



# Detection of Turkish Sign Language Using Deep Learning and Image Processing Methods

Bekir Aksoy, Osamah Khaled Musleh Salman & Özge Ekrem

To cite this article: Bekir Aksoy, Osamah Khaled Musleh Salman & Özge Ekrem (2021) Detection of Turkish Sign Language Using Deep Learning and Image Processing Methods, Applied Artificial Intelligence, 35:12, 952-981, DOI: [10.1080/08839514.2021.1982184](https://doi.org/10.1080/08839514.2021.1982184)

To link to this article: <https://doi.org/10.1080/08839514.2021.1982184>



Published online: 23 Sep 2021.



Submit your article to this journal [↗](#)



Article views: 1994



View related articles [↗](#)



View Crossmark data [↗](#)



Citing articles: 4 View citing articles [↗](#)



# Detection of Turkish Sign Language Using Deep Learning and Image Processing Methods

Bekir Aksoy , Osamah Khaled Musleh Salman , and Özge Ekrem 

Mechatronics Engineering Department, Isparta University of Applied Sciences, Isparta University of Applied Sciences Technology Faculty, Isparta, Turkey

## ABSTRACT

Sign language is a physical language that enables people with disabilities to communicate using hand and facial gestures. For this reason, it is very important for people with disabilities to express themselves freely in society and to make the sign language understandable to everyone. In this study, the data set was created by taking 10223 images for 29 letters in the Turkish Sign Language Alphabet. Images are made suitable for education by using image enhancement techniques. In the final stage of the study, classification processes on images were carried out by using CapsNet, AlexNet and ResNet-50, DenseNet, VGG16, Xception, InceptionV3, NasNet, EfficientNet, Hitnet, Squeezenet architectures and TSLNet, which was designed for the study. When the deep learning models were examined, it was found that CapsNet and TSLNet models were the most successful models with 99.7% and 99.6% accuracy rates, respectively.

## ARTICLE HISTORY

Received 25 February 2021  
Revised 13 September 2021  
Accepted 14 September 2021

## Introduction

Communication is a skill necessary for people who live in the community to express themselves and live in harmony (Turan 2014). While the people who make up the community can communicate using a verbal and auditory language, individuals who lack these skills can express themselves using sign language, which is a visual language (Aly, Aly, and Almotairi 2019). Sign language is a communication tool in which people with hearing disabilities or hearing difficulties use body movements such as hands, arms, and gestures to communicate (Aran et al. 2007). Since the sign language used by people with hearing disabilities to communicate and express themselves is not known by most people in our society, it is seen that these people have difficulty in expressing themselves in their daily lives (Pigou et al. 2014).

Artificial intelligence methods, which emerged with the rapid development of technology, are used in many areas as well as in sign language. Artificial intelligence is a method that aims to model human thinking structure and decision-making ability (Özkan and Ülke 2017). Artificial intelligence consists of machine learning and deep learning methods. Machine learning is one of

the frequently used branches of artificial intelligence and deep learning is a sub-branch of machine learning. Deep learning consists of supervised or unsupervised learning algorithms used to perform operations such as feature extraction and conversion by accepting the output of the previous layers as input to the current layer (Deng and Yu. 2014; Şeker, Diri, and Balık 2017). In deep learning, there is a structure, based on the learning of multiple feature levels or representations of data. High-level features are derived from low-level features, creating a hierarchical representation. This representation learns multiple levels of representation that correspond to different levels of abstraction (Bengio 2009). Deep learning is used in many areas such as image analysis, sound analysis, robot technology, remote sensing, genetic analysis, cancer and disease diagnosis (Toğaçar and Ergen 2019).

When the academic literature is examined; In their study, Oyedot and Kashman made use of deep learning methods to recognize 24 different static hand movements belonging to American Sign Language. The convolutional neural networks (CNN) and stacked denoising auto encoders (SDAE) used in the study were trained on an open access database. With the training of SDAEs at different depths, improvements in performance can be observed as the number of layers' increases (Oyedotun and Khashman 2017). In the Haitham and Kareem's study, they proposed a vision-based recognition algorithm to recognize 6 static hand movements (Open, Close, Cut, Paste, Enlarge, Shrink) used for HCI (Human-computer interaction). The proposed algorithm consists of three stages: preprocessing, feature extraction and classification (Hasan and Abdul-Kareem 2014). Devineau et al. In their work, they developed a multi-channel CNN with two feature extraction modules and one residual branch per channel, that enables the recognition and classification of 3-dimensional hand movements using only hand-skeletal joint positions. The created model uses hand-skeleton (3D) data set, which is processed faster in recognizing hand movements, rather than using RGB-D image sequences for recognition of existing deep learning methods (Devineau et al. 2018). Kapuściński and Warchoń presented a method for the recognition of static hand postures based on hand-skeletal data obtained by using the Leap Motion sensor. In the presented method, a new descriptor has been proposed to encode information regarding the distances between hand points corresponding to the fingertips and the center of the palm (Kapuściński and Warchoń 2020). Chuan et al. presented the American Sign Language recognition system using a 3D motion sensor. They classified 26 letters of the English alphabet belonging to the American Sign language by using the k-nearest neighbor method and vector machine by using the Leap Motion sensor and the features consisting of sensory data (Chuan, Regina, and Guardino 2014).

In the study, one of the biggest problems for hearing impaired individuals in Turkey is that the use of Turkish sign language is not known by individuals who do not have a disability in the society. An exemplary application has been

developed with the CNN-based Turkish sign language (TSL) deep learning architecture, which is original designed to recognize the letters used in Turkish sign language. Thus, it is aimed to establish easier communication between hearing impaired individuals and individuals who do not know how to use sign language. In this study, images were taken in different positions for sign language corresponding to 29 Turkish letters. Image processing techniques such as filtering and segmentation have been applied to the images to increase the accuracy of the system. TSL alphabet was classified using 12 different deep learning architectures. Among the methods used, the test accuracy value of the TSLNet architecture, which was designed for the study, was obtained as 99.6%.

## Material and Method

The structure of the deep learning algorithms and the images used in the study are expressed in the material section. The method part consists of application information about image processing and deep learning algorithms used in the study.

### Material

In the study, 10223 images of the sign language equivalents of the 29 letters of the Turkish alphabet are collected using the camera specified in [Table 1](#) and these images are tagged manually. Example images of the letters A and B used in Turkish sign language are given in [Figure 1](#).

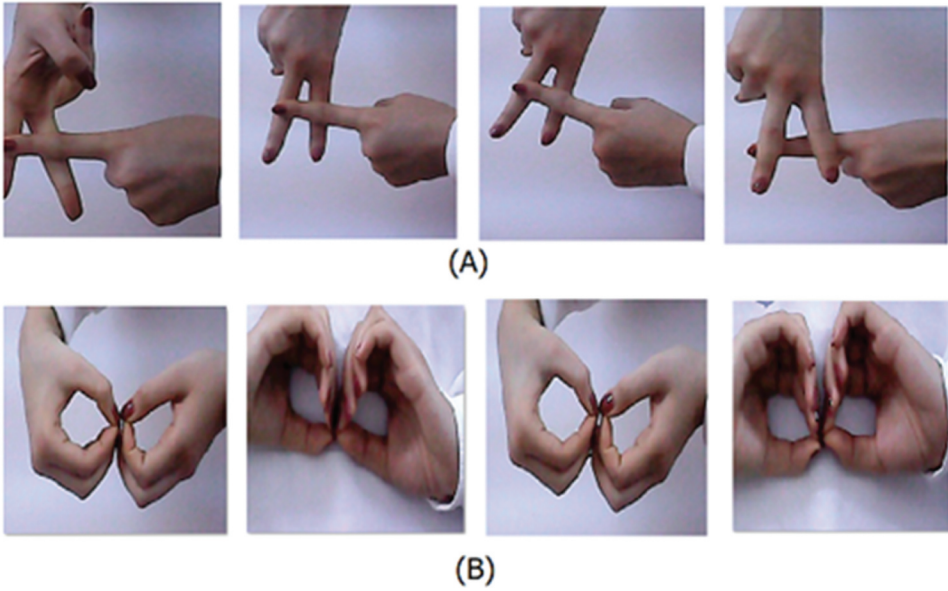
Information about the data set used in the study given in [Table 2](#).

When [Table 2](#) is examined; In the study, 10223 images with 29 classes were collected from 5 different individuals, three different backgrounds and varying distances from 0 to 100 cm from the camera. Images with three different backgrounds were used: plain white, in-room conditions (night) and in-room conditions (daytime). Some problems were encountered while implementing the study. The first problem encountered while implementing the study is that some of the images are mislabeled due to the large number of images in the data set used. To solve this problem, incorrectly labeled images were corrected by considering images that negatively affect accuracy during the training phase. Another problem encountered in the study is that it becomes difficult

**Table 1.** General features of the camera and notebook used in the study.

Camera features		Notebook features	
<b>Sensor Type</b>	CMOS	<b>CPU</b>	Intel(R) Core(TM) i7-7700HQ CPU @ 2.80 GHz 2.80 GHz
<b>Resolution (HxV)</b>	2592 px x 1944	<b>RAM</b>	16.0 GB DDR4
<b>Resolution</b>	5 MP	<b>GPU</b>	2.0 GB GTX 1050
<b>Frame Rate</b>	14 fps		
<b>Colored/Black-White</b>	Colored		
<b>Data Bus</b>	USB 3.0		





**Figure 1.** Representation of the letter A and B in Turkish sign language.

**Table 2.** Technical information about the data set used in this study.

Number of individuals	5
Number of classes	29
Background	3
Distance	Up to 100 cm
Data size	10223

to detect hand and finger images in the image in-room environment conditions. To solve this problem, the performance of deep learning models was increased by using segmentation methods and thus, images taken in-room environment conditions were detected more successfully.

### *Image Processing Techniques*

In the study, image enhancement operations were carried out using the median, morphology and watershed techniques on the images of the 29 letters of the Turkish alphabet.

**Median Filter.** The median filter is a non-linear filter type and used to reduce the noise in the images (Mythili and Kavitha 2011). In the median filter, filtering takes place with the kernel matrix determined on the image (Göreke, Uzunhisarcıklı, and Güven 2014). The mathematical expression of the median filter is given in Equation 1 (Gupta, Chaurasia, and Shandilya 2015).

$$M_x = \text{Median}(w_x) = \begin{cases} x_{i(n+1)/2} & \text{if } n \text{ is odd} \\ \frac{1}{2} [x_{i(\frac{n}{2})} + x_{i(\frac{n}{2}+1)}] & \text{if } n \text{ is even} \end{cases} \quad (1)$$

In the equation, a size kernel is expressed as  $W(x)$ , and  $W(x)$ 's kernel values are expressed as  $x_1, x_2, x_3, \dots, x_n$ .

