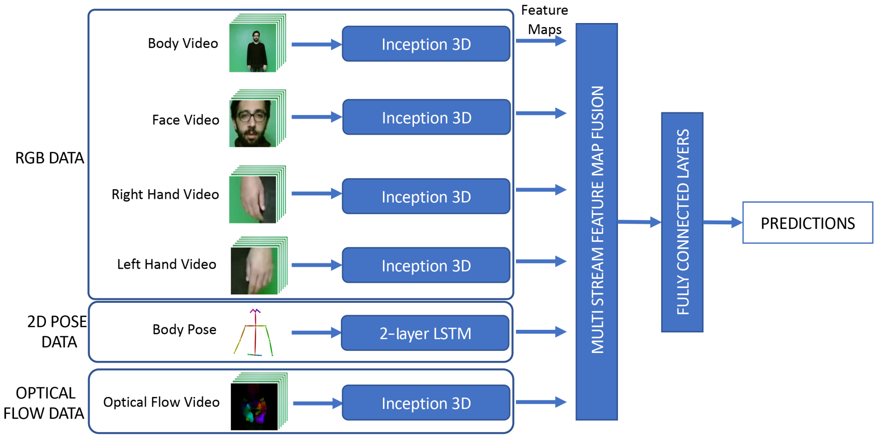

Figure 5. Proposed multistream data fusion model.

4. Experiments and results

4.1. Experimental setup

For the implementation of the proposed SLR system, we used Keras TensorFlow library running on Ubuntu operating system. In the experiments, a PC with AMD 6 Core Ryzen 5 2600 processor, 32 GB RAM and 12 GB NVIDIA Titan X GPU was used.

We carried out the experiments in 3 stages. In the first stage, each data type was trained with the help of deep learning architectures according to the data type of streams given in Figure 3. In the second stage, the outputs of the trained models were concatenated in various combinations, and then given to a fully connected network as the input for recognition. Second-stage experiments were aimed to detect the data combination proving the highest accuracy. In the third stage, the streams which gave the highest accuracy were trained together, and a holistic SLR system was developed by training a single model. Recognition accuracy was used for evaluating the performance of the SLR system.

In the first experiments, I3D models were trained for a period of 10 epochs with Adam optimizer and 10^{-3} learning rate; then the learning rate was decreased to 1/10 of its value, and the training was carried out for 10 more epochs. The two LSTM layers we used for pose data had 256 and 128 LSTM units with dropout rates of 0.5 and 0.3, respectively. We started the training with random weights and trained them for 20 epochs.

4.2. Experimental results

4.2.1. Experiments on each stream separately

Each of the communication channels used in sign languages had varying significance. It is possible to observe this in the results of the conducted experiments. Testing results, which were obtained upon individual training of each stream, are given in Table 1.

Table 1. Testing results for each stream separately.

Data type	Streams	Precision (%)	Recall (%)	F1-Score (%)	Accuracy (%)
RGB	Body video	80	83	79	79.6
	Face	28	33	27	29.1
	Left hand	27	28	24	27.6
	Right hand	46	51	44	45.4
Pose	Body	49	52	47	50.6
	Left hand	9	10	9	9.6
	Right hand	24	28	23	23.8
Optical flow	Body flow	79	81	78	78.5

According to the test results, it is seen that the highest accuracy was achieved through the body RGB video. While face and left hand presented similar accuracy rates, right hand presented 45% accuracy alone. Pose data presented an accuracy half as much as the one presented by the video data in the same channel.

Many SLR studies that use hands attach primary importance to right hand as it is the dominant hand in most cases, and they only take the right hand into account for SLR. However, taking our results into consideration, although all of our signers are right-handed, the left hand also contains data which might contribute to sign recognition in a supplementary role. A similar case is also valid for the pose data. Sign recognition accuracy was found as 50.6% when the body pose data was used alone. It suggests that body pose data alone is not enough for sign recognition; however, it can improve the recognition performance in combination with other channels.

