Computer Modeling in Engineering & Sciences

Tech Science Press

DOI: 10.32604/cmes.2024.047649

REVIEW

A Survey on Chinese Sign Language Recognition: From Traditional Methods to Artificial Intelligence

Xianwei Jiang[1], Yanqiong Zhang[1,*], Juan Lei[1] and Yudong Zhang[2,3,*]

[1]School of Mathematics and Information Science, Nanjing Normal University of Special Education, Nanjing, 210038, China

[2]School of Computing and Mathematical Sciences, University of Leicester, Leicester, LE1 7RH, UK

[3]Department of Information Technology, Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, 21589, Saudi Arabia

*Corresponding Authors: Yanqiong Zhang. Email: zhangyanqiong@njts.edu.cn; Yudong Zhang. Email: yudongzhang@ieee.org

Received: 13 November 2023 Accepted: 27 February 2024 Published: 16 April 2024

ABSTRACT

Research on Chinese Sign Language (CSL) provides convenience and support for individuals with hearing impairments to communicate and integrate into society. This article reviews the relevant literature on Chinese Sign Language Recognition (CSLR) in the past 20 years. Hidden Markov Models (HMM), Support Vector Machines (SVM), and Dynamic Time Warping (DTW) were found to be the most commonly employed technologies among traditional identification methods. Benefiting from the rapid development of computer vision and artificial intelligence technology, Convolutional Neural Networks (CNN), 3D-CNN, YOLO, Capsule Network (CapsNet) and various deep neural networks have sprung up. Deep Neural Networks (DNNs) and their derived models are integral to modern artificial intelligence recognition methods. In addition, technologies that were widely used in the early days have also been integrated and applied to specific hybrid models and customized identification methods. Sign language data collection includes acquiring data from data gloves, data sensors (such as Kinect, Leap Motion, etc.), and high-definition photography. Meanwhile, facial expression recognition, complex background processing, and 3D sign language recognition have also attracted research interests among scholars. Due to the uniqueness and complexity of Chinese sign language, accuracy, robustness, real-time performance, and user independence are significant challenges for future sign language recognition research. Additionally, suitable datasets and evaluation criteria are also worth pursuing.

KEYWORDS

Chinese Sign Language Recognition; deep neural networks; artificial intelligence; transfer learning; hybrid network models

1 Introduction

Chinese Sign Language is a particular expression that has its own characteristics, cultural significance, and aesthetic value. On the one hand, this expression combines the pronunciation and meaning of Chinese to teach and express. On the other hand, it expresses the meaning of Chinese in the form of gestures, uses hand movements to publicize Chinese characteristics, and expresses cultural

aesthetics. Although the meticulous characters of Chinese have influenced the development of Chinese Sign Language, it still has its own characteristics and culture. It expresses the meaning of Chinese through quick gestures, forming an interesting way of expression. Hand movements can replace the writing of Chinese characters and are faster and more attention-grabbing. Though the expression of sign language cannot completely replace Chinese characters, it provides an effective oral expression in contemporary society and helps those who cannot read and write Chinese characters pass on their cultural knowledge orally. In addition, Chinese Sign Language has its own cultural references and aesthetics in the field of cognitive language, and it is widely employed in daily life. For example, in TV, movies, songs, music, and dramas, Chinese sign language is frequently applied to enhance the literary effect and increase the sense of art. In public places, it is sometimes seen that sign language helps deaf people communicate.

As a minority language, Chinese Sign Language has a long history. At present, the number of hearing-impaired people in China is close to 30 million, which is the largest number of disabled groups in China. Sign language is the main modus of communication for the hearing impaired. Barrier-free communication is a significant way for the majority of hearing-impaired people to break the island of limited information and carry out equal social communication. The main requirement for realizing barrier-free communication for the hearing-impaired is that the hearing person can understand the sign language expression of the hearing-impaired person. It is against this background that more and more scientists and scholars have begun to pay attention to Chinese Sign Language and study it and its various recognition technologies. The classification of Chinese Sign Language Recognition can be referred to as Fig. 1. In the past two decades, Chinese Sign Language Recognition technology has made rapid progress. More and more researchers have been focusing their research on Chinese Sign Language Recognition technology and developed technologies such as hand movement recognition, tone recognition, dynamic recognition, semantic recognition, etc. Additionally, in recent years, Chinese scholars have also begun to focus on intelligent speech synthesis and natural language processing technology and developed a series of Chinese Sign Language Recognition systems to improve the lives of the hearing-impaired. With the development of artificial intelligence technology, especially the progress of computer vision and natural language processing research, it is possible to realize this requirement. The study of sign language recognition and translation is a specific research task to realize the above needs.

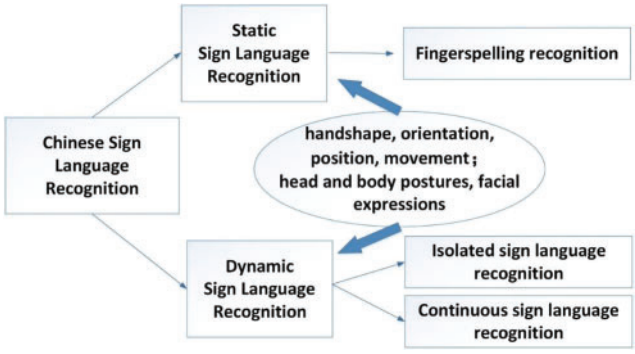

Figure 1: Classification of Chinese Sign Language Recognition

2 Literature Review

Sign language recognition can be defined as the process of describing and interpreting sequences of gestures using algorithms and technology to convert them into text or speech. The goal of sign language recognition is to translate sign language videos into corresponding sign language annotations automatically. Chinese Sign Language Recognition (CSLR) technology has gone through the process of going from traditional methods to modern deep learning. Below, we will look back at this procedure by conducting a literature review of the past 20 years.

In the first ten years, the focus of Chinese Sign Language Recognition was on sensor-based applications and systems. The mainstream technologies used in traditional classification methods were Hidden Markov Models (HMM) and Support Vector Machines (SVM). In comparison, the cost of recognition at this stage is relatively high, while the accuracy of recognition is relatively low. In the past ten years, there has been a proliferation of high-tech and new technologies for sign language recognition. Some representative research papers are presented below.

Yang et al. [1] proposed a gesture recognition method based on gesture principal direction and class-Hausdorff distance template matching. Firstly, the segmented gesture image was standardized, and the main direction of the gesture in the standardized image was obtained. Then, a two-dimensional gesture Cartesian coordinate system was established according to the main direction of the gesture to extract spatial gesture features. Then, the spatial gesture coordinate point distribution feature method was used to conduct preliminary recognition of gestures. Finally, the final gesture was recognized by using the idea of class-Hausdorff distance template matching. The experimental results showed that, under the condition of relatively stable illumination, the method could accurately realize gesture recognition in real-time, and the overall recognition rate reached 95%; the recognition rate for gestures with rotation could exceed 90%. A method for continuous sign language recognition based on the second-order Hidden Markov Model (HMM2) was proposed by Mei et al. [2]. In this method, the sliding window algorithm was used to divide the sign language video into multiple short sign language videos, and the feature vectors of the short sign language video and the sign language vocabulary video were obtained through a three-dimensional convolution model. By calculating the relevant parameters of the second-order hidden Markov model, they employed the Viterbi algorithm to realize the recognition of continuous sign language. Experiments proved that sign language recognition based on the second-order hidden Markov model achieved a recognition accuracy rate of 88.6%, which was higher than the traditional first-order hidden Markov model. Li et al. [3] combined the gray-level co-occurrence matrix and other multi-features to recognize CSL. SVM with a linear kernel function was employed for classification. The experiment was conducted on 30 groups of alphabet images of CSL and achieved 93.09% average accuracy. Zhang et al. [4] proposed a novel system with the dynamic time warping (DTW) algorithm for continuous sign language recognition. The system was evaluated with 180 sentences obtained from Kinect. The results indicated the effectiveness of the approach.