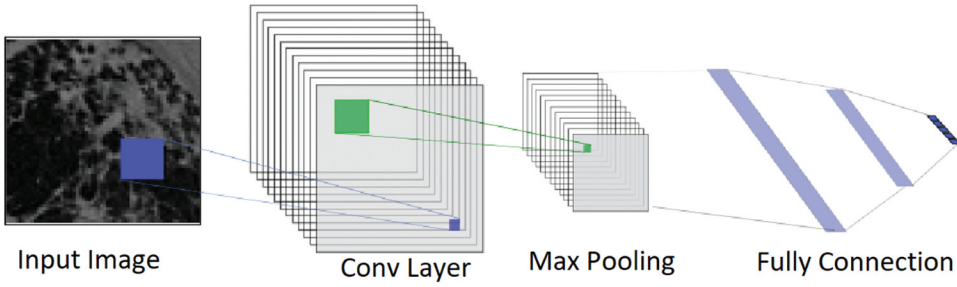
**Morphology Operations.** Morphological operations are used in many tasks such as removing the desired object from the background and distinguishing it from different objects, removing noise in the image and segmentation (Karhan et al. 2011). Dilation and erosion are basic morphology processes. All methods used in morphological operations are performed using these two basic processes (Atalı, Özkan, and Karayel 2016). Erosion removes the low values in the image, making the object smaller or thinner (Boztoprak, Çağlar, and Merdan 2007). Dilation is a morphological process that allows the object to grow or thicken by adding high values (Balci, Altun, and Taşdemir 2016).

**Watershed Segmentation.** The Watershed segmentation algorithm is derived from a mathematical morphology related to the topographic representation of an image (Nguyen, Worring, and Van Den Boomgaard 2003). Watershed segmentation is used to perform unsupervised image segmentation for the shapes or structures of the objects in the image (Sakai and Imiya 2009). Topographic representation refers to the shapes of original or artificial details on a surface (Yokoyama, Shirasawa, and Pike 2002). The mathematical morphology accepts gray-level images as a set of dots in three-dimensional space (Levner and Zhang 2007).

### **CNN Method**

CNN is one of the methods frequently used in image processing systems (Ballester and Araujo 2016). The CNN method is used in many different applications such as face recognition (Xing et al. 2019), object classification (Szegedy et al. 2019), traffic sign detection (Li et al. 2014). CNN method is mostly used for computer vision (Radu et al. 2020). The general structure of CNN architecture is given in Figure 2 (Li et al. 2014).

Convolutional layers seen in Figure 2 are one of the important components of CNN architecture. Two-dimensional matrices are used in the input and output layers of convoluted layers for image classification. The number of matrices used in the input and output layers may vary. The matrix for a single output is calculated by the mathematical expression given in equation 2 (Li et al. 2014).



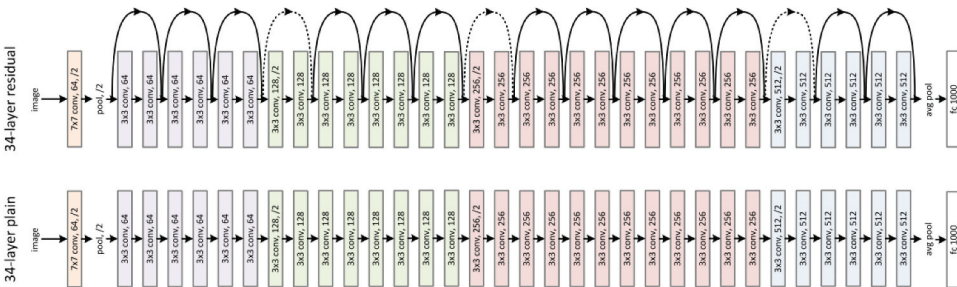
**Figure 2.** General structure of CNN architecture.

$$A_j = f \left( \sum_{i=1}^N I_i \cdot K_{i,j} + B_j \right) \tag{2}$$

In the equation,  $A_j$  represents the output matrix,  $I_i$  is the input matrix, is the kernel matrix, and  $B_j$  is the bias value. The pooling layer is used to reduce the output size of the convolutional layers (Baydar 2018). The fully connected layer, the latest layer of the CNN architecture, converts the features into space, allowing for easier classification of the output (Sainath et al. 2015).

Information about ResNet-50, AlexNet, CapsNet and Turkish Sign Language Net (TSLNet) deep learning architectures that yield the most successful results among 12 different deep learning architectures used in the study are given below.

**ResNet-50 Architecture.** ResNet-50 architecture is a pre-trained CNN architecture. Resnet architecture differs from other architectures with its micro-architecture structure. The success rate was increased by ignoring the changes between some layers and switching to the lower layers (Doğan and Türkoğlu 2018). In addition, ResNet-50 was chosen as the most successful model in the ImageNet competition held in 2015 with an error rate of 3.6% (He et al. 2015).



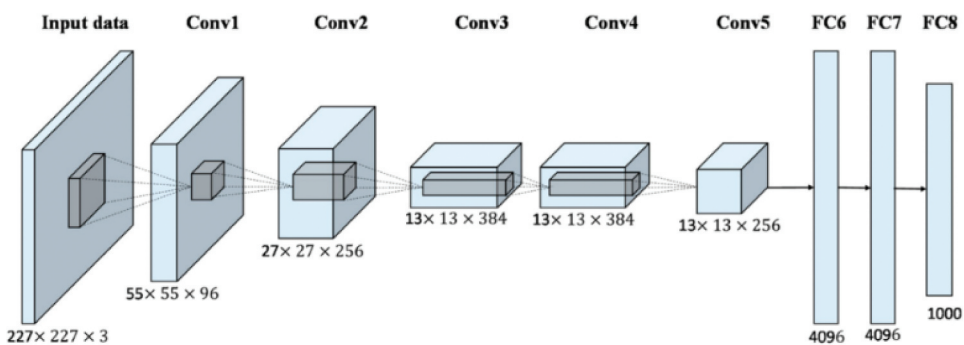
**Figure 3.** Resnet-50 architecture.

As seen in [Figure 3](#), the most important feature that distinguishes ResNet-50 architecture from other architectures is the addition of blocks that feed layers (He et al. 2015; Tan 2019).

**AlexNet Architecture.** AlexNet is a deep learning architecture developed by Alex Krizhevsky, Ilya Sutskever and Geoffrey Hinton. 25-layer CNN architecture consists of 5 convolution layers, 3 max-pooling layers, 2 dropout layers, 3 fully connected layers, 7 Relu layers, 2 normalization layers, a SoftMax layer, input and classification (output) layer. Layers of AlexNet architecture are shown in [Figure 4](#). AlexNet is a deep learning architecture that has achieved an accuracy rate close to 80% in the ImageNet database (Sainath et al. 2015; Doğan and Türkoğlu 2018; Tan 2019; Hinton et al. 2012; Han et al. 2017).

**CapsNet Architecture.** One of the frequently used architectures in deep learning methods is CNN architecture. Although CNN deep learning method gives successful results, data losses occur in the pooling layer. Therefore, alternative methods have been developed for CNN deep learning method. One of these methods is CapsNet networks. CapsNet is an architecture that enhances learning accuracy by creating a capsule layer of neuron network to prevent data loss that may occur in input parameters (Mukhometzianov and Carillo ; Körez and Barışcı 2019). In the CapsNet architecture, the depth is not created between layers, but it is created with the help of capsules. Thus, better feature extraction and better learning are realized. In CapsNet, apart from the Relu function used in the CNN method, the squashing function, whose input and output are vectors, is used (Tampubolon et al. 2019; Zhang, Tang, and Zhao 2019). General components of CapsNet architecture can be seen in [Figure 5](#) (Wen et al. 2019).

**TSLNet Architecture.** In the study, in addition to the pre-trained CNN deep learning architectures, a CNN architecture called the new TSLNet was designed as shown in [Figure 6](#). Image data were used as input in



**Figure 4.** AlexNet architecture.

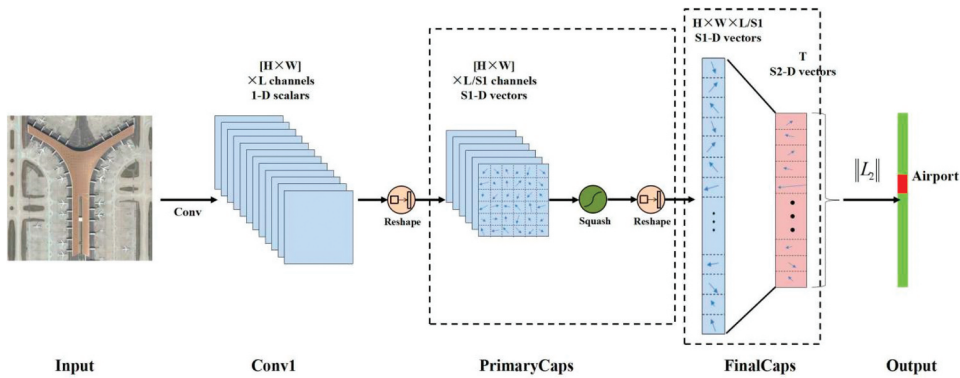
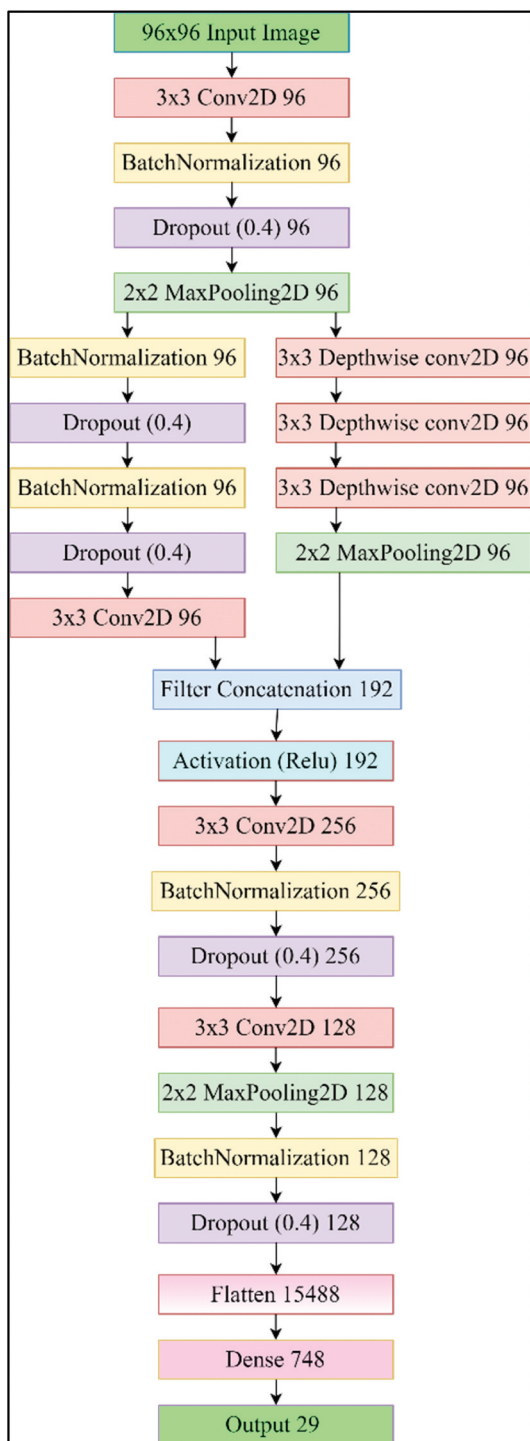


Figure 5. CapsNet architecture.

architecture. TSLNet architecture consists of 4 convolutional layers of 3x3, 3 depth wise convolutional layers of 3x3, 5 batch normalization layers, 5 dropout layers, 3 MaxPooling layers, 1 Relu activation layer and 2 fully connected layers.