4.2.2. Experiments on stream combinations

Having determined each stream's accuracy alone, we fused streams in various combinations. We chose RGB body video model as the baseline model and aimed to increase the sign recognition accuracy by fusing streams. While fusing streams, instead of combining streams and training them every time, we used bottleneck features that were extracted from the models we trained in the first set of the experiments. The extracted bottleneck features were concatenated and fed to a neural network with 2 fully connected layers. Test results of the fused streams in different combinations are given in Table 2.

Table 2. Results of fused streams in different combinations.

Streams (number of streams)	Precision (%)	Recall (%)	F1-Score (%)	Accuracy (%)
Body RGB (1) - baseline model	80	83	79	79.6
Body RGB + pose (2)	79	80	77	79.0
Body RGB + optical flow (2)	85	87	84	84.9
Body RGB + face RGB (2)	81	84	80	81.1
Body RGB + hands RGB (3)	79	83	78	78.5
Body RGB + all pose streams (4)	72	75	70	72.8
Body RGB + face + hands (4)	81	86	81	81.2
Body RGB + pose + optical flow (3)	86	88	85	85.9
Only pose streams (3)	41	44	40	42.1
All streams (8)	86	89	86	85.9
All streams - hand pose streams (6)	88	91	88	88.3

Of all the streams, optical flow stream provided the most remarkable positive effect. When the optical flow data were combined with the body videos, the accuracy increased by 4.7% relative to the baseline.

When 4 RGB video streams were fused together (body, hands, and face), the accuracy increased by 2% relative to the baseline model; when all streams were fused, an accuracy of 85.9% was reached. The experiments showed that hands' pose streams decreased the accuracy of the stream they were fused with. It can be accounted for by the fact that the skeletal data obtained by Openpose cannot detect many joints on fingers due to the occlusion. For this reason, we removed hands' pose streams from the architecture in which all the streams were fused. This change boosted the accuracy by 8.7% relative to the baseline model, which raised the overall performance to 88.3%.

4.2.3. Proposed model using the best stream combination

After all these results, we developed a model to simultaneously process the streams that gave the highest accuracy and to combine them with a fusion mechanism. The details of this model are given in Figure 5. The proposed model used the optical flow and body pose data together with RGB videos of body, face, and hands in a total of 6 streams. The feature maps were created from each stream, and then a late fusion was performed. After the fusion, feature maps were transferred to the fully connected neural network. Instead of initializing with random weights, we used the weights we got from the previous trainings for the training of the new model with the fusion mechanism. After training the new model with fusion mechanism for 15 epochs, the test result revealed an accuracy of 89.35%. The final confusion matrix of the results is given in Figure 6.