Yang et al. [5] proposed an attention-based continuous sign language recognition algorithm called ACN (Attention-based 3D convolutional neural network), which could recognize continuous sign language even in complex backgrounds. The algorithm used the background removal module to preprocess sign language videos containing complex backgrounds. Then, it extracted spatiotemporal fusion information using a 3D-ResNet that incorporates a spatial attention mechanism. Finally, a Long Short-Term Memory (LSTM) network was integrated to perform sequence learning and obtain recognition results. The algorithm achieved excellent performance on the CSL100 dataset. In the case of different complex backgrounds, the algorithm showed good generalization performance, and the spatiotemporal attention mechanism introduced by the model proved to be effective. Zhang et al. [6]

integrated the algorithm in OpenCV to propose a gesture recognition system using YOLO V3, which greatly improved the accuracy of recognition, ran fast, and was suitable for different scenarios. Experimental results showed that the system's gesture recognition accuracy was around 90%. It could complete barrier-free communication with deaf-mute people, the production cost was low, and the model was easy to transplant, which was suitable for popularization. Xie et al. [7] proposed a new model architecture, PiSLTRc, which was a position-informed sign language transformer with content-aware convolutions. Compared with the ordinary Transformer model, the model achieved superior performance on three large-scale sign language benchmarks. Jiang et al. [8] proposed an end-to-end continuous sign language recognition method based on Transformer, which achieved an accuracy of 96.30% on the CSL data set. A multimodal fusion framework (SeeSign) was proposed by Zhang et al. [9], in which multimodal features were input to a network based on Transformer. This model obtained an accuracy of 93.17%, 81.66%, and 77.92% on isolated words, one-handed and two-handed SL data sets, respectively.

3 Traditional SLR Modus and Approaches

Traditional sign language recognition can be roughly divided into the following four stages: obtaining gesture samples, preprocessing images (including segmentation and detection), feature extraction, and classification recognition. There are different approaches and technologies at each stage, which constitute different sign language recognition models and systems.

3.1 Data Collection

In the early stage of sign language data collection, hand modeling devices such as data gloves were employed to collect data. The hand shape, movement trajectory, and three-dimensional space position of the sign language demonstrator describe the process of sign language movement change. In the research of sign language recognition and translation based on visual features, the color image of the sign language demonstrator is obtained by the camera and processed accordingly, which is used as the input data for the simulation of sign language recognition. In addition, some other modal sign language information is also concerned [10], such as a somatosensory camera, to obtain visual image information, depth information, and skeleton information at the same time. In general, compared with non-vision-based acquisition methods, vision-based acquisition methods have the advantages of low cost, convenient acquisition, and low equipment dependence. Still, at the same time, they are more challenging in feature processing and algorithm modeling.

Sign language data sets can be roughly divided into isolated word sign language data sets and continuous sign language data sets. With the continuous development of sign language research techniques, the need for large-scale, multilingual sign language data sets is also increasing. At present, sign language research has involved the sign languages of Germany [11], China [12], the United States [13], Poland [14], Arabia [15], Italy [16], South Korea [17], Argentina [18] and nearly 30 other countries.

The list of sign language datasets from major countries is shown in Table 1. Among them, the USTC-CCSL dataset is currently the most widely used Chinese sign language dataset, which contains about 25,000 labeled sign languages demonstrated by 50 sign language demonstrators. The data set adopts a Kinect camera to collect data, which can provide RGB visual information, depth information, and skeleton information.

Table 1: List of sign language datasets

Dataset	Country	Number of samples	Data characteristics	Data type	Availability
RWTH-PHOENIX-Weather	Germany	45760	RGB	Sentence	Public
ChaLearn	America	50000	RGB/Deep	Word	Partially public
DGS Kinect 40	Germany	3000	Multiple points of view	Isolated words	
CSL	China	25000	Deep/Skeleton/RGB	Isolated Word/ Sentence	Public
SIGNUM	Germany	33210	RGB	Sentence	Public
GSL 20	Greece	840	RGB	Word	
Boston ASLLVD	America	9800	RGB	Word	Public
PSL Kinect 30	Poland	300	RGB/Deep	Word	Public
LSA64	Argentina	3200	RGB	Word	Public
DEVISGN-G	China	432	RGB	Word	
DEVISGN-D		6000			
DEVISGN-L		24000			
CUNY ASL	America		RGB	Sentence	
Signs World Atlas	Arab		RGB	Word	Public
ASL Fingerspelling	America	131000	RGB/Deep	Word	Public

3.2 Pre-Processing

In image segmentation and image recognition data processing, it is usually necessary to preprocess the dataset image before training the model. The advantage of this is to avoid the influence of solid interference factors, such as noise in the image, on the final training results, accuracy, and processing time. Grayscale conversion, smooth filtering, normalization, noise reduction, and various morphological transformations are commonly utilized in image preprocessing. In the study of sign language recognition, the input image size is usually adjusted, the resolution is reduced, and the feature regions are extracted before and after segmentation to reduce the computational load and improve the computational efficiency.

3.3 Detection and Segmentation

Sign language detection aims to detect hand information in images and position information in space. Segmentation is to divide the sign language image into regions of interest and other regions and separate the regions of interest from the image. There are typically two types of segmentation methods, namely context-dependent and context-independent. Context-sensitive segmentation considers the spatial relationship between features, such as edge detection technology. Context-free does not consider spatial relations but groups pixels based on global attributes. The rise of deep learning brings new opportunities to sign language segmentation. After massive data training, the model completes the corresponding sign language segmentation, making the segmentation more convenient and having good application prospects in sign language segmentation. However, there are also some

shortcomings. Some networks have complex hierarchical structures, slow segmentation speed, fuzzy edge information, and edge detection accuracy needs to be improved.

3.4 Feature Extraction

It is called feature extraction to transform the interested part of the input data into a feature set. After the hand segmentation and tracking, the feature information in the image needs to be extracted. Features include not only temporal information but also spatial information. The features in dynamic sign language recognition can be divided into local features, global features, and fusion features. Local feature mainly extracts local feature points with obvious changes in image sequence, mainly including corners, interest points, etc., to find corresponding points and regions in the image. Global features extract features based on depth images, including texture, shape, etc., to obtain the representation information of images. Fusion features mainly include global features and local *features*.

3.4.1 Gray Level Co-Occurrence Matrix

Repeated changes in grayscale distribution in spatial position form the texture. Therefore, there must be some grayscale relationship between any two pixels in the image space, which is known as grayscale spatial correlation. The gray level co-occurrence matrix (GLCM) is a method used to describe texture by analyzing the spatial correlation characteristics of gray levels. This method was first introduced by Haralick et al. [19] in 1973. The gray-level co-occurrence matrix processing of sign language images is shown in Fig. 2.

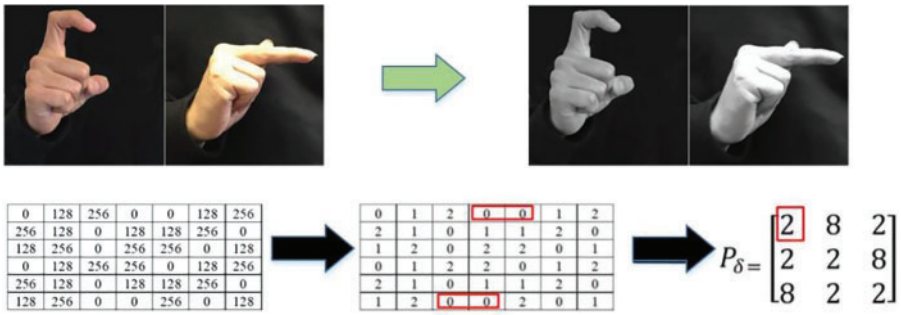

Figure 2: Gray-level co-occurrence matrix processing of sign language images

3.4.2 Histogram of Oriented Gradients

Oriented gradient histogram (HOG) is a feature description method widely used in computer vision and image processing [20]. Including object orientation, HOG is invariant for geometric and photometric conversion. HOG is particularly suitable for human body detection in images. As shown in Fig. 3, the main flow of HOG algorithm implementation was provided. Mahmud et al. [21] employed HOG to feature extraction and utilized k-Nearest Neighbor (KNN) to classify American sign language. This method provides superior accuracy (94.23%) to the compared approach (86%).