### Tuning Learning Rate

Artificial neural networks are trained using the Stochastic gradient descent (SDD) optimization algorithm. Stochastic gradient descent with a large initial learning rate is widely used to train modern neural network architectures (Wen et al. 2019). One of the most important hyper parameters that must be tuned for training a neural network architecture is the learning rate for gradient descent. This hyper parameter controls the range that the model can change by scaling the magnitude of weight updates to minimize the loss rate of the network (Smith et al., 2017). If the learning rate is set low, very small updates occur in the weight of the network architecture, training progresses very slowly and the possibility of overfitting increases. In addition, if the learning rate is set too high, the accuracy value of the architecture decreases as the validation loss increases (Li, Wei, and Ma 2019; Takase, Oyama, and Kurihara 2018). In order to find solutions to these two problems that may occur in the study, the learning rate hyper parameter for all models was determined using the cyclical learning rate method. In the cyclical learning rate method, the minimum and maximum limits are determined and the learning speed changes cyclically between these limits. This method provides improved classification accuracy by using the cyclic learning rate instead of fixed values, without the need for any adjustment and generally with less iteration (Smith 2017).



**Figure 6.** Structure of the TSLNet architecture designed for this study.

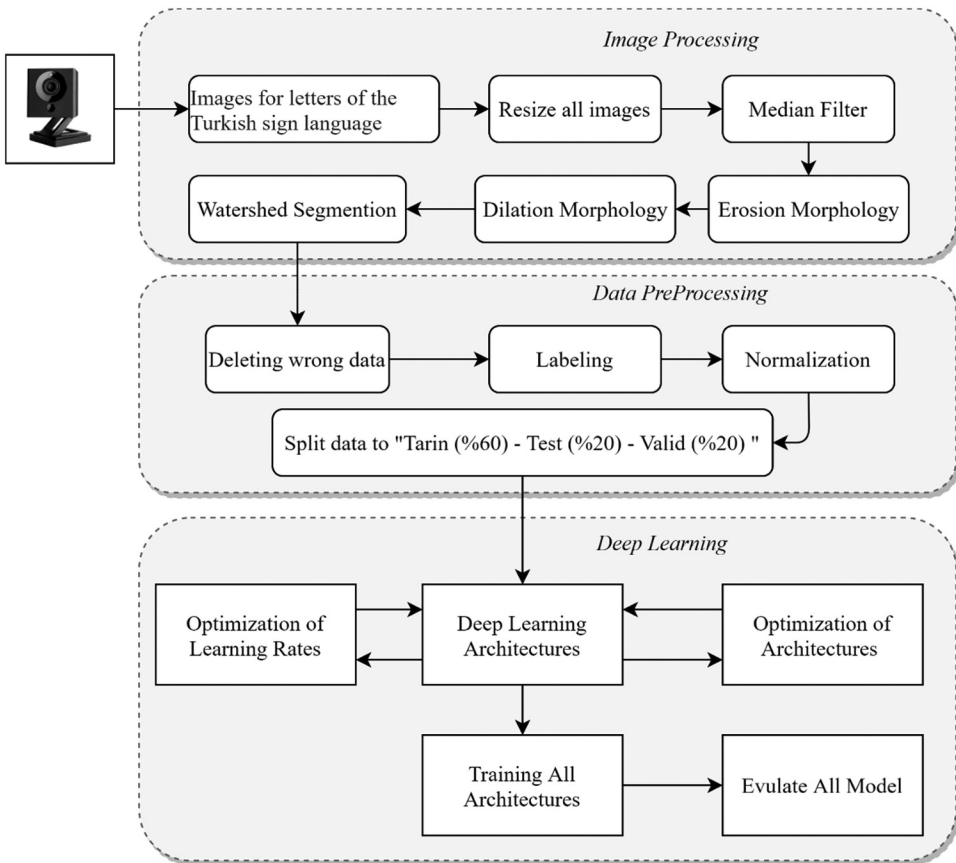


Figure 7. Work flow diagram.

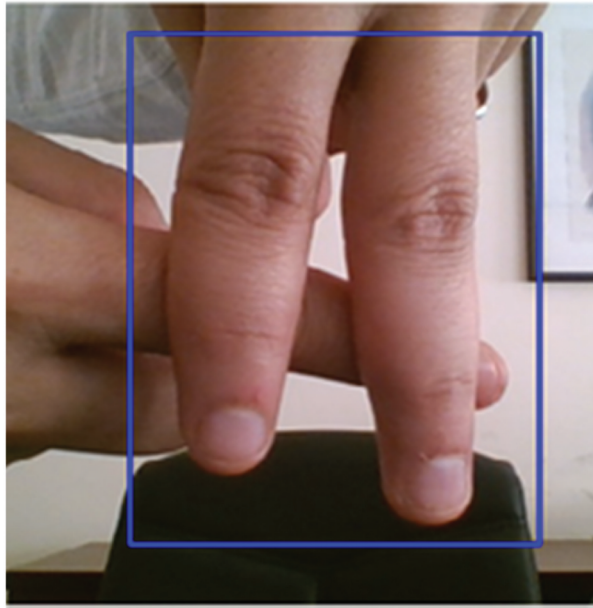
## Method

The workflow diagram of the study is given in Figure 7. As seen in the figure, in the first stage of the study, a total of 10223 pictures were collected for the 29 letters of the Turkish alphabet.

An object tracking system has been developed in the Python programming language so that data collection can be carried out quickly and easily. As shown in Figure 8, the Region of Interest (ROI) of the image was selected and the data set was created with the KCF Tracker.

Images of different sizes collected were brought to  $150 \times 150$  size to make the sizes standard. In the next step, median filtering with a  $3 \times 3$  kernel was used to perform smoothing on all images. After softening, erosion and dilation morphology processes were applied to the images to improve the result of the segmentation process. After erosion and dilation processes, the watershed





**Figure 8.** Determining the ROI for the letter A.

segmentation method was used on the images to increase learning success in deep learning methods. In [Figures 9](#) and [Figures 10](#), sample images of the original, grayscale, median filter, erosion, dilation and segmentation processes.

In addition to the Watershed method, the effect of Deeplab V3, Fully Connected Networks (FCN), Screen Content Image (SCI) and SegNet segmentation methods on processing time and accuracy in the TSLNet model was investigated. In the second stage of the study, data pre-processing operations were performed for data sets that will be used in deep learning methods. In data pre-processing, the images that do not include sign language images were detected manually and removed from the data set. Then, separate folders are created for each letter, and images are divided into 29 different labels. After labeling, normalization was performed by limiting the pixel values on the images between 0 and 1 using the Image Data Generator class in the Keras library of the Python programming language. In the final stage of data pre-processing, the data set is divided into three sections: 60% training, 20% test and 20% validation, with cross-validation.

In the final stage of the study, CapsNet, AlexNet and ResNet-50, DenseNet, VGG16, Xception, InceptionV3, NasNet, EffcientNet, Hitnet, Squeezenet and TSLNet architectures designed for the study were used to classify the letters in the Turkish alphabet. For the 12 deep learning techniques used in the study, the most appropriate learning rate was determined by optimizing the learning

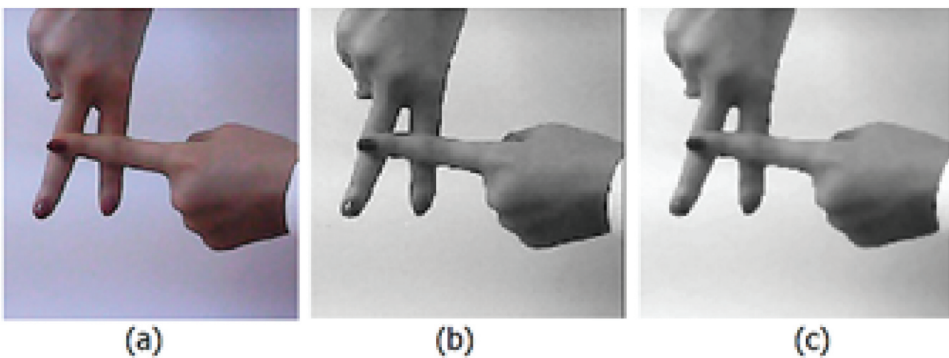


rate parameter. Finally, the deep learning technique that gives the most accurate result according to the determined learning rate was determined in the study. The pseudocode of the work carried out is given below.

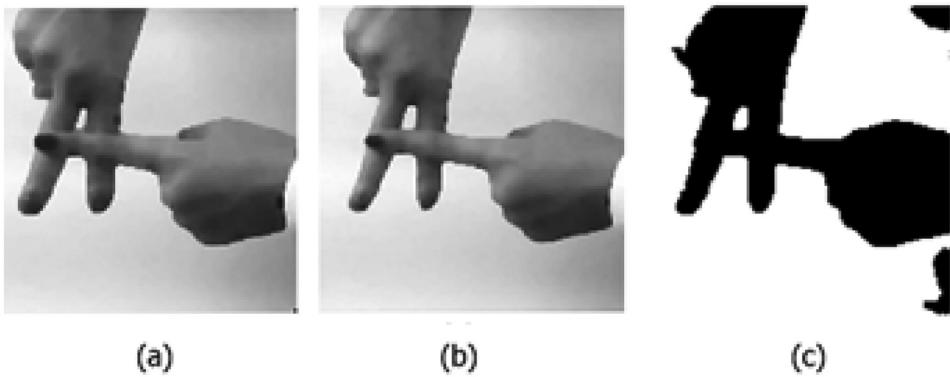
- 
1. **Begin**
  2. **For** all images:
  3. Image resize
  4. Image filter
  5. Image morphology
  6. Image watershed segmentation
  7. Labeling and data splitting
  8. **End for**
  9. Build CNN models
  10. Train all models
  11. Test and evaluate all models
  12. **End**
- 

## Research Findings

In the study, the letters in the TSL alphabet were classified by using libraries such as TensorFlow, Torch, fastai, Keras in Python programming language and CapsNet, AlexNet, ResNet-50, DenseNet, VGG16, Xception, InceptionV3, NasNet, EfficientNet, Hitnet, Squeezenet and TSLNet deep learning techniques. In the deep learning architectures used in the study; In order to obtain the best model, the learning rate hyper parameter was tuned using the cyclical learning rate method. Learning rate parameter values are a numerical value varying between 10<sup>-6</sup> and 1 for all architectures. When the learning rate value is chosen small, processing time and the possibility of overfitting increase. In addition, as the learning rate increases, the accuracy value of the model decreases as the loss of validation increases. In order to solve these two problems that may occur in the study, the cyclical learning rate method was



**Figure 9.** (a) original image (b) grayscale image (c) median filtered image.



**Figure 10.** (a) erosion process (b) dilation process (c) segmentation process.

used to determine the optimal learning rate values of all models. In [Figure 11](#), learning rate – loss graphs of all models are given with the software prepared in Python programming language.