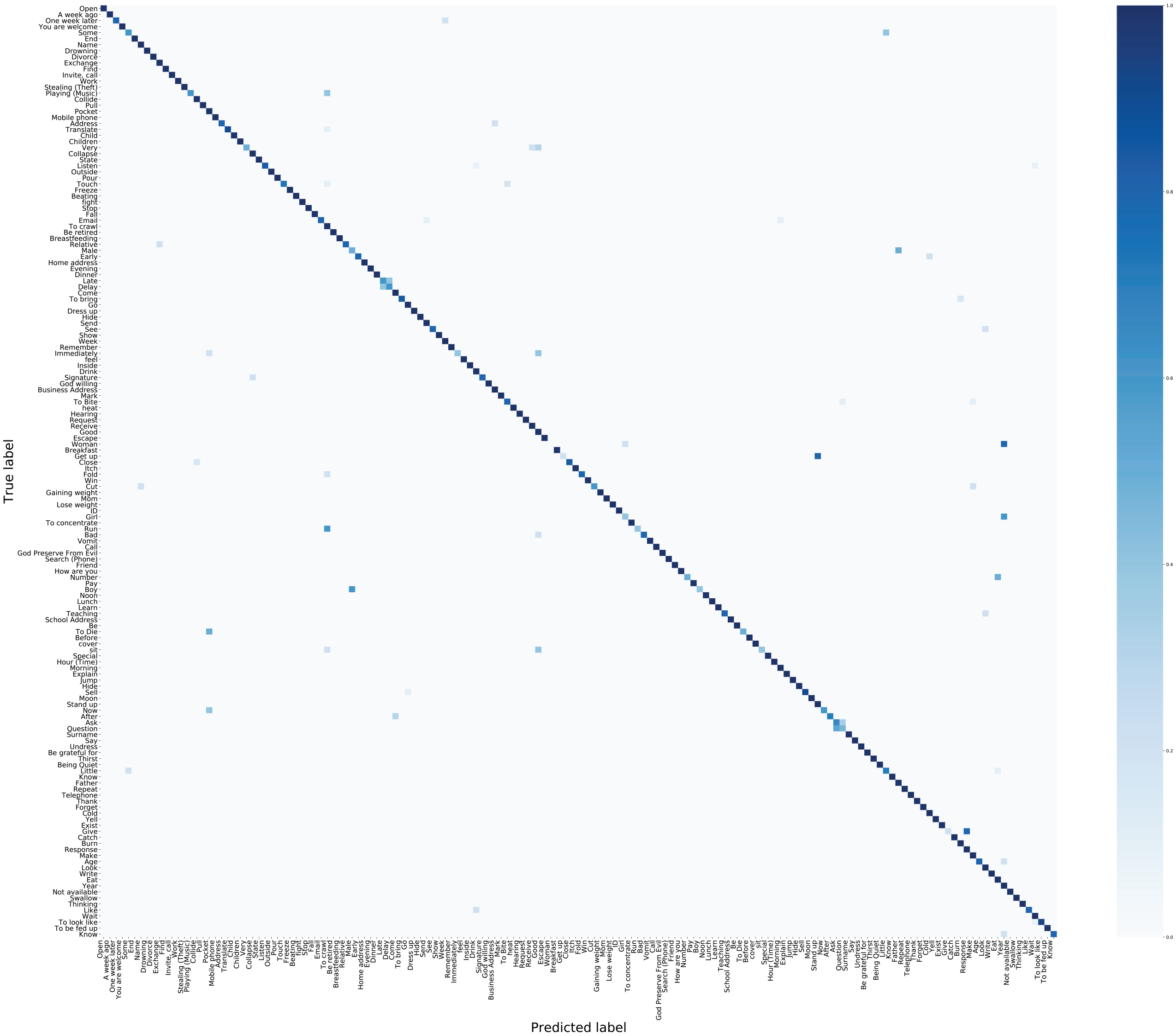

Figure 6. Confusion matrix of the proposed multistream model.

When the sign recognition results are examined in detail, we saw that the most common confused signs were those derived from the same word stems in Turkish spoken language. To ask*(sormak)* / question*(soru)*; late*(geç)* / to be late*(geç kalmak)*; and little*(az)* /some*(biraz)* are some examples of it. Another reason for confusion was that some signs had more than one meaning. For example, the sign for male*(erkek)* could also mean father*(baba)* or son*(oğlan)* depending on the context. Since the same sign is located in different classes in the dataset, this caused confusion in the model.

In some of the confused signs, there were some subtle differences among fingers. For example, the motions for crawling*(emeklemek)* and running*(koşmak)* signs are similar to each other. However, while performing the signs, hands have different shapes. It was observed that the proposed SLR system could not detect this difference clearly. A similar case existed among to give*(vermek)* / response*(cevap)*; and year*(yıl)* / woman*(kadın)* / girl*(kız)*, too. Sample sequences of those signs are given in Figure 7.