3.4.3 Wavelet Entropy

The energy distribution of wavelet packet coefficients can be employed to analyze the characteristics of EMG (electromyogram) signals, combining information entropy to analyze their uncertainty and complexity [22]. According to the EMG wavelet packet transform, the wavelet packet coefficient matrix can be extracted, and wavelet packet entropy can be calculated. Then, an eigenvector

constructed with EMG signal wavelet packet entropy can be adopted to classify the hand actions. Wavelet packet function has the characteristic of frequency domain localization, which can provide the function's orthogonality and each function's orthogonality based on time axis translation. Wavelet packet decomposition is a natural extension of wavelet transform. It can decompose the signal into a subspace of equal bandwidth in a binary tree way. When the signal is decomposed into N layers, the whole signal space is decomposed into $2N$ subspace. The signal of the n-TH subspace can be reconstructed. In Fig. 4, a process of 2-level two-dimensional discrete wavelet transform was provided.

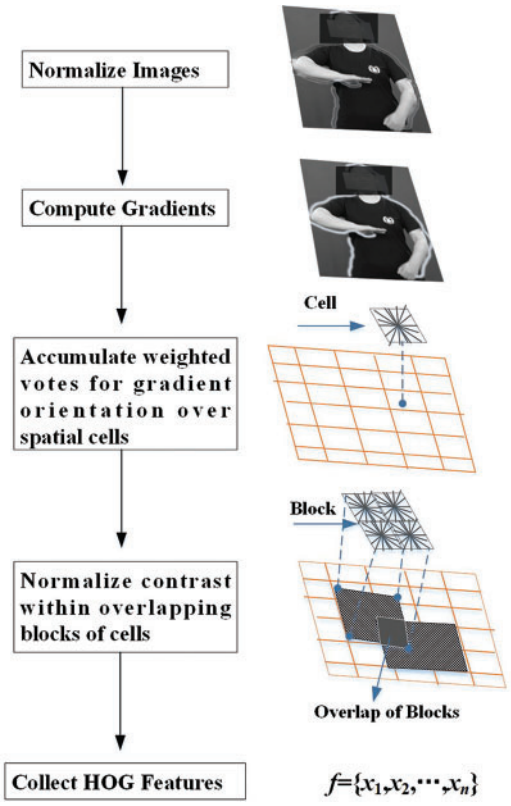

Figure 3: Main flow of HOG algorithm implementation

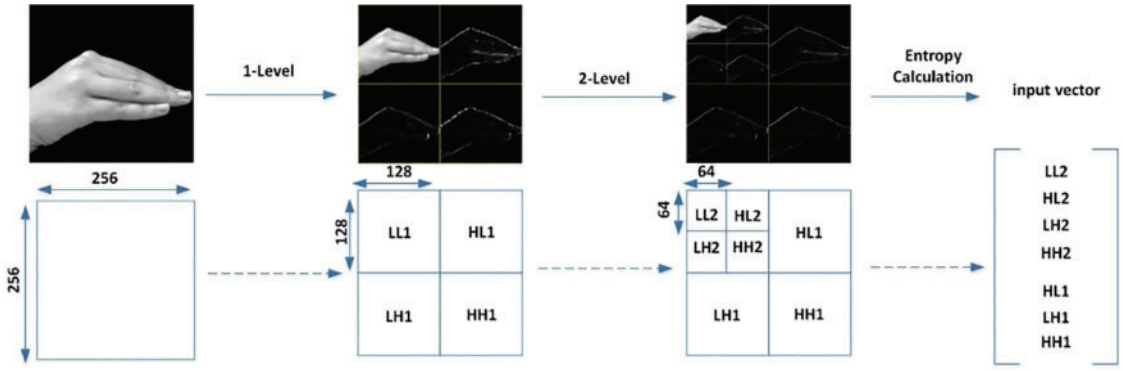

Figure 4: A process of 2-level two-dimension discrete wavelet transform

The energy distribution probability is also called relative wavelet packet energy. It reflects the distribution of the signal in various frequency bands. In information theory, entropy provides a measure of the amount of information contained in various probability distributions.

Wavelet entropy (WE) can quantitatively measure the order and disorder of information distribution and reflect some useful information qualitatively. If the energy of the EMG signal is all concentrated in a sub-band, then WE = 0 indicates that the EMG is ordered. On the other hand, if the energy is randomly distributed among the sub-bands, WE is large, which is a sign of disorder. Zhu et al. [23] proposed a WE-RBF method for Chinese fingerspelling identification and achieved an overall accuracy of 88.76%.

3.4.4 Principal Component Analysis

Principal component analysis (PCA) is also known as the Karhunen-Loeve Transform, and its transformation essence is a method to approximate a vector or image by using a low-dimensional subspace [24]. This method usually uses the minimum mean square error criterion (MSE) to obtain the optimal subspace. Its advantage is that it can effectively reduce the dimension of the original feature vector on the basis of fully retaining useful information, so it has been widely used in the field of biometric recognition technology [25]. Gweth et al. [26] combined PCA and neural network features to construct an automatic SLR system. They improved the word error rate of the best-published results on the SIGNUM database by more than 6%.

3.4.5 Other Feature Extraction Approaches

The following feature extraction methods are often mentioned. For instance, Scale Invariant Feature Transform (SIFT) [27–29] can always be employed to extract the features of the sign language image as the sign language visual vocabulary in the image. In addition, Hu moment invariant (HMI) [30], Fourier descriptors (FD) [31,32], Speeded Up Robust Features (SURF) [33], and Latent Dirichlet Allocation (LDA) [34], etc., also appear frequently in some papers.

3.5 Classification

For a long time in the past, researchers have been trying to achieve effective sign language recognition through traditional machine learning methods, which integrate functional modules such as “body detection, body tracking, feature extraction, classifier”. In theory, sign language recognition uses data to train a model so that input information can be processed by detection, tracking, and feature extraction modules to obtain features representing sign language differences, and then these extracted features are connected to a classifier. In order to obtain specific features from the data to explain the meaning of sign language, most research methods rely on manual definition and feature selection. In terms of classifiers, the following models are commonly employed in machine learning models.

3.5.1 Hidden Markov Model

The hidden Markov model can be regarded as a concrete example of a state space model in which potential variables are discrete. However, if we look at a single time slice of the model, we see that it corresponds to a mixed probability distribution, and the corresponding component density is $p(r|z)$. Therefore, it can also be expressed as a generalization of the mixed probability model, in which the mixing coefficient of each observation is not selected independently but depends on the selection of the

component of the previous observation. HMM is widely used in speech recognition, natural language modeling, online handwriting recognition, and analysis of biological sequences [35].

As in the case of the standard mixed model, the potential variable is the discrete variable zn subject to polynomial distribution, which describes the mixed component used to generate corresponding observations n. As before, it is convenient to use the 1-of-K representation method. We now let the probability distribution of z_n depend on the previous potential variable z_{n-1} through the conditional probability distribution $p(z_n, k = 1|z_{n-1}, j = 1)$. Since the potential variable is a K-dimensional binary variable, the conditional probability distribution corresponds to a table composed of numbers recorded as A, and its elements are called transition probabilities. The element is A_{jk}, equaling to $p(z_n, k = 1|z_{n-1}, j = 1)$. Because they are probability values, they satisfy $0 \leq A_{jk} \leq 1$ and $\sum_k A_{jk} = 1$, so matrix A has $K(K-1)$ independent parameters. In this way, we can explicitly write the conditional probability distribution as

$$p(z_{n-1}|z_n, A) = \prod_{k=1}^{k} \prod_{j=1}^{k} A_{jk}^{z_{n-1}, jz_{nk}} \tag{1}$$

The initial potential node z_1 is very special because it has no parent node, so its edge probability distribution $p(z_1)$ is represented by a probability vector π, and the element is π_k equal to $p(z_{1k} = 1)$, that is

$$p(z_1|\pi) = \prod_{k=1}^{k} \pi_k^{z1k} \tag{2}$$

where $\sum_k \pi_k = 1$.

A probability model can be determined by defining the conditional probability distribution $p(x_n|Z_n, \varnothing)$ of the observation variable, where $\varnothing$ is the parameter set that controls the probability distribution. These conditional probabilities are called emission probabilities. They can be Gaussian distributions or conditional probability tables. For a given value of $\varnothing$, the probability distribution $p(x_n|z_n)$ is composed of a K-dimensional vector, corresponding to K possible states of the binary vector z_n. We can express the launch probability as

$$p(x_n|Z_n, \varnothing) = \prod_{k=1}^{k} p(xn|\varnothing k)^{Z_{nk}} \tag{3}$$

A strong property of the HMM is that it is invariant to local deformation (compression and extension) on the time axis to some extent. In speech recognition problems, the deformation of the time axis is related to the natural variation in the speed of speech. Hidden Markov models can adapt to this deformation without exerting too much influence. A framework utilized by HMM was proposed by Zhang et al. [36], fusing trajectories and hand shape features. The approach was evaluated effectively on the self-building dataset. Gao et al. [37] proposed a CSLR system employing SOFM-HMM. In comparison to the existing system, the performance indicated superiority, and the word recognition rate reached 82.9% on the dataset containing 5113 samples.