When [Figure 11](#) is examined, it is seen that although the learning rate-loss graphs of all deep learning architectures used in the study are similar to each other, the optimal learning rate values are different from each other. The optimal learning rate values for the learning rate-loss charts obtained by the cyclical learning rate method are given in [Table 3](#).

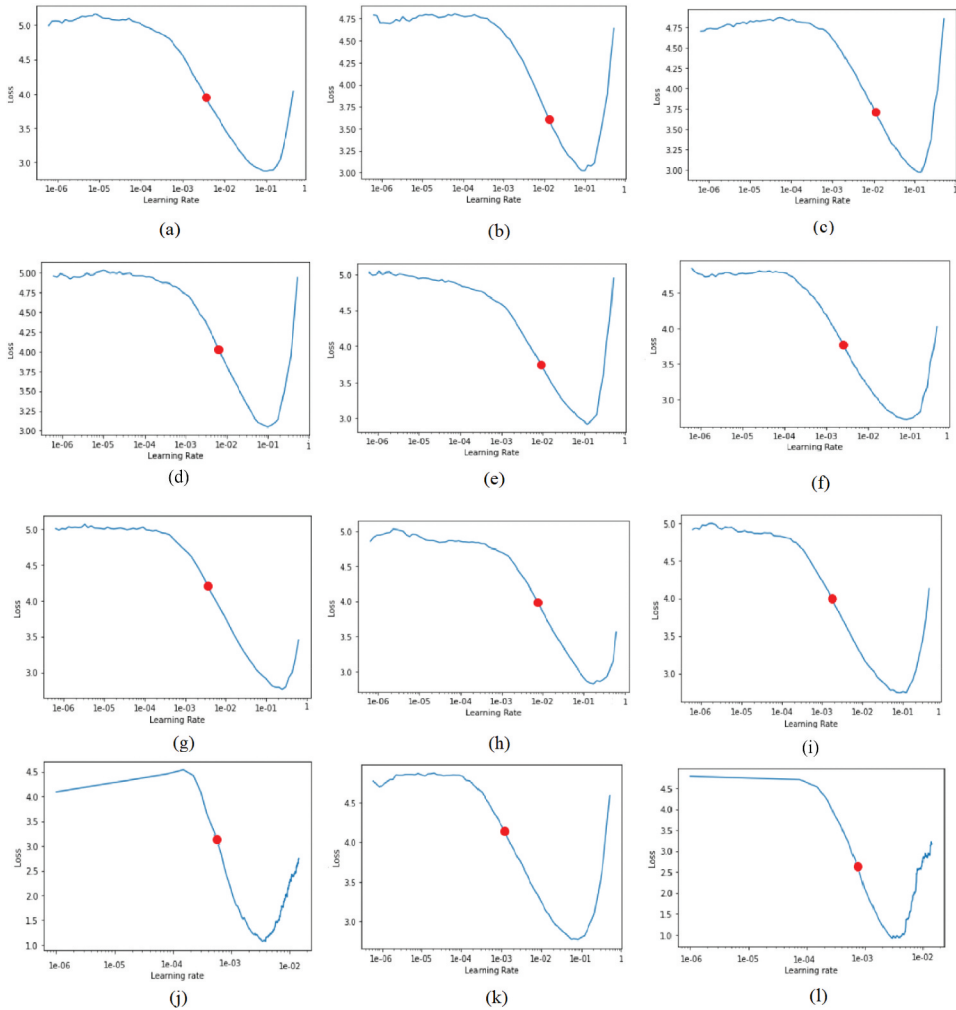
In addition to the Watershed method, Deeplab V3, Fully Connected Networks (FCN), Screen Content Image (SCI) and SegNet segmentation methods were used for segmentation in the study. The processing time information obtained by applying five different segmentation methods for a total of 10223 images in the data set is given in [Table 4](#).

**Table 3.** The optimal learning rate values obtained for all deep learning architectures used in the study.

Architecture	Optimal LR	Architecture	Optimal LR
DenseNet121	0.00363	Squeezenet	0.00363
VGG-16	0.01320	AlexNet	0.00759
Xception	0.01100	ResNet-50	0.00174
Inception V3	0.00631	HitNet	0.00020
NasNet	0.00912	CapsNet	0.00120
EfficientNet	0.00251	TSLNet	0.00016

**Table 4.** Segmentation processing times for Deeplab V3, SegNet, FCN, SCI and Watershed methods.

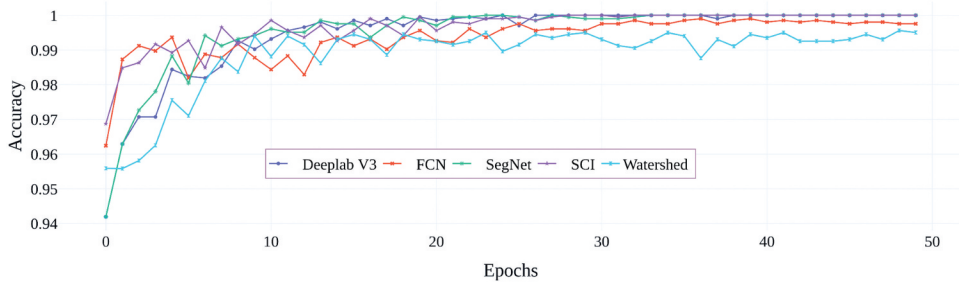
Method	Backbone	Time (s)
Deeplab V3	ResNet 101	44493
SegNet	ResNet 101	35321
FCN	ResNet 101	28017
SCI	Sparse Decomposition and Total Variation Minimization	21512
WaterShed	-	2902



**Figure 11.** Learning rate – loss graphs of the deep learning architectures used in the study (a) DenseNet121 (b) VGG-16 (c) Xception (d) Inception V3 (e) NasNet (f) EfficientNet (g) Squeezenet (h) AlexNet (i) ResNet- 50 (j) HitNet (k) CapsNet (l) TSLNet.

When [Table 4](#) is examined, it is seen that the Watershed segmentation method is approximately 15 times faster than Deeplab V3, 12 times faster than Segnet, approximately 10 times faster than FCN, 7 times faster than SCI in terms of processing time. [Figure 12](#) shows the graph of the accuracy results of the TSLNet architecture trained according to the segmentation results obtained.

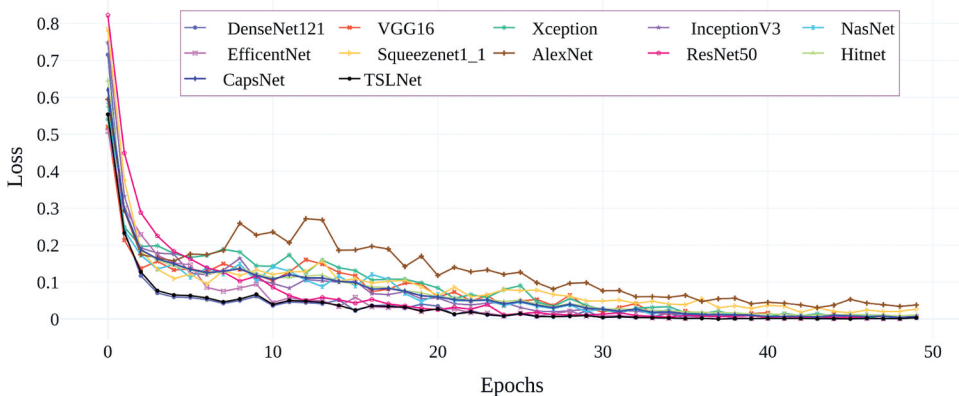
When [Figure 12](#) is examined, it is seen that Deeplab V3, SegNet and FCN deep learning based and SCI segmentation methods are slightly more successful than Watershed method. However, Watershed segmentation method appears to be very fast compared to other segmentation methods in terms of



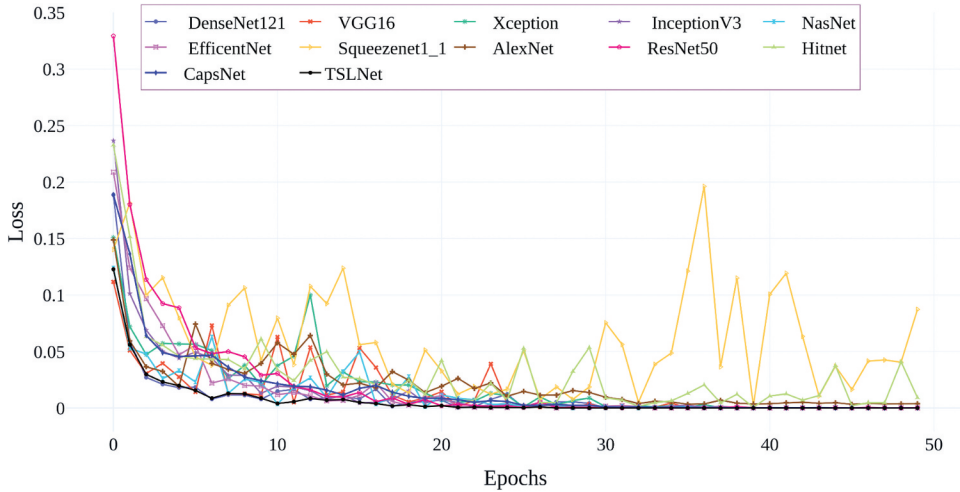
**Figure 12.** Validation accuracy graphs for Deeplab V3, FCN, SegNet, SCI and Watershed segmentation methods of TSLNet architecture.

processing time. Therefore, Watershed method, which is the fastest segmentation method in terms of processing time, was used in the study. CapsNet, AlexNet and ResNet-50, DenseNet, VGG16, Xception, Inception V3, NasNet, EffcientNet, Hitnet, Squeezenet and TSLNet architectures used in the study, and the performance results obtained by training with 50 epochs on the data set are given in Figure 13–15.