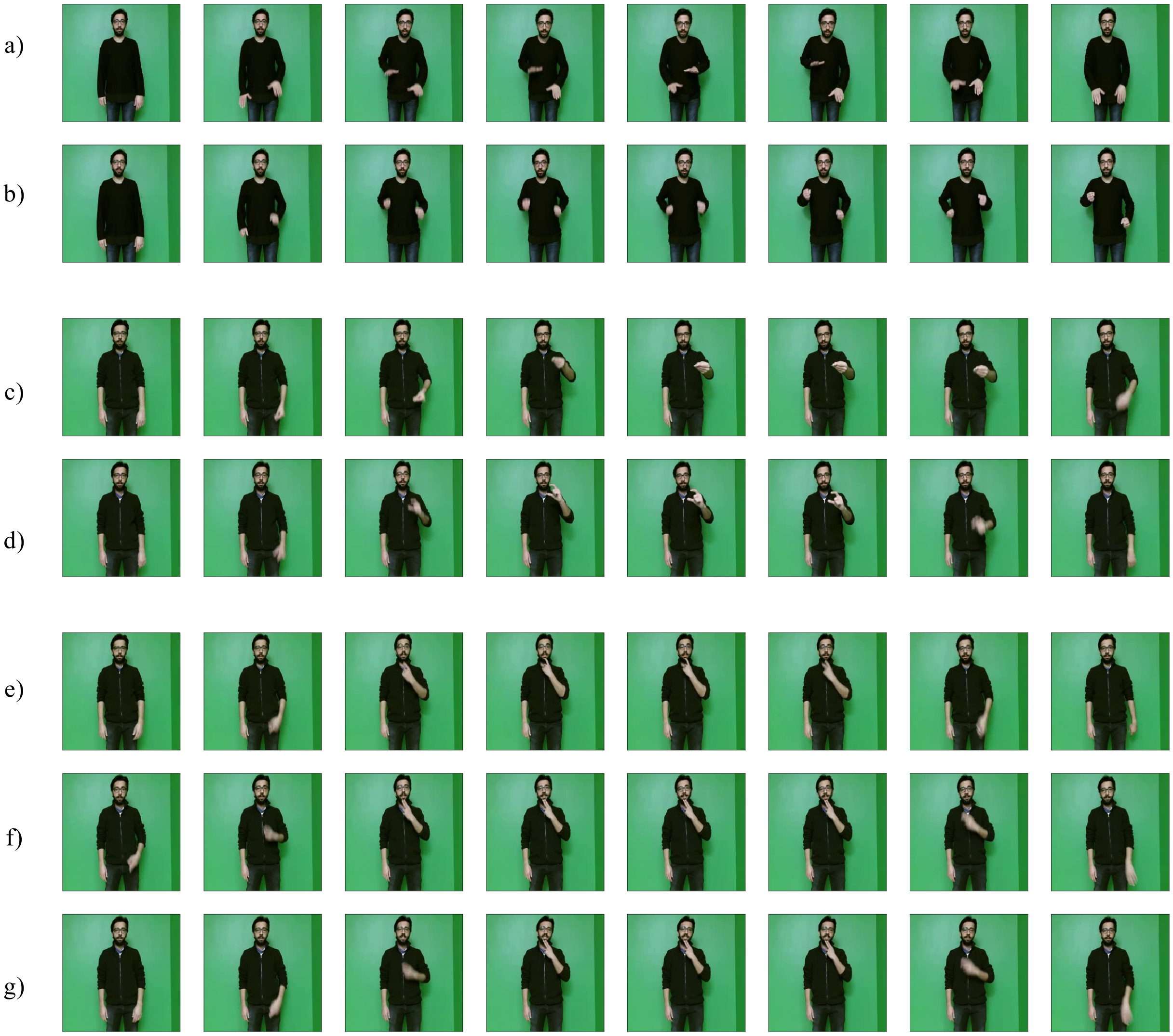

Figure 7. Sequences of confused signs. Crawl (a) and run (b), give (c) and response (d), year (e), woman (f), and girl (g) have similar signs.