3.5.2 Support Vector Machine

A support vector machine is proposed based on statistical learning theory and structural risk minimization criteria. Under such a background, support vector machine technology has a strong generalization and discrimination ability [38]. The focus is on finding the optimal classification hyperplane for input data samples. The corresponding problem can be solved by using a quadratic function that maximizes the classification interval of data samples. As shown in Fig. 5, based on two types of linearly separable data, circles and diamonds are employed to represent each type of data,

respectively. The margin represents the maximum classification interval between the classification planes, and the data points on both sides of the classification line are the samples to be classified. In this example, the equation of the basic classification surface is shown in the formula.

$$w^T x + b = 0 \tag{4}$$

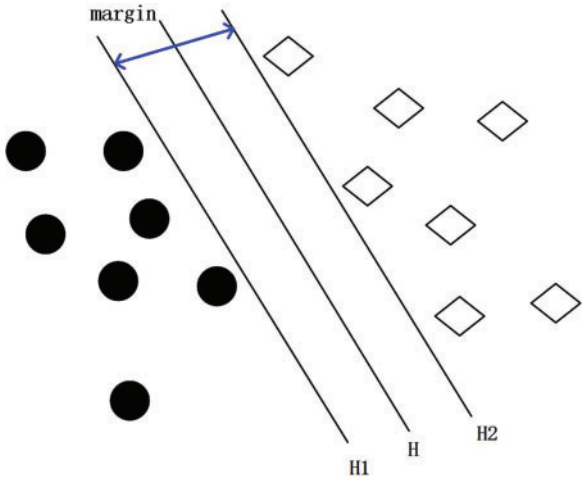

Figure 5: An example of the optimal classification line

From the above, combined with effective constraints and the introduction of Lagrange multipliers, the optimal classification discriminant function can be solved. The kernel function is often commonly combined with the optimal classification discriminant surface to create a support vector machine model. The corresponding general support vector machine classification function expression is shown as follows:

$$f(x) = \text{sgn}\left\{\sum_{i=1}^{k} a_i^* y_i K(x_i \cdot x) + b^*\right\} \tag{5}$$

where a_i^* and b^* are parameters that regulate the support vector machine to determine the optimal classification plane.

Combined with HMM and SVM, a multilayer architecture classifier was proposed by Ye et al. [39], which was considered effective for Chinese Sign Language Recognition with a large vocabulary. Pu et al. [40] studied automatic sign language recognition and introduced SVM to classification. The results show the approach is effective on the sign language dataset, including more than 500 words.

3.5.3 Dynamic Time Warping

Dynamic gesture recognition methods commonly employ the DTW algorithm and HMM. The HMM algorithm requires a large amount of gesture data for template training. After multiple training calculations, appropriate model parameters can be obtained. The DTW algorithm does not require additional training and is simple, fast, and easy to implement [41].

DTW algorithm adopts point-by-point matching to calculate the cumulative distance and uses dynamic programming to find the optimal path. As shown in Fig. 6, the DTW algorithm consists of two main steps. One is to calculate the distance matrix between each point in the two sequences. The second task is to find a path from the lower left corner to the upper right corner of the matrix,

ensuring that the sum of the elements on the path is the smallest. Assuming the matrix is M, the shortest path length from the lower left corner of the matrix $(1, 1)$ to any point (i, j) is denoted as $L_{min}(i, j)$. We can use a recursive algorithm to find the shortest path length. The recursion rules are as follows: $L_{min}(i, j) = \min\{L_{min}(i, j-1), L_{min}(i-1, j), L_{min}(i-1, j-1)\} + M(i, j)$. Among them, the initial conditions are as follows: $L_{min}(1, 1) = M(1, 1)$.

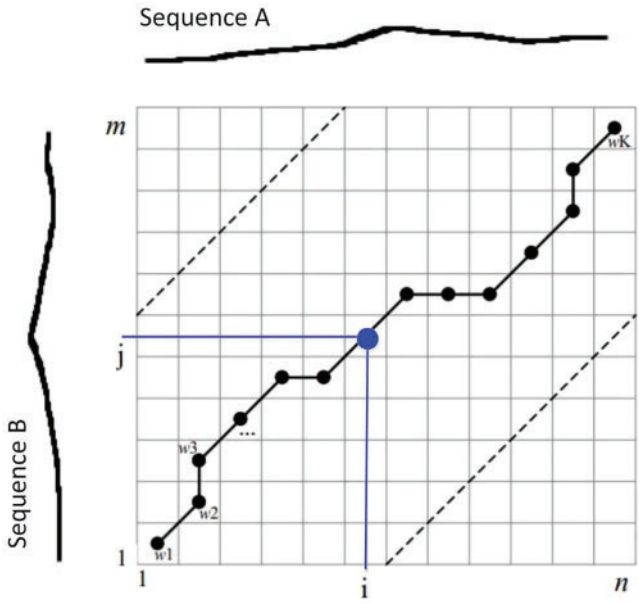

Figure 6: Calculation method of warp path distance

However, when using the DTW algorithm for matching calculations, the upper and lower boundaries will be calculated every time the grid point being searched moves forward by one grid. The amount of computation is still significant, especially when two matching sequences are long, leading to more repetitive operations [42].

A novel system by Zhang et al. [4] was designed for continuous sign language recognition, adopting the DTW algorithm. The experiments were conducted on 180 sentences and demonstrated effective superiority.

3.5.4 Random Forest

The random forest (RF) strategy combines the Bagging ensemble, constructed by decision tree learners, with selecting random attributes during the training process. This algorithm is relatively simple, has low computational overhead, and performs well in many real-world tasks [43].

Random forest classification can be viewed as a complex of multiple decision tree classification models. The basic idea is as follows: firstly, k samples are extracted from the original training set using bootstrap sampling, and the sample size of each sample remains unchanged. Then, k decision tree models are established to obtain k classification results. Finally, based on k classification results, each record is voted on to determine its final classification. The schematic diagram is shown in Fig. 7.

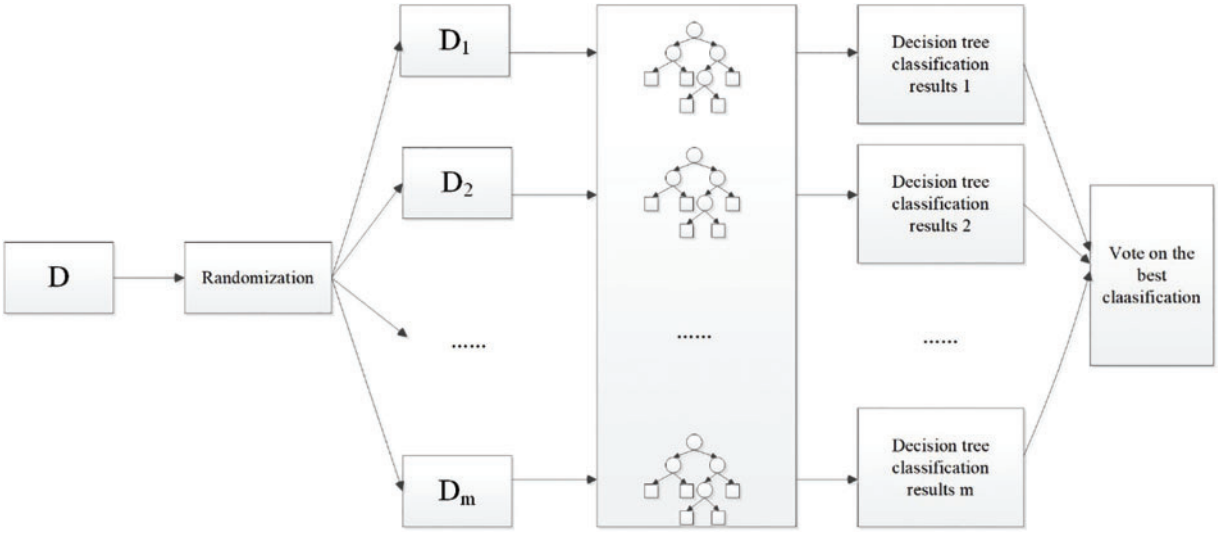

Figure 7: Schematic diagram of random forest classification

Simply put, Random Forest consists of multiple decision trees and is a comprehensive learning model. When a new classification is initiated, specific attributes of the object are chosen as the standard. All decision trees in RF will make their own decision, and then "vote" collectively. The classification output of RF is determined by the decision tree with the highest number of votes.