In Figure 13, the graph of the categorical cross entropy training losses obtained by training 12 deep learning architectures with a total of 50 epochs is given. When the graphic is examined, it is seen that all models used give similar loss values in 50 epoch training process. In terms of educational losses, it is seen that TSLNet and DenseNet architectures give lower education loss among 12 architectures. It is seen that AlexNet and Squeezenet architectures give the highest education losses. Figure 14 shows the graph of categorical cross entropy validation losses obtained by training 12 deep learning architectures with a total of 50 epochs. When the graph is examined, it is seen that TSLNet and DenseNet architectures, which give the most successful results,

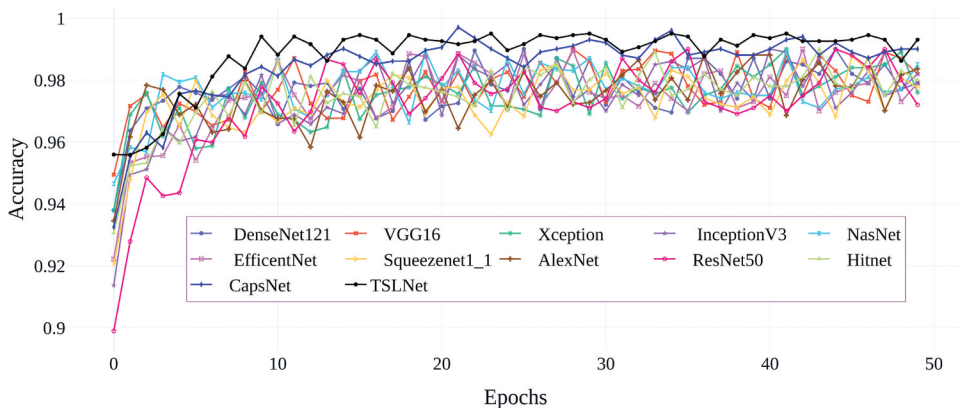


**Figure 13.** Loss results of deep learning architectures used in the study in the training phase.



**Figure 14.** Loss results of deep learning architectures used in the study in the validation phase.

have low validation losses, while SqueezeNet and Hitnet architectures have the highest validation loss values. Finally, in Figure 15, the graph of the accuracy values obtained by training 12 deep learning architectures used in the study with a total of 50 epochs is given. When the graph is examined, it is seen that CapsNet and TSLNet architectures give the most successful accuracy results. In Figure 15, it is seen that all deep learning architectures used give accuracy results between 96% and 99%. The successful results obtained from all models are an indication that the image processing methods used in the data pre-processing stage are correct and the data set is suitable for deep learning architectures. In the study, in order to see the effect of learning rate on



**Figure 15.** Accuracy values of deep learning architectures used in the study obtained on validation data.

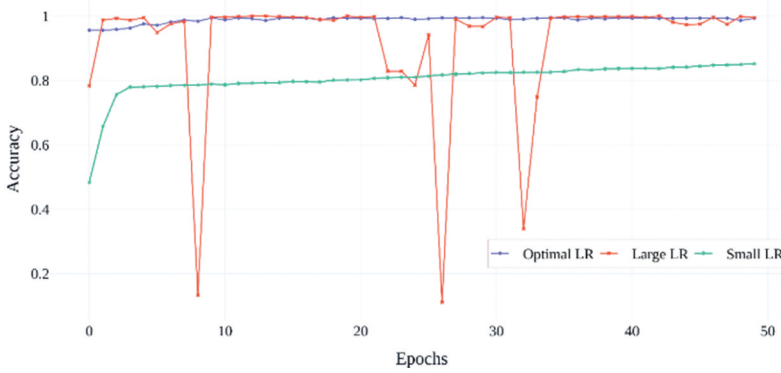


Figure 16. Validation accuracy results of TSLNet architecture with different learning rates.

TSLNet architecture, the training was carried out using the optimal learning rate value obtained and the learning rate values larger and smaller than the optimal value. Here, the optimal learning rate value obtained is taken as 0.1 for a value greater than 0.00016 and 0.000001 for a smaller value. The graphs of the effect of the learning rate hyper parameter used for the TSLNet architecture on the final results are shown between the Figures 16 and Figures 18.

When the validation accuracy results of the learning rate values in Figure 16 are examined; It is seen that the model trained by using the optimal learning rate value obtained for the TSLNet model, quickly reaches a more stable and high accuracy value. When the large learning rate is used, it is seen that the model is unstable because the accuracy value of the model in some epochs for the TSLNet architecture shows a very large decrease. Finally, when using the small learning rate, it is seen that the accuracy value is low, although the TSLNet architecture is stable. In Figure 17–18, the graphs of training and verification losses for different learning rate values in TSLNet architecture are

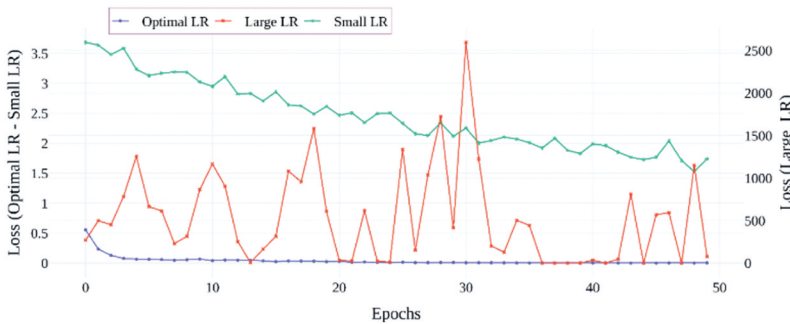
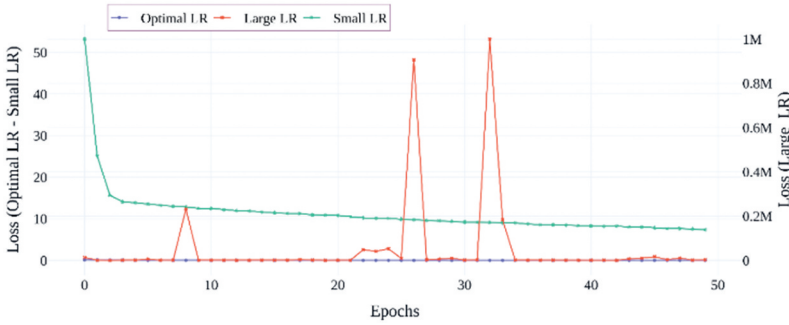


Figure 17. Training loss results of TSLNet architecture with different learning rates.



**Figure 18.** Validation loss results of TSLNet architecture with different learning rates.

given. When the graphs are examined, it is seen that the model is stable since the training and validation losses are small and decreasing in both graphs when the optimal learning rate value is used for the TSLNet architecture. For the large learning rate value, training and verification losses are quite high and unstable. For the small learning rate value, although the training and validation losses are stable and decreasing, the loss values are higher than the optimal learning rate value. As a result of these evaluations, it has been determined that using the optimal learning rate value determined for the TSLNet architecture is more appropriate in terms of both time and accuracy. In the study, the accuracy results of 12 different deep learning architectures on the test data set are given in [Table 5](#).

When [Table 5](#) is examined, it is seen that test accuracy values vary between 0.985 and 0.997. It is seen that the created TSLNet architecture within 12 deep learning architectures is more successful than all other architectures except CapsNet architecture, which has an accuracy of 0.996.

In the study, confusion matrices of CapsNet and TSLNet architectures are given in [Figures 19](#) and [Figures 20](#)

When the confusion matrices of CapsNet and TSLNet architectures given in [Figures 19](#) and [Figures 20](#) are examined, it is seen that both architectures successfully predict Turkish letters correctly. It is seen that CapsNet architecture classifies 2027 images correctly and only 6 images incorrectly in 2033 test

**Table 5.** Accuracy values of deep learning architectures used in the study obtained on test data.

Model	Accuracy	Model	Accuracy
DenseNet121	0.987	Squeezenet	0.992
VGG16	0.986	AlexNet	0.995
Xception	0.989	ResNet-50	0.996
InceptionV3	0.985	HitNet	0.989
NasNet	0.992	CapsNet	0.997
EfficientNet	0.991	<i>TSLNet</i>	<i>0.996</i>



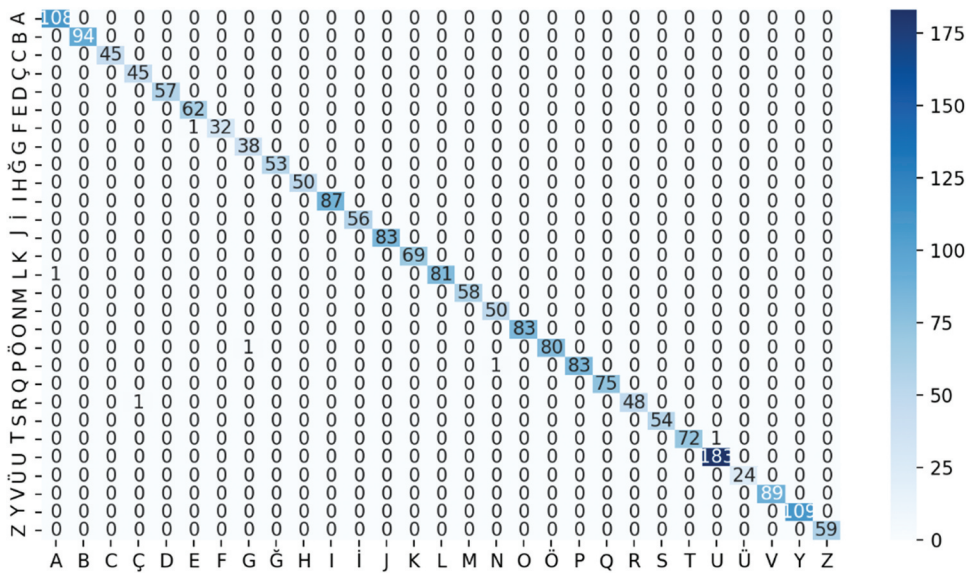


Figure 19. Confusion matrix results for CapsNet architecture.

data. TSLNet architecture classified 2025 images correctly and 8 images incorrectly. It has been determined that both architectures are quite successful on the test data.

### Discussion

It has been observed that there are many academic studies on sign language that use deep learning methods. Some of these academic studies are as follows. Rao et al. predicted selfie-mode sign language movements using the Convolutional Neural Network (CNN) deep learning method with an accuracy of 92.8% (Rao et al. 2018). In their study, Strezoski et al. evaluated 5 convolutional neural network models, including their proprietary Custom model, GoogleNet, AlexNet, LeNet and VGG models on the Marcel data set consisting of 6 hand signals (A, B, C, FIVE, POINT, V) performed by 24 people on three different backgrounds. As a result of the accuracy evaluation of the model performances, the proposed Custom model has achieved an accuracy rate of 84.32%. However, the highest accuracy value was obtained from the GoogleNet model with an accuracy rate of 90.41% (Strezoski et al. 2016). Garcia and Viesca developed a real-time sign language translator based on a convolutional neural network, using a pre-trained GoogleNet architecture, where they can implement transfer learning on American Sign Language. In the first case, 97% accuracy on the letters a and e and 74% accuracy on the letters a and k were obtained. (Garcia and Viesca 2016). Savur and Şahin, in their study, proposed an American Sign Language recognition system using