4.2.4. Comparison with other studies

There is not an open benchmark SLR dataset developed for Turkish. Thus, most studies are usually undertaken based on small datasets created by researchers. The BosphorusSign dataset we used in this study is the most comprehensive dataset in Turkish sign language. The studies on this dataset is limited to the ones conducted by its developers [39]. The state-of-the-art accuracy on ”General” subset of this dataset is 81.58% [32]. Our proposed approach, multistream data fusion-based sign language recognition, has increased this accuracy by

7.7%. For comparison purposes, we also preprocessed the LSA64 dataset [40] and trained our model on it. This dataset is created in Argentinian sign language and has 64 signs. The state-of-the-art accuracy on this dataset is 99.91% [2]. We achieved 99.90% accuracy with our multistream data fusion based sign language recognition method using signer-dependent testing scheme proposed by the original study. When we used a signer-independent testing scheme like we used on BosphorusSign, we achieved a 99.49% accuracy. Comparative results are presented in Table 3.

Table 3. Comparative results on BosphorusSign and LSA64 datasets.

Dataset	Method	Data modalities	Accuracy (%)
BosphorusSign	I3D	RGB Only	79.6
	TAF[32]	RGB Only	81.58
	Ours	RGB, optical flow & pose	89.35
LSA64	[29]	RGB & pose	98.09
	[2]	RGB & pose	99.91
	Ours	RGB, optical flow & pose	99.9

5. Conclusion

In this study, a multistream data fusion-based sign language recognition system was proposed. Each stream corresponds to communications channels used in Turkish sign language and each communication channel has various levels of importance for understanding a sign. Moving beyond the traditional mono stream paradigm, we focused on combining the multimodal data obtained from RGB videos. To this aim, we created a total of 8 streams—4 RGB, 3 pose, and 1 optical flow—from RGB videos. Inception 3D was used to learn RGB and optical flow data; and LSTM-RNN was used to learn pose streams. Firstly, each stream was trained and tested separately. Having found the testing accuracy of each stream, we saw that none of the individual streams could prove a very high sign recognition performance. We hypothesized that using different streams together could improve sign language recognition performance. Therefore, we conducted experiments by fusing the streams in various combinations. We aimed to determine the stream combinations which could provide the highest sign recognition accuracy in these experiments. The highest sign recognition accuracy was obtained from 6 streams, which were 4 RGB streams (i.e. processing the videos of hands, face, and whole body), pose stream, and optical flow. These 6 streams were late fused, and the feature maps of the streams were concatenated. The output of the fusion process was given to a neural network with 2 fully connected layers as the input. Sign language recognition results were gathered from fully connected neural network output.

Our experiments showed that our proposed multistream data fusion-based sign language recognition approach provided better results compared to the other studies on the BosphorusSign dataset. However, since each sign language is unique in terms of the signs and linguistic patterns, the significance of the channels might differ in other sign languages. Therefore, the findings of this study about the significance of communication channels are limited to BosphorusSign dataset and Turkish sign language. The significance of the channels in other languages can be evaluated through conducting similar studies for other sign languages.

Acknowledgment

We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan X Pascal GPU used for this research.

References

[1] Kumar P, Gauba H, Roy PP, Dogra DP. A multimodal framework for sensor based sign language recognition. Neurocomputing. 2017; 259: 21-38. doi: 10.1016/j.neucom.2016.08.132

[2] Escobedo E, Ramirez L, Camara G. Dynamic sign language recognition based on convolutional neural networks and texture maps. In: 2019 32nd SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI); 2019. doi: 10.1109/sibgrapi.2019.00043

[3] Zeshan U. Aspects of Türk isaret dili (Turkish sign language). Sign Language and Linguistics 2003; 6 (1): 43-75. doi: 10.1075/sll.6.1.04zes

[4] Liu T, Zhou W, Li H. Sign language recognition with long short-term memory. In: 2016 IEEE International Conference on Image Processing (ICIP); 2016. pp. 2871-2875. doi: 10.1109/icip.2016.7532884

[5] Cheok MJ, Zaid O, Hisham JM. A review of hand gesture and sign language recognition techniques. International Journal of Machine Learning and Cybernetics 2019; 10 (1): 131-153. doi: 10.1007/s13042-017-0705-5

[6] Abavisani M, Joze HRV, Patel VM. Improving the performance of unimodal dynamic hand-gesture recognition with multimodal training. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2019: 1165-1174.