Yuan et al. [44] employed sEMG and an RF algorithm to identify 30 alphabets, achieving an average accuracy of 95.48%. Su et al. [45] utilized the RF algorithm to implement SLR systems based on ACC-sEMG. The proposed approach obtained an average accuracy of 98.25% in classifying 121 CSL subwords.

3.5.5 Long Short-Term Memory

RNN has succeeded in speech recognition, machine translation, computer vision, and other fields. One of its significant advantages is that it can process inputs of different lengths and effectively extract temporal features between frames. As an improvement of RNN, it can be seen in Fig. 8 that LSTM [46] adds a processor to judge whether the information is useful, so LSTM is widely used in timing classification. LSTM can detect temporal changes in sign language and learn the corresponding relationships between gesture changes, thereby enhancing the classification of sign language [47]. Some sign language actions take a long time to recognize, so many researchers use the LSTM network to predict the next sign language action.

Liu et al. [48] applied an LSTM-based SLR method to evaluate isolated CSL vocabularies. Experiments indicated that the proposed approach was effective. Additionally, Yang et al. [49] combined CNN with LSTM to recognize 40 daily vocabularies and achieved a high recognition rate. Xiao et al. [50] proposed a multimodal fusion method (LSTM2-DHMM) to identify CSL. This framework was evaluated on two CSL data sets and obtained effectiveness.

3.5.6 Other Classification Approaches

There are other taxonomies that are often mentioned, such as artificial neural networks (ANNs) [51], Nave Bayes classifier (NBC) [52], Relevance Vector Machine (RVM) [53], etc.

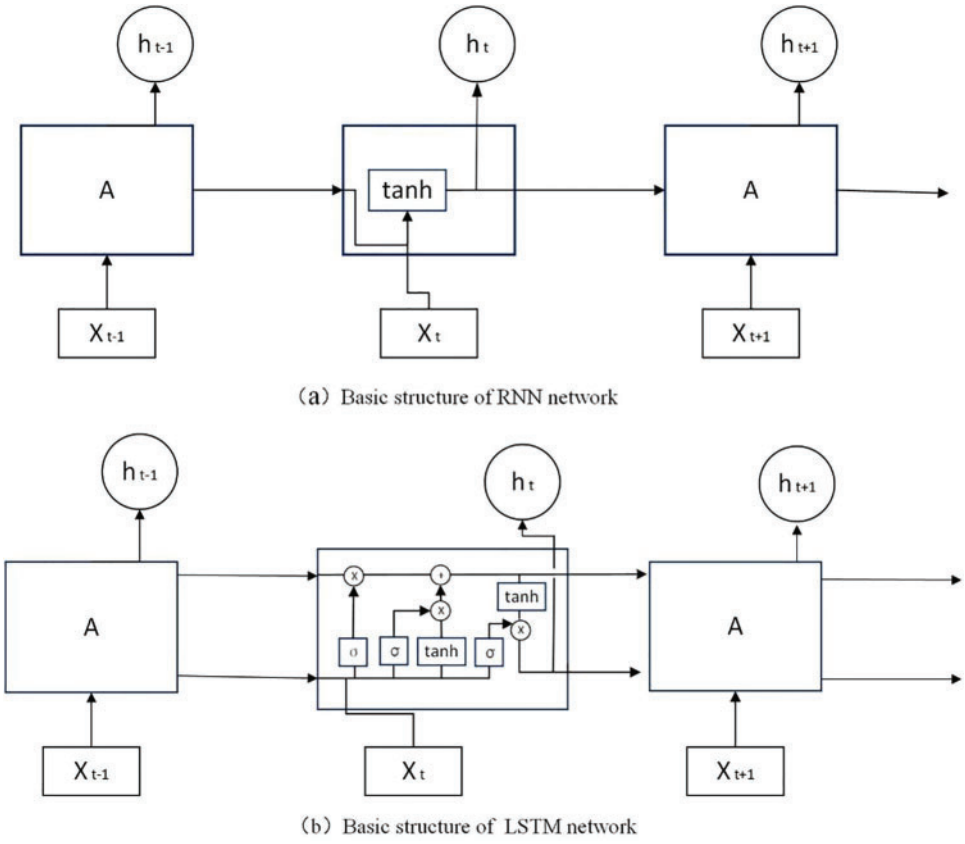

Figure 8: Basic structure of RNN and LSTM networks

4 Modern Sign Language Recognition Modes and Techniques

Artificial intelligence refers to the capability to simulate, extend, and enhance human intelligence through intelligent systems. These systems can perform tasks that necessitate human intelligence, such as learning, reasoning, perception, interaction, problem-solving, and understanding natural language. Artificial intelligence encompasses various fields, including machine learning, deep learning, expert systems, natural language processing, and computer vision. Artificial intelligence has its origins in the 20th century. In 1943, McCullocah and others [54] proposed that propositional logic could be used to explain neural events and the relationships between them, which is considered the origin of artificial neural networks (ANN). However, due to the perceptron's inability to solve nonlinear problems, research on artificial neural networks subsequently declined. In 1982, the proposal of recursive artificial neural networks reignited research interest, leading to the emergence of deep learning in the public consciousness. Deep learning was proposed by Professor Geoffrey Hinton in 1985. However, the computing power at that time was extremely limited, making it very difficult to execute deep learning. But starting from 2010 to 2012, deep learning began to gain popularity in the field of artificial intelligence. At present, computing power, bandwidth, and storage space have improved by millions of times, enabling the widespread realization of the concept of deep learning. Today's vast data and high-speed processing capabilities also empower our previous speech recognition and image recognition algorithms to execute a large number of calculations in a very short time, leading to improved results.

Traditional methods for sign language recognition have offered some solutions, but as the demands for sign language recognition increase, previous methods can no longer meet the new requirements. Therefore, new technologies and methods have become new areas of focus.

1. The ultimate goal of sign language recognition is to achieve continuous recognition and establish an efficient system for recognizing sign language. Video-based continuous sign language recognition aims to transcribe sign language videos into a series of annotations. Traditional methods of sign language recognition play a lackluster role. The CSLR model based on deep learning consists of three components: a vision module, a sequence (context) module, and an alignment module. It occupies a dominant position because of its superiority over traditional methods.
2. There are specific criteria for evaluating the naturalness and authenticity of sign language recognition and translation, including the presence of a "deaf flavor" and the incorporation of emotional factors. By fusing multi-modal information and combining lips and facial expressions, deep neural networks can support and help achieve this goal.
3. The development of artificial intelligence (AI) technology has led to the promotion and application of AI sign language digital humans in specific sign language interpretation scenarios. This represents an important advancement in modern intelligent sign language recognition methods.

Therefore, the recognition method for Chinese Sign Language is transitioning from traditional methods to modern AI-based approaches. Deep learning, transfer learning, and hybrid network models based on deep neural networks offer new and improved solutions for sign language recognition.

4.1 Convolutional Neural Networks

A convolutional neural network (CNN) refers to a feed-forward neural network with a convolutional computing function and a deep structure. Due to its superior feature extraction ability and accurate classification ability of image information, it is considered to be the most representative deep neural network for recognition and classification [55–58]. A typical convolutional neural network consists of several layers, including an input, convolutional, pooling, fully connected, and output layers. Fig. 9 shows a simple CNN graph. Among them, the convolution layer performs feature extraction through convolution operations. The fully connected layer is equivalent to a "classifier".