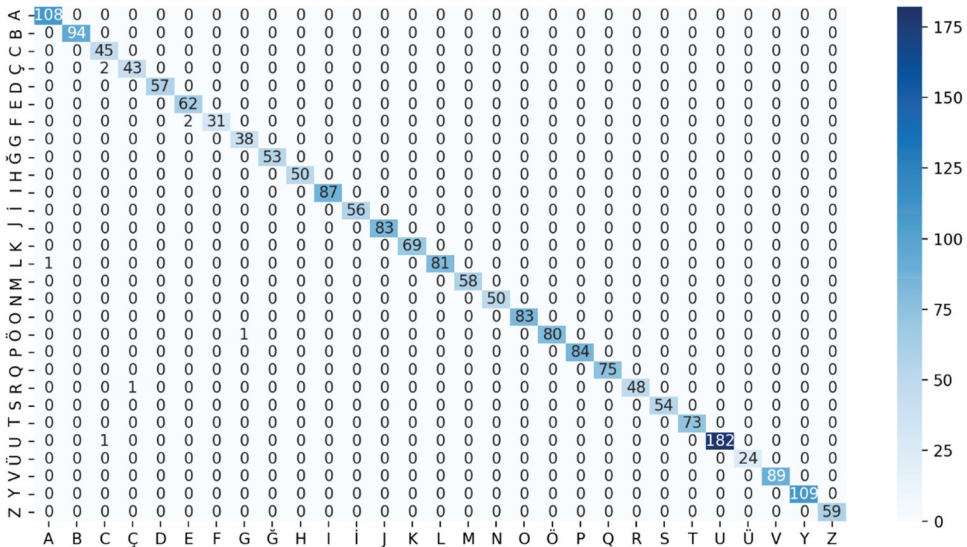


Figure 20. Confusion matrix results for TSLNet architecture.

armbands. In this system, a total of 2080 samples of American Sign Language were collected and classified using SVM. Experimental results show 82.3% accuracy in real time (Savur and Şahin 2015). Lin et al. proposed a hand sign recognition system based on CNN. To develop the created system, they applied the calibration of skin model, hand position and orientation and achieved a recognition accuracy of 95.96% (Lin, Hsu, and Chen 2014, August). Farhadi et al. demonstrated the effects of comparative distinguishing features and transfer learning on sign language using American Sign Language. Training was carried out on 90 words that make up the data set, and they obtained a transfer result of 62.5% with an error rate of 37.5% (Farhadi, Forsyth, and White 2007, June). Kin used a webcam for 29 Turkish sign language alphabets and specially defined characters to write text in his study. He carried out a software-based work on real-time recognition of the TSL alphabet and on creating a word or a sentence with characters. According to the test results, the system achieved 99.9% success on the data set and 90% in the tests performed with different users (Kın 2019). Savaş and Yıldırım used a data set consisting of letters and numbers created for American Sign Language (ASL) in their study. They used the CNN model to extract attributes from hand gestures and to classify them. A general success rate of 96% –98% was achieved in the study. A success rate varying between 45% and 52% was achieved in the test data set prepared in different environments and conditions (Savaş and Yıldırım, 2018). Within the scope of their study, Karacı et al. obtained a data set by using the 3D positions of all bones in the skeletal hand model measured by the Leap Motion sensor for 10 letters in

**Table 6.** Studies conducted in the literature on sign language using deep learning.

Authors of the study	Method	The sign language used in the study	Accuracy
Rao et al. (2018)	CNN	ISL	92,8%
Strezoski et al. (2016)	GoogleNet	A, B, C, FIVE, POINT, V (Marcel data set)	90,41%
Kin (2019)	CNN	TSL	90%
Savaş and Yıldırım (2018)	CNN	ASL	96%-98%
Karacı, Akyol, and Gültepe (2018)	DNN	TSL	100%
	ANN		93%
	Decision Tree		98,5%
Chong and Lee (2018)	SVM	ASL	80,30%
	DNN		93,81%
Tao et al. (2018)	CNN	ASL	80,1% (loo) 99,7% (hh)
Çelik and Odabaş (2020)	CNN+LSTM	TSL	97%
Yalçinkaya, Atvar, and Duygulu (2016)	MHI	TSL	95%
Beşer et al. (2018)	CapsNet	0–9 numbers	92,2%
Unutmaz, Karaca, and Güll (2019)	CNN	TSL	92,60% 99,25%
Farhadi, Forsyth, and White (2007, June)	Transfer Learning	ASL	62,5%
Garcia and Viesca (2016)	GoogleNet	ASL	97% (a,e) 74% (a, k)
Bird, Ekárt, and Faria (2020)	transfer learning multimodality approach	ASL	82,5%
Thiracitta and Gunawan (2021)	transfer learning+CNN	SLR	97,5%
<b>In this study</b>	<b>TSLNet</b>	<b>TSL</b>	<b>99,6%</b>

the TSL. In the study, Artificial Neural Network, Deep Learning and Decision Tree based models were designed. Accuracy rates for the data set with 120 features in the study are 100% for DNN, 99% for ANN and 100% for Decision Tree. Accuracy rates for the data set with 390 features in the study are 100% for DNN, 93% for ANN and 98,85% for Decision Tree (Karacı, Akyol, and Gültepe 2018). In their study, Chong and Lee developed a system that aims to recognize ASL, which consists of 26 letters and 10 numbers using Leap Motion Controller (LMC). When the experimental results were examined, recognition rates of 80.30% and 93.81% were obtained from the Support Vector Machine (SVM) and Deep Neural Network (DNN) models used for 26 letters respectively, and for 36 letters, 88.79% recognition rate was obtained from the DNN model (Chong and Lee 2018). In their study, Tao et al. presented a proposal for the recognition of the ASL alphabet by using hand-skeleton data, Leap Motion Controller and two infrared radiation images. Within the scope of the study, they created a data set by using 24 static hand signs belonging to the ASL alphabet, which were made by 5 people to evaluate the created model. As a result of the study, they developed a CNN vision module. With the developed module, they obtained an accuracy of 80.1% with leave-one-out (loo) and 99.7% with half-half (hh) (Tao et al. 2018). In their study, Çelik and Odabaş detected hand movements using a webcam and

translated them into sign language. In the study, they used two kinds of deep learning techniques, CNN and LSTM, and they made 97% successful predictions in CNN + LSTM models trained with sign language movements of 10 numbers and 29 letters (Çelik and Odabaş 2020). In their study, Yalçinkaya et al. used the Motion History Image (MHI) algorithm, which is a motion recognition algorithm, to perform text translation in response to dynamic hand/arm/head movements obtained from the camera and achieved a 95% success rate for 8 movements in the training set (Yalçinkaya, Atvar, and Duygulu 2016). Beşer et al. conducted a study by using a capsule network and dynamic routing algorithm to classify motion images of sign language. They achieved an average success rate of 94.2% with the capsule network model applied on the data set created by taking 10 sample images for each expression from 218 different people (Beşer et al. 2018). In their study, Unutmaz et al. developed a system that automatically converts the letters of the TSL into words using Convolutional Neural Networks. For the developed system, they used the skeleton information obtained from Microsoft's Kinect device. Different classification methods such as Decision tree, K nearest neighbor, Linear SVM and Gauss SVM were compared with the proposed CNN network. The success rate of the proposed CNN network has been achieved between 92.60% and 99.25% and CNN network has been found to have best performance (Unutmaz, Karaca, and Güll 2019). In the study carried out by Lee et al., 100 samples were taken for each letter of the ASL alphabet and the model was trained with 2600 samples. For the recognition of static and dynamic letters in the alphabet, features such as the angles between the fingers and the distance between the finger positions were used as classification inputs, and the Long-Short-Term Memory Recurrent Neural Network with the k-Nearest-Neighbor method was adopted in the processing of the model inputs. As a result of the experiments, an average of 99.44% accuracy and 91.82% accuracy were obtained in 5-fold cross validation with the use of the jump motion controller, and it was revealed that it has a stronger performance overall than those using SVM (Lee et al. 2021). Abiyev et al., in their study, first used One Shot Multi-Box Detection (SSD) architecture to detect hand signals. They proposed a deep learning model based on Inception v3 and Support Vector Machine (SVM) to perform feature extraction and classification on the hand gestures. An average of 99.9% accuracy rate was obtained by using the hybrid system used on the data set created with 24-letter ASL and the cross-validation approach (Abiyev, Arslan, and Idoko 2020). Wu et al. proposed a sign language recognition system using electromyography and arm sensors. They obtained 40 samples of American sign language and classified them using Naïve Bayes, Nearest Neighbor, Decision Tree and LibSVM classifiers. The experimental results obtained revealed that SVM outperformed all other classification methods (Wu et al. 2015). Aryanie and Heryadi developed a camera-based sign recognition system. In total, 5000 samples of 5000

American Sign languages were obtained. In the developed system, features were extracted using the color histogram and PCA was used to reduce the size of the extracted feature set. Collected signals were classified using KNN and the best 99.8% accuracy rate was achieved (Aryanie and Heryadi 2015). Hammadi et al. used the 3BCNN method for sign language recognition. The method was evaluated in both person-dependent and person-independent modes for the three datasets created. Recognition rates of 98.12%, 100% and 76.67%, respectively, were obtained for the person-dependent mode, and 84.38%, 34.9% and 70%, respectively, for the person-independent mode (Al-Hammadi et al., 2020). Azar and Seyedarabi developed a model for the recognition of dynamic Persian Sign Language using Gaussian probability density functions and Hidden Markov Model (HMM). The data set was created by obtaining 1200 video images by making 20 dynamic signs from 12 people using white gloves. From each video obtained, information such as the trajectory of the hands and the shape of the hands were extracted using a simple region enlargement technique. Individual-dependent and individual-independent experiments were carried out on the proposed system and an average accuracy of 97.48% was obtained (Azar and Seyedarabi 2020). Raghuvvera et al. have developed an effective system for translating hand gestures in Indian Sign Language (ISL) into English text and speech. Feature extraction was performed for 4600 hand gestures obtained through Microsoft Kinect. In the developed system, Accelerated Robust Features, Directed Gradient Histogram and Local Binary Models, which are feature extraction methods using Dester Vector Machines, were brought together and the average recognition accuracy was increased up to 71.85%. They also achieved 100% accuracy for signs representing 9, A, F, G, H, N, and P (Raghuvvera et al. 2020). Bird et al. compared single image classification (88.14%) and Leap Motion data classification (72.73%) approaches with multi modal late fusion using British and American Sign Language. They concluded that a multi modal approach outperforms two single sensors during both training and classification of invisible data. At the same time, it was stated in the study that transfer learning improves the sign language recognition capabilities of Leap Motion and multi modal classification approaches, and the best model overall for American Sign Language classification is the transfer learning multi modal approach, which achieves 82.55% accuracy (Bird, Ekárt, and Faria 2020). Barbhuiya et al. used CNN architecture for both letters and numbers of American Sign Language. The CNN architecture is based on both modified AlexNet and modified VGG16 models for the classification process. With the developed model, 99.82% recognition accuracy was obtained in the study (Barbhuiya, Karsh, and Jain 2021). Buckley et al. developed a real-time British Sign Language recognition system using a webcam. The CNN architecture used for the classification of the dataset consisting of 11,875 images also identified 19 single-handed and double-handed static marks, giving an

average recognition accuracy of 89% during the test phase (Buckley, Sherrett, and Secco 2021). Thiracitta and Gunawan developed a Convolutional Neural Network (CNN) model, which is an Inflated 3D model combined with transfer learning method from ImageNet and Kinectic dataset to overcome Sign Language Recognition (SLR) problems. Using 2 people, a dataset containing 200 videos of 10 words was created and trained. As a result of the study, it was determined that transfer learning on small datasets was effective by obtaining 97.5% accuracy on the test dataset. Thiracitta and Gunawan (2021). Sharma and Kumar used the 3D CNN model to classify 100 words out of more than 3300 English words obtained from 6 people. They stated that with this study, it provided automatic recognition of signs and gave better results than the latest models (Sharma and Kumar 2021). Kamruzzaman proposed a CNN applied vision-based system for recognizing and translating Arabic sign language letters into spoken language and achieved an accuracy rate of 90% (Kamruzzaman 2020). Al-Jarrah and Al-Omari, developed a system for recognizing the letters of Arabic Sign Language in their study. The developed system is a visual-based system that does not rely on the use of gloves or visual signals. The study was conducted by training 30 ANFIS models, each dedicated to the recognition of a particular gesture. After the preprocessing step, features are extracted from the image and a twin approach is used in which boundary and region features are calculated. From the experimental results, they obtained that the proposed system achieves 100% recognition rate when using about 19 rules per ANFIS model and 97.5% recognition rate when using about 10 rules (Al-Jarrah and Al-Omari 2007). With the study carried out in Table 6, studies on determining sign language with deep learning methods in the literature were examined and compared. When the results in Table 6 are examined, it is thought that the TSLNet architecture designed for the study will provide a highly accurate result and bring a different perspective to the academic literature.