[7] Cooper H, Holt B, Bowden R. Sign language recognition. In: Moeslund TB, Hilton A, Krüger V, Sigal L (Editor). Visual Analysis of Humans. New York, NY, USA: Springer, 2011: 539-562.

[8] Lim KM, Tan AW, Tan SC. Block-based histogram of optical flow for isolated sign language recognition. Journal of Visual Communication and Image Representation. 2016; 40: 538-545. doi: 10.1016/j.jvcir.2016.07.020

[9] Chuan CH, Regina E, Guardino C. American sign language recognition using leap motion sensor. In: 2014 IEEE 13th International Conference on Machine Learning and Applications; 2014. pp. 541-544.

[10] Huang CL, Huang WY. Sign language recognition using model-based tracking and a 3D Hopfield neural network. Machine Vision and Applications 1998; 10 (5-6): 292-307. doi: 10.1007/s001380050080

[11] Grobel K, Assan M. Isolated sign language recognition using hidden Markov models. In: 1997 IEEE International Conference on Systems, Man, and Cybernetics; Computational Cybernetics and Simulation; 1997. pp. 162-167. doi: 10.1109/icsmc.1997.625742

[12] Büyüksaraç B. Sign language recognition by image analysis. PhD, Middle East Technical University, Ankara, Turkey, 2015.

[13] Haberdar H. Real time Turkish sign language recognition system from video using hidden Markov models. Masters, Yıldız Technical University, İstanbul, Turkey, 2005.

[14] Starner T, Weaver J, Pentland A. Real-time American sign language recognition using desk and wearable computer based video. IEEE Transactions on Pattern Analysis and Machine Intelligence 1998; 20(12): 1371-1375. doi: 10.1109/34.735811

[15] Vogler C, Metaxas D. Parallel hidden markov models for American sign language recognition. In: Proceedings of the Seventh IEEE International Conference on Computer Vision; 1999. pp. 116-122. doi: 10.1109/iccv.1999.791206

[16] Zhang LG, Chen Y, Fang G, Chen X, Gao W. A vision-based sign language recognition system using tied-mixture density HMM. In: Proceedings of the 6th International Conference on Multimodal Interfaces - ICMI 04; 2004: 198-204. doi: 10.1145/1027933.1027967

[17] Deng L. A tutorial survey of architectures, algorithms, and applications for deep learning. APSIPA Transactions on Signal and Information Processing 2014; 3. doi: 10.1017/atsip.2013.9

[18] Koller O, Ney H, Bowden R. Deep learning of mouth shapes for sign language. In: 2015 IEEE International Conference on Computer Vision Workshop (ICCVW); 2015. pp. 85-91. doi: 10.1109/iccvw.2015.69

[19] Demircioğlu B, Bülbül G, Köse H. Turkish sign language recognition with Leap Motion. In: 2016 24th Signal Processing and Communication Application Conference (SIU); 2016: 589-592. doi: 10.1109/siu.2016.7495809

[20] Huang J, Zhou W, Li H, Li W. Sign language recognition using 3D convolutional neural networks. In: 2015 IEEE International Conference on Multimedia and Expo (ICME); 2015: 1-6. doi: 10.1109/icme.2015.7177428

[21] Pigou L, Dieleman S, Kindermans PJ, Schrauwen B. Sign language recognition using convolutional neural networks. In: European Conference on Computer Vision; 2014: 572-578.

[22] Vaezi Joze H, Koller O. MS-ASL: A large-scale data set and benchmark for understanding american sign language. In: The British Machine Vision Conference (BMVC); 2019.