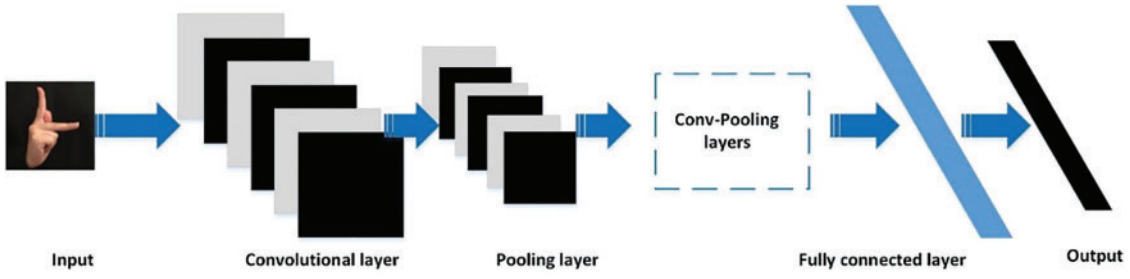

Figure 9: A simple CNN diagram

However, the performance of big data-driven deep learning models increases as the number of samples increases. Therefore, this also places greater requirements on sample size and network training. Simple CNN does not achieve better performance. Therefore, various optimization algorithms have been incorporated into the convolutional neural network model, and the performance

has been continuously improved. For example, batch normalization (BN) techniques can keep the inputs of layers more evenly distributed. Dropout technology can refine the network, effectively reduce overfitting, and achieve a certain degree of regularization. The ReLU function can accelerate the convergence of stochastic gradient descent [59]. Data augmentation (DA) technology can effectively expand the dataset and help alleviate overfitting [60].

CNN is typically utilized for processing array-like data. Its components include corresponding parts in ANN and also feature pooling and flattening functions, which can reduce the dimensionality of features extracted by CNN blocks. CNN and its variants are widely used in various types of Chinese Sign Language Recognition, including fingerspelling recognition, isolated sign language recognition, and continuous sign language recognition. In view of the important position of CNN in deep learning networks, researchers have conducted a series of studies on CNN-based sign language recognition since 2013. For example, literature [61] proposed a CNN network focusing on hand shape changes. It feeds hand shape features into an end-to-end weakly supervised classification framework for accurate recognition. This system is capable of real-time recognition of small-scale isolated word sign language datasets. For another example, in the context of continuous sign language recognition, literature [62] utilized an adaptive video sampling method to effectively preprocess the video to remove interference from irrelevant backgrounds. After using CNN to extract the features of the video frame, the BLSTM model is employed to learn the bidirectional dependency information of the sequence in order to model the spatio-temporal sequence. Finally, the recognition result is obtained using the CTC algorithm.

4.2 3D-CNN

Although CNN has a robust feature extraction ability, it is limited to processing single-frame image data. Sign language recognition also requires auxiliary methods to mine inter-frame correlation, and the 3D convolutional neural network (3D-CNN) has emerged as a solution. 3D-CNN mainly solves the correlation between pictures and adds a new dimension of information. Discriminative features from both spatial and temporal dimensions can be captured by 3D-CNN [63–65].

The essence of 2D convolution is to extract local neighborhood features from the feature map of the previous layer and obtain a 2D feature map by convolution in the spatial dimension. The convolution process can be expressed as follows:

$$v_{ij}^{xy} = tanh\left(b_{ij} + \sum_m \sum_{p=0}^{P_{i-1}} \sum_{q=0}^{Q_{i-1}} w_{ijm}^{pq} v_{(i-1)m}^{(x+p)(y+q)}\right) \tag{6}$$

where v_{ij}^{xy} indicates the value of pixel (x, y) of the j^{th} feature map in layer i, and b_{ij} denotes the deviation of the j^{th} feature map in layer i, m denotes the number of feature maps in the $i-1$ layer, P_i and Q_i indicate the spatial dimension size of 2D convolution kernel in the i layer, and w_{ijm}^{pq} denotes the weight of the convolution kernel connected by the m^{th} feature map in the $i-1$ layer.

In the video analysis problem of sign language, the motion information data to be acquired are in multiple consecutive frames, so 2D convolution is expanded to 3D convolution, and features are calculated from spatial and temporal dimensions.

Multiple continuous frames pass through the convolutional layer sequentially. Each feature map is connected to multiple adjacent continuous frames in the previous layer in order to obtain specific motion information [66], which can be expressed as:

$$v_{ij}^{xyz} = tanh\left(b_{ij} + \sum_m \sum_{p=0}^{P_{i-1}} \sum_{q=0}^{Q_{i-1}} \sum_{r=0}^{R_{i-1}} w_{ijm}^{pq} v_{(i-1)m}^{(x+p)(y+q)(z+r)}\right) \tag{7}$$

where v_{ij}^{xyz} denotes the result of pixel (x, y, z) of the j^{th} feature map in layer i, and b_{ij} indicates the deviation of the j^{th} feature map in layer i, m denotes the number of feature maps in the $i-1$ layer, P_i, Q_i and R_i are the spatial dimension size of the 2D convolution kernel in the i layer, and w_{ijm}^{pq} is the weight of the convolution kernel connected by the m^{th} feature map in the $i-1$ layer. Compared with 2D convolution, 3D convolution adds the time dimension, and its frame structure is shown in Fig. 10.

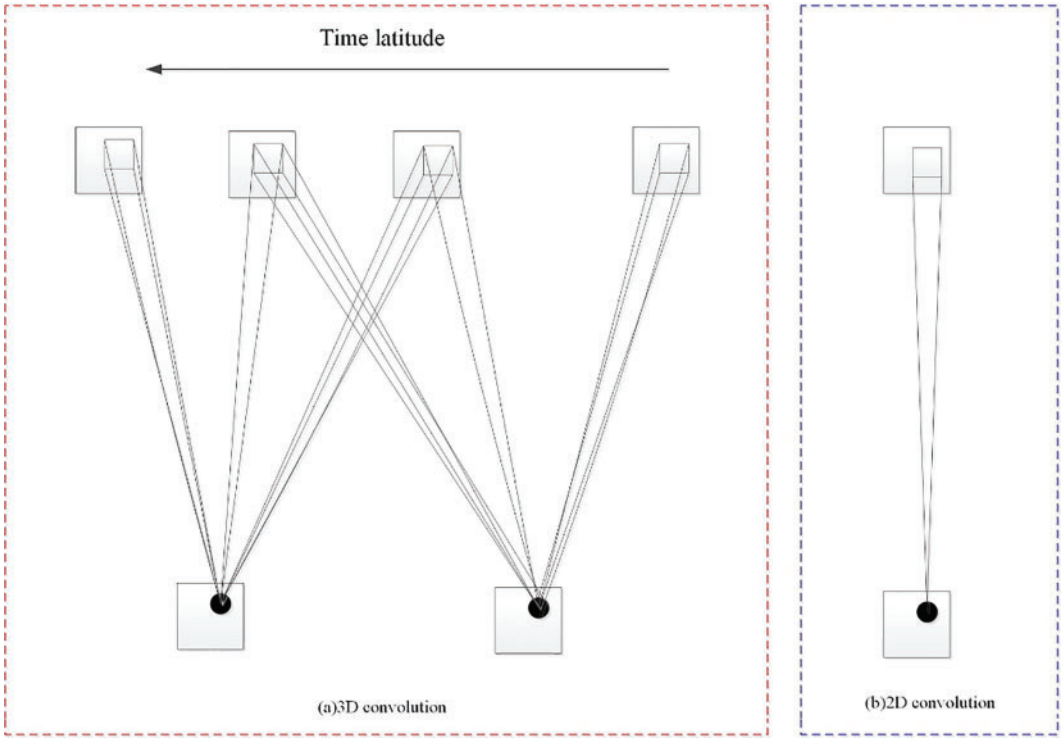

Figure 10: 3D convolution and 2D convolution frame structures

The network structure of 3D-CNN mainly consists of the following:

i) 3D convolution layer: 3D convolution is implemented by convolving a 3D kernel into a temporal cube formed by stacking multiple consecutive frames together. With this structure, the feature maps are connected to the previous multiple consecutive frames, and motion information is captured.