In the study, a total of 10223 images belonging to 5 people in the data set were trained with deep learning architectures and more stable models were obtained. It has been determined that the specially designed TSLNet architecture in the study works faster than the CapsNet, AlexNet and ResNet-50, DenseNet, VGG16, Xception, Inception V3, NasNet, EfficentNet, Hitnet, Squezenet deep learning methods. In addition, when the 99.6% accuracy rate obtained in TSLNet architecture is compared with the results obtained from artificial intelligence models developed for different languages, it has been seen that it is more successful than other artificial intelligence methods. One of the important disadvantages of TSLNet architecture is that only Turkish letters can be detected with TSLNet architecture, while the success of TSLNet architecture on words or word groups has not been tested. It would

be possible to develop the TSLNet architecture by testing the success of the images obtained from different individuals by increasing the number of five individuals used in the study.

## Results

In this study, a system with a deep learning model has been developed for the Turkish sign language alphabet for hearing impaired people, which is one of the important problems of today. The study is based on two main purposes. The first purpose is; since TSL is not known by many people in the society, it is thought that the communication problem between people with hearing impairment and people who do not know TSL can be solved with the software implemented in the study. The second purpose is; It is thought that the model, which has high accuracy rate obtained with deep learning architectures for letter images in TSL, will create an infrastructure for combining letters into word form in real time in the future. In this developed system, a data set was created by taking 10223 images for 29 letters of the Turkish alphabet. The resulting dataset was improved using image processing methods such as filtering, morphology and segmentation. Classification processes were performed on the improved images using CapsNet, AlexNet and ResNet-50, DenseNet, VGG16, Xception, InceptionV3, NasNet, EffcientNet, Hitnet, Squeezenet and TSLNet deep learning architectures designed for the study. When deep learning models are examined, CapsNet and TSLNet models have been determined as the most successful models. In the study, the CNN-based TSLNet deep learning architecture, which was originally designed, and 11 deep learning architectures popularly used in image analysis in the literature were compared. Thus, the performance and compatibility of TSL deep learning architecture compared to other deep learning architectures were evaluated. As a result of the evaluation, the TSLNet architecture designed for the study gave very close results with CapsNet and more successful results than all other architectures. The reasons for obtaining such results from all of the deep learning models used are:

- The images obtained from the camera are good quality
- Application of image processing techniques
- The number of data sufficient for the used deep learning techniques
- Appropriate selection of learning rate for deep learning models

In addition, in future studies, it is thought that the TSLNet model, which is obtained by translating the voice of the person into sign language in order to communicate with people with hearing impairment, in synchronization with



the model used in the study, can serve as a basis. The limitations of the study carried on and the factors affecting the applied TSLNet deep learning method are given below.

- Only letters belonging to TSL have been identified. Numbers, words and expressions were not detected.
- It has been determined that the success of the TSLNet model decreases in the real world environment.
- As the distance from which the images are taken increases, the success of the TSLNet model decreases.
- The success of the TSLNet model decreases in environments where the light intensity varies greatly.

In the future, studies can be carried out to eliminate the problems given above.

### Disclosure Statement

No potential conflict of interest was reported by the author(s).

### ORCID

Bekir Aksoy  <http://orcid.org/0000-0001-8052-9411>

Osamah Khaled Musleh Salman  <http://orcid.org/0000-0001-6526-4793>

Özge Ekrem  <http://orcid.org/0000-0001-9142-405X>

### Compliance With Ethical Standards

Ethical approval for the study and informed consent (information text on processing personal data from 5 people) was obtained and there is no conflict of interest.

### References

- Abiyev, R. H., M. Arslan, and J. B. Idoko. 2020. Sign language translation using deep convolutional neural networks. *KSII Transactions on Internet and Information Systems (TIIS)* 14 (2):631–53.
- Al-Hammadi, M., G. Muhammad, W. Abdul, M. Alsulaiman, M. A. Bencherif, and M. A. Mekhtiche. 2020. Hand gesture recognition for sign language using 3DCNN. *IEEE Access* 8:79491–509.
- Al-Jarrah, O., and F. A. Al-Omari. 2007. Improving gesture recognition in the Arabic sign language using texture analysis. *Applied Artificial Intelligence* 21 (1):11–33.
- Aly, W., S. Aly, and S. Almotairi. 2019. User-Independent American Sign Language Alphabet Recognition Based on Depth Image and PCANet Features. *IEEE Access* (2019 (7):123138–50.
- Aran, O., I. Ari, A. Guvensan, H. Haberdar, Z. Kurt, I. Turkmen, and L. Akarun. A database of non-manual signs in turkish sign language IEEE 15th Signal Processing and Communications Applications, 2007, June, pp. 1–4, Eskisehir, Turkey: IEEE.

- Aryanie, D., and Y. Heryadi (2015). American sign language-based finger-spelling recognition using k-Nearest Neighbors classifier. In 2015 3rd International Conference on Information and Communication Technology (ICoICT) (pp. 533–36). Nusa Dua, Bali, Indonesia: IEEE.
- Atalı, G., S. S. Özkan, and D. Karayel. 2016. Morfolojik Görüntü İşleme Tekniği ile Yapay Sinir Ağlarında Görüntü Tahribat Analizi. *Akademik Platform Mühendislik Ve Fen Bilimleri Dergisi* 4:1.
- Azar, S. G., and H. Seyedarabi. 2020. Trajectory-based recognition of dynamic Persian sign language using hidden Markov model. *Computer Speech & Language* 61:101053.
- Balci, M., A. A. Altun, and Ş. Taşdemir. 2016. Görüntü İşleme Teknikleri Kullanılarak Napolyon Tipi Kirazların Sınıflandırılması. *Selçuk-Teknik Dergisi* 15 (3):221–37.
- Ballester, P., and R. M. Araujo. (2016). On the performance of GoogLeNet and AlexNet applied to sketches. In Thirtieth AAAI Conference on Artificial Intelligence, Phoenix, Arizona, US .
- Barbhuiya, A. A., R. K. Karsh, and R. Jain. 2021. CNN based feature extraction and classification for sign language. *Multimedia Tools and Applications* 80 (2):3051–69.
- Baydar, B. (2018). Convolutional neural network based brain MRI segmentation: Master's thesis. MIDDLE EAST TECHNICAL UNIVERSITY.
- Bengio, Y. 2009. Learning Deep Architectures for AI. *Found. Trends® Mach. Learn* 2 (1):1–127.
- Beşer, F., M. A. Kızrak, B. Bolat, and T. Yıldırım. (2018). Kapsül ağları kullanılarak işaret dilinin tanınması. 2018 26. Sinyal İşleme ve İletişim Uygulamaları Konferansı (SIU), İzmir, s. 1–4, doi: [10.1109/SIU.2018.8404385](https://doi.org/10.1109/SIU.2018.8404385).
- Bird, J. J., A. Ekárt, and D. R. Faria. 2020. British sign language recognition via late fusion of computer vision and leap motion with transfer learning to american sign language. *Sensors* 20 (18):5151.
- Boztoprak, H., M. F. Çağlar, and M. Merdan. (2007). Alternatif morfolojik bir yöntemle plaka yerini saptama. XII. Elektrik, Elektronik, Bilgisayar, Biyomedikal Mühendisliği Ulusal Kongresi. Eskişehir, Kasım.
- Buckley, N., L. Sherrett, and E. L. Secco (2021). A CNN sign language recognition system with single & double-handed gestures.
- Çelik, Ö., , and A. Odabaş. 2020. Sign2Text: Konvolüsyonel Sinir Ağları Kullanarak Türk İşaret Dili Tanıma. *Avrupa Bilim Ve Teknoloji Dergisi* (19):923–34.
- Chong, T. W., and B. G. Lee. 2018. American sign language recognition using leap motion controller with machine learning approach. *Sensors* 18 (10):3554.
- Chuan, C. H., E. Regina, and C. Guardino. (2014). Leap Motion Sensor Kullanarak American Sign Language Recognition. 13th International Conference on Machine Learning and Applications, Detroit, MI, USA, 2014, s. 541–44, doi: [10.1109/ICMLA.2014.110](https://doi.org/10.1109/ICMLA.2014.110).
- Deng, L., and D. Yu. (2014). Deep Learning: Methods and Applications. Found. Trends® Signal Process., vol. 7, no. 3–4, pp. 197–387.
- Devineau, G., F. Moutarde, W. Xi, and J. Yang. (2018). Deep Learning for Hand Gesture Recognition on Skeletal Data. 13th IEEE International Conference on Automatic Face & Gesture Recognition, 2018, Xi'an, China, 2018, pp. 106–13, doi: [10.1109/FG.2018.00025](https://doi.org/10.1109/FG.2018.00025).
- Doğan, F., and İ. Türkoğlu. 2018. Derin öğrenme algoritmalarının yaprak sınıflandırma başarımlarının karşılaştırılması. *Sakarya University Journal of Computer and Information Sciences* 1 (1):10–21.
- Farhadi, A., D. Forsyth, and R. White (2007, June). Transfer learning in sign language. In 2007 IEEE Conference on Computer Vision and Pattern Recognition (pp. 1–8). Minneapolis, MN, USA: IEEE.
- Garcia, B., and S. A. Viesca. 2016. Real-time American sign language recognition with convolutional neural networks. *Convolutional Neural Networks for Visual Recognition* 2:225–32.