[23] Li D, Opazo CR, Yu X, Li H. Word-level deep sign language recognition from video: A new large-scale dataset and methods comparison. In: 2020 IEEE Winter Conference on Applications of Computer Vision (WACV); 2020: 1459-1469. doi: 10.1109/wacv45572.2020.9093512

[24] Ferreira PM, Cardoso JS, Rebelo A. On the role of multimodal learning in the recognition of sign language. Multimedia Tools and Applications 2019; 78 (8): 10035-10056. doi: 10.1007/s11042-018-6565-5

[25] Zhang S, Meng W, Li H, Cui X. Multimodal spatiotemporal networks for sign language recognition. IEEE Access 2019; 7: 180270-180280. doi: 10.1109/access.2019.2959206

[26] Camgoz NC, Hadfield S, Koller O, Bowden R. SubUNets: End-to-end hand shape and continuous sign language recognition. In: 2017 IEEE International Conference on Computer Vision (ICCV); 2017. pp. 3056-3065. doi: 10.1109/iccv.2017.332

[27] Charles J, Pfister T, Everingham M, Zisserman A. Automatic and efficient human pose estimation for sign language videos. International Journal of Computer Vision. 2014; 110 (1): 70-90. doi: 10.1007/s11263-013-0672-6

[28] Gattupalli S, Ghaderi A, Athitsos V. Evaluation of deep learning based pose estimation for sign language recognition. In: Proceedings of the 9th ACM International Conference on PErvasive Technologies Related to Assistive Environments - PETRA 16; 2016. doi: 10.1145/2910674.2910716

[29] Konstantinidis D, Dimitropoulos K, Daras P. Sign language recognition based on hand and body skeletal data. In: 3DTV-Conference: The True Vision - Capture, Transmission and Display of 3D Video (3DTV-CON); 2018. doi: 10.1109/3dtv.2018.8478467

[30] Unutmaz B, Karaca AC, Güllü MK. Turkish sign language recognition using Kinect skeleton and convolutional neural network. In: 27th Signal Processing and Communications Applications Conference (SIU); 2019. doi: 10.1109/siu.2019.8806380

[31] Özdemir O, Kindiroglu AA, Akarun L. Isolated sign language recognition with fast hand descriptors. In: 2018 26th Signal Processing and Communications Applications Conference (SIU). 2018: 1-4.

[32] Kindiroglu AA, Ozdemir O, Akarun L. Temporal accumulative features for sign language recognition. In: IEEE/CVF International Conference on Computer Vision Workshop (ICCVW); 2019. doi: 10.1109/iccvw.2019.00164

[33] Tamer NC, Ozdemir O, Saraclar M, Akarun L. Dynamic time warping based sign retrieval. In: 27th Signal Processing and Communications Applications Conference (SIU); 2019. doi: 10.1109/siu.2019.8806601

[34] Camgöz NC, Kındıroğlu AA, Karabüklü S, Kelepir M, Özsoy AS et al. BosphorusSign: a Turkish sign language recognition corpus in health and finance domains. In: Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16); 2016: 1383-1388.

[35] Carreira J, Zisserman A. Quo Vadis, action recognition? A new model and the Kinetics dataset. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2017: 6299-6308. doi: 10.1109/cvpr.2017.502

[36] Cao Z, Martinez GH, Simon T, Wei SE, Sheikh YA. OpenPose: Realtime multi-person 2D pose estimation using part affinity fields. Transactions on Pattern Analysis and Machine Intelligence 2019. doi: 10.1109/tpami.2019.2929257

[37] Farnebäck G. Two-frame motion estimation based on polynomial expansion. Image Analysis Lecture Notes in Computer Science 2003: 363-370.

[38] Tran D, Bourdev L, Fergus R, Torresani L, Paluri M. Learning spatiotemporal features with 3D convolutional networks. In: IEEE International Conference on Computer Vision (ICCV); 2015. pp. 4489-4497. doi: 10.1109/iccv.2015.510

[39] Kosmopoulos D, Oikonomidis I, Constantinopoulos C, Arvanitis N, Antzakas K et al. Towards a visual Sign Language dataset for home care services. In: 2020 15th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2020); 2020: 622-626.

[40] Ronchetti F, Quiroga F, Estrebou CA, Lanzarini LC, Rosete A. LSA64: an Argentinian sign language dataset. In: XXII Congreso Argentino de Ciencias de la Computación (CACIC 2016). 2016.