Among them, the 3D convolution calculation formula is as follows:

$$I_{xyt} = f\left(\sum_{i=0}^{n}\sum_{j=0}^{n}\sum_{k=0}^{m} w_{ijk} v_{(x+i)(y+j)(t+k)} + b\right) \tag{8}$$

Among them, $f(\cdot)$ indicates the neural activation function, such as Tanh, Sigmoid, or Relu, etc., I_{xyt} denotes the feature map value, w_{ijk} denotes the (i, j, k)th value of the kernel connected to the previous feature map, and m denotes the size of the 3D kernel along the time dimension, $v_{(x+i)(y+j)(t+k)}$ indicates the input unit at position $(x+i, y+j, t+k)$, and b is the deviation of the graph.

ii) The related technologies and calculation formulas used in the batch normalization, ReLU, and pooling layers are the same as those used in two-dimensional CNN.

As early as the CVPR2015 conference, Molchanov et al. of the NVIDIA Research Institute first proposed using 3D-CNN for dynamic gesture recognition [67]. A bidirectional sub-network is

constructed using multi-scale data as the network input to extract spatio-temporal feature sets of gestures. The model has achieved good recognition results in autonomous driving scenarios. 3D-CNN and its combined model can be applied to isolated sign language recognition and continuous sign language recognition in Chinese sign language. Pu et al. first employed 3D-CNN for Chinese Sign Language Recognition in 2016 [40]. A sign language recognition algorithm based on a 3D-CNN network based on multimodal data was proposed by Liang et al. [68]. They performed convolutional fusion on various data and verified its effectiveness on a large-scale dataset.

4.3 YOLO

(You Only Look Once) YOLO is one of the well-known models in the field of computer vision. Unlike other classification methods, this approach combines the task into a regression problem, eliminating the need to separate the detection results into two categories (classification and regression). Although the accuracy is slightly reduced, it detects much faster and is suitable for real-time object detection [69,70].

The development of YOLO has gone through several stages, from YOLO V1 to YOLO V8. The YOLO V1 algorithm divides each image into a grid system of size S ∗ S. Each grid identifies objects by predicting the number of bounding boxes of objects within the grid. It scans the entire image using a multi-scale sliding window to identify various objects in an image and determine their locations. It is crucial to determine the optimal size and number of sliding windows, as varying the number of candidates or including irrelevant candidates will yield different results. Among them, B bounding boxes will be predicted for each grid. Including its own position, each bounding box also predicts a confidence value, which represents the confidence level of the object contained within the predicted box and the accuracy of the prediction for this box. The calculation formula is as follows:

$$\text{Confidence} = \Pr(\text{Object}) * IoU_{pred}^{truth} \tag{9}$$

Finally, during testing, the class information predicted by each grid is multiplied by the confidence information predicted by the bounding box, and the class-specific confidence score of each box is obtained. The formula is as follows:

$$\Pr(Class_i|\text{Object}) * \Pr(\text{Object}) * IoU_{pred}^{truth} = \Pr(Class_i) * IoU_{pred}^{truth} \tag{10}$$

Then, by setting a threshold, filtering out low-scoring boxes, and performing NMS processing on the retained boxes, the final inspection result is obtained. In this way, YOLO is comparable to traditional algorithms but much faster. The advantage of YOLO V1 is that it can detect objects in real-time at high speed, understand generalized object representation, and the model is not overly complex. A limitation of YOLO V1 is that the model is less effective when small objects appear in clusters or groups.

Compared to YOLO V1, YOLO V2 has made various improvements in terms of speed, accuracy, and the ability to detect a large number of objects. Softmax is used in the YOLO V2 architecture to assign an objectivity score to each bounding box. BN (Batch Normalization) operations have been added in V2. The BN layer performs standardization and normalization on the input from the previous layer, scaling the input values. Additionally, higher-resolution inputs are utilized. All of these improvements enhance accuracy.

In the new structure of predicting boundaries, YOLO V3 adds logistic regression to predict the score of each bounding box. The Faster R-CNN method was also introduced, and only one bounding box was given priority. These small improvements lead to big improvements.

YOLO V4 achieves a superior and more efficient model by incorporating and integrating new features. An important theme that YOLO V4 focuses on is real-time object detection using traditional neural network models. These models only require traditional GPU training, making it possible to train, test, and implement convincing object detection models.

YOLO V5 is a single-stage target detection algorithm that makes the following improvements based on YOLO V4. The model training phase introduces Mosaic data augmentation and adaptive anchor frame calculation. The baseline network incorporates ideas from other detection algorithms, such as Focus and CSP structure. FPN and PAN structures are added between the "BackBone" and the final "Head" output layer. The DIOU_nms of the prediction box screening and training loss function are improved in the "Head" output layer. Therefore, YOLO V5 has apparent advantages, namely, the framework structure is user-friendly, convenient for training data sets, and easy to put into production; it integrates a large number of computer vision technologies, easy to configure the environment, and has fast training speed; batch inference produces real-time results. Object recognition speeds are impressive.

The model of YOLO V5 target detection studied by Li et al. [71] is in line with the life-scenes and is well-suited for real-time applications. The YOLO method was mentioned in the isolated word recognition of Chinese Sign Language. By fusing the attention mechanism, Zhang et al. [72] improved the YOLO V5 model, recognizing over 40 daily CSL and achieving an mAP of 98.92%. The proposed CSL detection model is easy to apply on mobile devices and valuable for communicating with the hearing impaired.

4.4 CapsNet

Convolutional neural networks (CNN) have achieved great success in the field of image processing, but they also have certain limitations. For instance, it ignores the relative positions between different features and cannot identify poses, textures, and image changes. In addition, the pooling operation in CNN makes the model spatially invariant, so the model is not equivariant. At the end of 2017, Geoffrey Hinton et al. introduced capsule architecture in their "Dynamic Routing between Capsules" paper. This is a new deep neural network model that is currently primarily used in the field of image recognition. In deep learning, a capsule refers to a group of embedded neurons. A capsule network (CapsNet) comprises capsules instead of neurons [73,74].

Artificial neurons output a single scalar quantity. Each convolution kernel in the CNN convolutional layer copies the weight of the same kernel to the entire input image and outputs a two-dimensional matrix. Each number in the matrix is the convolution of a part of the input image with the convolution kernel. This two-dimensional matrix can be regarded as the output of the repeating feature detector. The two-dimensional matrices of all convolution kernels are stacked together to obtain the output of the convolution layer. CNN utilizes max pooling to achieve invariance, but max pooling discards valuable information and lacks a relative spatial relationship with the encoded features.

Different from traditional neurons, the input and output of a Capsule are both vectors. The vector length denotes the probability in traditional neurons, while the vector direction represents other information, including position information. The Capsule network utilizes dynamic routing based on an agreement to replace Max-Pooling in traditional CNN, which can also be understood as an original routing mechanism. The capsule encodes the probability of feature detection as its output vector length and the detected feature state as the vector direction. When the detected feature changes, the probability remains the same, but its direction changes.

A comparison of capsules and neurons is presented in Table 2.

Table 2: Comparison between Capsule and traditional neuron

	Input	Operation				Output
		Affine transform	Weighting	Sum	Nonlinear activation	
Capsule	$vector\,(u_i)$	$\hat{u}_{j\|i} = W_{ij}u_i$	$s_j = \sum_i c_{ij}\hat{u}_{j\|i}$		$V_j = \frac{\|s_j\|^2}{1+\|s_j\|^2}\frac{s_j}{\|s_j\|}$	$vector\,(v_j)$
Traditional neuron	$scalar\,(x_i)$	–	$a_j = \sum_i w_i x_i + b$		$h_j = f\,(a_j)$	$vector\,(h_j)$

The Capsule Network consists of six neural network layers, including a convolution layer, a PrimaryCaps layer, a DigitCaps layer, the first fully connected layer, the second fully connected layer, and the third fully connected layer. The first three layers are encoders, and the last three layers are decoders. The structure of the capsule network is shown in Fig. 11.