- Göreke, V., E. Uzunhisarcıklı, and A. Güven. (2014). Gri Seviyeli Eşoluşum Matrisleri Kullanılarak Sayısal Mamogram Görüntüsünden Doku Özneliklerinin Çıkarılması ve Yapay Sinir Ağı ile Kitle Tespiti. In *Tıp Teknolojileri Ulusal Kongresi-TıpTekno'14*, Nevşehir, Türkiye, s. 95-98.
- Gupta, V., V. Chaurasia, and M. Shandilya. 2015. Random-valued impulse noise removal using adaptive dual threshold median filter. *Journal of Visual Communication and Image Representation* 26:296–304.
- Han, X., Y. Zhong, L. Cao, and L. Zhang. 2017. Pre-trained alexnet architecture with pyramid pooling and supervision for high spatial resolution remote sensing image scene classification. *Remote Sensing* 9 (8):848.
- Hasan, H., and S. Abdul-Kareem. 2014. *RETRACTED ARTICLE: Static Hand Gesture Recognition Using Neural networks*. *Artificial Intelligence Review* 41 (2):147–81.
- He, K., X. Zhang, S. Ren, and J. Sun. (2016). Deep Residual Learning for Image Recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, NV, USA, pp. 770–778.
- Hinton, G. E., N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov. (2012). Improving neural networks by preventing co-adaptation of feature detectors. Cornell University, arXiv, preprint arXiv:1207.0580.
- Kamruzzaman, M. M. (2020). Arabic sign language recognition and generating Arabic speech using convolutional neural network. *Wireless Communications and Mobile Computing*
- Kapuściński, T., and D. Warchoń. 2020. (2020). *Hand Posture Recognition Using Skeletal Data and Distance Descriptor*. *Applied Sciences* 10 (6):2132.
- Karacı, A., K. Akyol, and Y. Gültepe. (2018). Turkish Sign Language Alphabet Recognition with Leap Motion. *International Conference on Advanced Technologies, Computer Engineering and Science (ICATCES'18)*, May 11-13, pp. 189–92, Sarfanbolu, Turkey, Sensors, 18 (10),3554.
- Karhan, M., M. O. Oktay, Z. Karhan, and H. Demir. (2011). Morfolojik görüntü işleme yöntemleri ile kayıslarda yaprak delen (çil) hastalığı sonucu oluşan lekelerin tespiti. In *6 th International Advanced Technologies Symposium (IATS'11)*, pp. 172–76. Elazığ, Türkiye.
- Kın, Z. B. (2019). Türk işaret dili alfabesinin derin öğrenme yöntemi ile sınıflandırılması. Master's thesis. Başkent Üniversitesi Fen Bilimleri Enstitüsü.
- Körez, A., and N. Barışçı. (2019). İnsansız Hava aracı (İHA) Görüntülerindeki Nesnel-erin Kapsül Ağları Kullanılarak Sınıflandırılması. ISAS, Ankara, Türkiye.
- Lee, C. K., K. K. Ng, C. H. Chen, H. C. Lau, S. Y. Chung, and T. Tsoi. 2021. American sign language recognition and training method with recurrent neural network. *Expert Systems with Applications* 167:114403.
- Levner, I., and H. Zhang. 2007. Classification-driven watershed segmentation. *IEEE Transactions on Image Processing* 16 (5):1437–45.
- Li, Q., W. Cai, X. Wang, Y. Zhou, D. D. Feng, and M. Chen. (2014). Medical image classification with convolutional neural network. In *2014 13th international conference on control automation robotics & vision (ICARCV)*, pp. 844–48, Marina Bay Sands, Singapore: IEEE.
- Li, Y., C. Wei, and T. Ma. (2019). Towards explaining the regularization effect of initial large learning rate in training neural networks. Cornell University, arXiv, preprint arXiv:1907.04595.
- Lin, H. I., M. H. Hsu, and W. K. Chen (2014, August). Human hand gesture recognition using a convolution neural network. In *2014 IEEE International Conference on Automation Science and Engineering (CASE)* (pp. 1038–43). New Taipei, Taiwan: IEEE.
- Mukhometzianov, R., and J. Carillo. (2018). CapsNet comparative performance evaluation for image classification. Cornell University, arXiv, arXiv preprint arXiv:1805.11195.

- Mythili, C., and V. Kavitha. 2011. Efficient technique for color image noise reduction. *The Research Bulletin of Jordan ACM* 2 (3):41–44.
- Nguyen, H. T., M. Worring, and R. Van Den Boomgaard. 2003. Watersnakes Energy-driven watershed segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 25 (3):330–42.
- Oyedotun, O. K., and A. Khashman. 2017. Deep learning in vision-based static hand gesture recognition. *Neural Computing & Applications* 28 (12):3941–51.
- Özkan, İ. N. İ. K., and E. Ülke. 2017. Derin Öğrenme ve Görüntü Analizinde Kullanılan Derin Öğrenme Modelleri. *Gaziosmanpasa Bilimsel Arastırma Dergisi* 6 (3):85–104.
- Pigou, L., S. Dieleman, P. J. Kindermans, and B. Schrauwen. (2014). Sign language recognition using convolutional neural networks. In European Conference on Computer Vision, 2014, September, pp. 572–78, Springer, Cham.
- Radu, V., K. Kaszyk, Y. Wen, J. Turner, J. Cano, E. J. Crowley, and M. O’Boyle. (2020). Performance aware convolutional neural network channel pruning for embedded GPUs. In 2019 IEEE International Symposium on Workload Characterization (IISWC), Orlando, FL, USA, pp. 24–34, doi: [10.1109/IISWC47752.2019.9042000](https://doi.org/10.1109/IISWC47752.2019.9042000).
- Raghuveera, T., R. Deepthi, R. Mangalashri, and R. Akshaya. 2020. A depth-based Indian sign language recognition using microsoft kinect. *Sādhanā* 45 (1):1–13.
- Rao, G. A., K. Syamala, P. V. V. Kishore, and A. S. C. S. Sastry. (2018). Deep convolutional neural networks for sign language recognition. In 2018 Conference on Signal Processing and Communication Engineering Systems (SPACES), Vijayawada, India, pp. 194–97.
- Sainath, T. N., O. Vinyals, A. Senior, and H. Sak. (2015). Convolutional, long short-term memory, fully connected deep neural networks. In 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 4580–84), South Brisbane, QLD, Australia: IEEE.
- Sakai, T., and A. Imiya. (2009). Validation of Watershed Regions by Scale-Space Statistics. June, In International Conference on Scale Space and Variational Methods in Computer Vision pp. 175–86, Springer, Berlin, Heidelberg.
- Savaş, O. O., and T. Yildirim. (2018). The Effect of Convolutional Neural Network Parameters on Sign Language Recognition. In 2018 Innovations in Intelligent Systems and Applications Conference (ASYU), Adana, Turkey, pp. 1–4.
- Savur, C., and F. Şahin (2015) Yüzey EMG sinyalinin kullanan gerçek zamanlı amerikan işaret dili tanıma sistemi. In: IEEE 14. uluslararası makine öğrenimi ve uygulamaları konferansı (ICMLA), s. Miami, Florida, USA, 497–502
- Şeker, A., B. Diri, and H. H. Balık. 2017. Derin öğrenme yöntemleri ve uygulamaları hakkında bir inceleme. *Gazi Mühendislik Bilimleri Dergisi (GMBD)* 3 (3):47–64.4.
- Sharma, S., and K. Kumar. (2021). ASL-3DCNN: American sign language recognition technique using 3-D convolutional neural networks. In *Multimedia Tools and Applications* 80, 26319–26331.
- Smith, L. N. (2017). Cyclical learning rates for training neural networks. March, In 2017 IEEE Winter Conference on Applications of Computer Vision (WACV), pp. 464–72.
- Smith, S. L., P. J. Kindermans, C. Ying, and Q. V. Le. (2017). *Don’t decay the learning rate, increase the batch size*. Cornell University, *arXiv, arXiv preprint arXiv:1711.00489*.
- Strezoski, G., D. Stojanovski, I. Dimitrovski, and G. Madjarov. (2016). Hand gesture recognition using deep convolutional neural networks. In International conference on ICT innovations, pp. 49–58, Springer, Cham.
- Szegedy, C., W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. (2019). Going deeper with convolutions. 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, pp. 1–9, doi: [10.1109/CVPR.2015.7298594](https://doi.org/10.1109/CVPR.2015.7298594).

- Takase, T., S. Oyama, and M. Kurihara. 2018. Effective neural network training with adaptive learning rate based on training loss. *Neural Networks* 101:68–78.
- Tampubolon, H., C. L. Yang, A. S. Chan, H. Sutrisno, and K. L. Hua. 2019. Optimized capsnet for traffic jam speed prediction using mobile sensor data un-der urban swarming transportation. *Sensors* 19 (23):5277.
- Tan, Z. (2019). Derin Öğrenme Yardımıyla Araç Sınıflandırma. Master's thesis. Fırat Üniversitesi, Fen Bilimleri Enstitüsü
- Tao, W., Z. H. Lai, M. C. Leu, and Z. Yin. (2018). American sign language alphabet recognition using leap motion controller. In Proceedings of the 2018 Institute of Industrial and Systems Engineers Annual Conference (IISE 2018), Orlando, Florida, USA.
- Thiracitta, N., and H. Gunawan. 2021. SIBI Sign Language Recognition Using Convolutional Neural Network Combined with Transfer Learning and non-trainable Parameters. *Procedia Computer Science* 179:72–80.
- Toğaçar, M., and B. Ergen. 2019. Biyomedikal Görüntülerde Derin Öğrenme ile Mevcut Yöntemlerin Kıyaslanması. *Fırat Üniversitesi Mühendislik Bilimleri Dergisi* 31 (1):109–21.
- Turan, B. Genel ve Teknik İletişim. (2014). Accessed March 4, 2020. <http://web.bilecik.edu.tr/bulent-turan/files/2014/11/gti.pdf>.
- Unutmaz, B., A. C. Karaca, and M. K. Güll. (2019). Kinect İskeleti ve Evrişimsel Sinir Ağını Kullanarak Türk İşaret Dili Tanıma. 27th Signal Processing and Communications Applications Conference (SIU), Sivas, Turkey, pp. 1–4, doi: 10.1109/SIU.2019.8806380.
- Wen, S., S. Xiao, Y. Yang, Z. Yan, Z. Zeng, and T. Huang. 2019. Adjusting Learning Rate of Memristor-Based Multilayer Neural Networks via Fuzzy Method. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 38 (6):1084–94. doi:10.1109/TCAD.2018.2834436.
- Wu, J., Z. Tian, L. Sun, L. Estevez, and R. Jafari (2015). Real-time American sign language recognition using wrist-worn motion and surface EMG sensors. In 2015 IEEE 12th International Conference on Wearable and Implantable Body Sensor Networks (BSN) (pp. 1–6). Cambridge, MA, USA: IEEE.
- Xing, J., G. Fang, J. Zhong, and J. Li. (2019). Application of Face Recognition Based on CNN in Fatigue Driving Detection. In Proceedings of the 2019 International Conference on Artificial Intelligence and Advanced Manufacturing (pp. 1–5). New York, NY, USA.
- Yalçınkaya, Ö., A. Atvar, and P. Duygulu. (2016). işaret dilini tanıma uygulaması. In 24th Signal Processing and Communication Application Conference, 2016-Proceedings, Zonguldak, Turkey. pp. 801–04.
- Yokoyama, R., M. Shirasawa, and R. J. Pike. 2002. Visualizing topography by openness: A new application of image processing to digital elevation models. *Photogrammetric Engineering and Remote Sensing* 68 (3):257–66.
- Zhang, W., P. Tang, and L. Zhao. 2019. Remote sensing image scene classification using. *CNN-CapsNet. Remote Sensing* 11 (5):494.