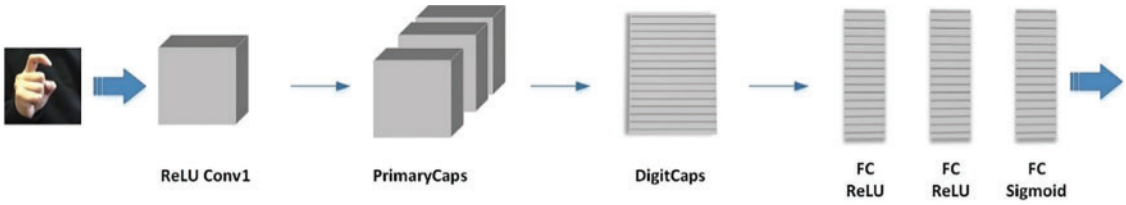

Figure 11: The structure of the capsule network

CapsNet is a new concept in deep learning that produces good results compared to CNNs and traditional neural networks. CNN classifiers are not robust against noisy data; however, CapsNets are more resilient to such data and can also adapt to affine transformations of the input data. At the same time, capsule networks have also been proven to reduce training time and minimize the number of parameters. It can solve tasks such as machine translation, autonomous driving, handwritten character and text recognition, target detection, and emotion detection, etc. CapsNet has been frequently mentioned in the context of continuous sign language recognition. Suri et al. [75] developed a novel IMU-CapsNet architecture for recognizing continuous Indian Sign Language. The method yielded an accuracy of 94% and 92.50% for three routings and five routings, respectively, which achieved higher Nash equilibrium.

4.5 Transformer

At present, BERT [76] and GPT [77] models have achieved great success. The Transformer [78] structure has replaced RNN and CNN, which has become the standard configuration for current NLP models. The internals of the Transformer are essentially an "Encoder-Decoder" structure. As shown in Fig. 12, the entire network structure is entirely composed of the "Attention mechanism" and adopts a 6-layer "Encoder-Decoder" structure. The encoder is responsible for mapping the natural language sequence into a hidden layer. The decoder remaps the hidden layer into a natural language sequence, allowing us to solve various problems, such as machine translation, summary generation, semantic relationship extraction, sentiment analysis, etc.

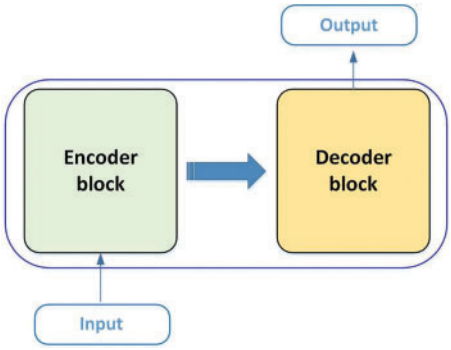

Figure 12: The overall structure of the Transformer

The workflow of the transformer is roughly described as follows:

Step 1: Obtain the representation vector X of each word of the input sentence. Where X is obtained by adding the Embedding of the word and the Embedding of the word position.

Step 2: Pass the obtained word representation vector matrix into the Encoder. The encoding information matrix C of all words in the sentence can be obtained through six Encoder blocks. The word vector matrix is represented by $X(n \times d)$, n indicates the number of words in the sentence, and d indicates the dimension of the vector. The matrix dimensions output by each Encoder block are exactly the same as the input.

Step 3: Pass the encoding information matrix C output by the Encoder to the Decoder. The Decoder will translate the next $word_{i+1}$ based on the currently translated $word_k$, $(k = 1 \sim i)$. Among them, when translating to $word_{i+1}$, the words after $word_i$ need to be covered by the Mask operation.

Compared with RNN, the Transformer can be trained in parallel better. However, it cannot utilize the order information of words, so positional embedding needs to be added to the input. The relevant calculation formula is as follows:

$$PE\,(Pos, 2k) = \sin\left(\frac{pos}{10000^{\frac{2k}{d}}}\right) \tag{11}$$

$$PE\,(Pos, 2k+1) = \cos\left(\frac{pos}{10000^{\frac{2k}{d}}}\right) \tag{12}$$

where Pos denotes the absolute position of the word in the sentence, d indicates the dimension of the word vector, and k denotes the sequence value of the dimension in the word vector.

The focus of Transformer is the self-attention structure. The result is shown in Fig. 13, in which the matrices Q, K, V are obtained by linear transformation of the output. Then, the output of Self-Attention can be calculated. The calculation formula is as follows:

$$Attention\,(Q, K, V) = softmax\left(\frac{qk^T}{\sqrt{d_k}}\right)V \tag{13}$$

where d_k is the number of columns of the matrix, which is the dimension of the vector.

In the transformer, multiple Self-Attentions in Multi-Head Attention can capture the correlation coefficient attention score in multiple dimensions between words [79]. The transformer and its derived models can be utilized for isolated and continuous sign language recognition in Chinese Sign Language. Du et al. [80] integrated a vision transformer and a temporal transformer to construct

a self-attention framework, which was utilized to recognize word-level sign language. Experimental results demonstrate its superiority on the WLASL dataset. Cui et al. [81] proposed a Spatial-Temporal Transformer Network (STTN) for continuous sign language recognition (CSLR). The STTN was evaluated on two datasets: CSL and PHOENIX-2014. The results indicated the superior effectiveness of the CSLR task.

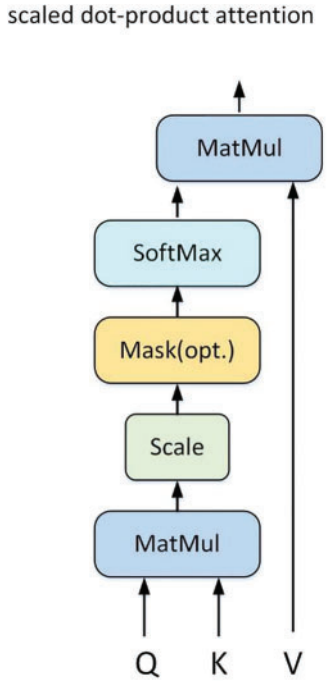

Figure 13: Self attention structure

4.6 Transfer Learning and Hybrid Network Model

Transfer learning refers to transferring the learned and trained model parameters to a new model to help the new model train. Transfer learning is different from traditional machine learning. Traditional machine learning builds different models for different learning tasks, whereas transfer learning utilizes data from the source domain to transfer knowledge to the target domain to complete model establishment. Since there are correlations between most data or tasks, the existing model parameters can be shared with the new model in some way, which is known as knowledge transfer. Transfer learning speeds up and optimizes the learning efficiency of the model without having to learn from scratch [82,83].

Transfer learning is defined by a domain and a task, and its mathematical representation is as follows. A domain $\mathcal{D}$ consists of feature space $\mathcal{X}$ and marginal probability distribution $P(X)$, where $X = \{x_1, \ldots, x_n\} \in \mathcal{X}$. Assume that the given domain $\mathcal{D} = \{\mathcal{X}, P(X)\}$, the task consists of two parts: the label space $\mathcal{Y}$ and the target prediction function $f: \mathcal{X} \to \mathcal{Y}$. The function f predicts the label $f(x)$ corresponding to x. Task $\mathcal{T} = \{\mathcal{Y}, f(x)\}$ is learned from training data containing sample pairs $\{x_i, y_i\}$, where $x_i \in \mathcal{X}$, $y_i \in \mathcal{Y}$. Given the original domain $\mathcal{D}_S$ and its task $\mathcal{T}_S$, the target domain $\mathcal{D}_T$ and its task $\mathcal{T}_T$, the following conditions are met: $\mathcal{D}_S \neq \mathcal{D}_T$ *or* $\mathcal{T}_S \neq \mathcal{T}_T$. Transfer learning aims to learn the target prediction function $f_T(\cdot)$ in the $\mathcal{D}_T$ domain by utilizing the knowledge of $\mathcal{D}_S$ and $\mathcal{T}_S$ [84].

There are three types of transfer in transfer learning: instance-based transfer, feature-based transfer, and shared parameter-based transfer [85]. Instance-based transfer learning focuses on selecting examples useful for training in the target domain from the source domain. For instance, effective weight distribution can be performed on labeled data instances from the source domain so that the instance distribution in the source domain is close to the instance distribution in the target domain, thereby establishing a reliable learning model with high classification accuracy in the target domain. However, since the data distribution in both the source domain and the target domain are often inconsistent, all labeled data instances in the source domain may not necessarily be useful to the target domain. Feature-based transfer includes transfer learning based on feature selection and transfer learning based on feature mapping. The former focuses on finding common feature representations between the source domain and the target domain, and the latter focuses on mapping the data of the source domain and the target domain from the original feature space to a new feature space. Since the data distribution is the same in the source and target domain spaces, feature-based transfer can better utilize existing labeled data samples for classification training and testing. Transfer learning, based on shared parameters, investigates the common parameters or prior distributions between two spatial models of source and target data.

As shown in Fig. 14, there are two strategies for applying transfer learning. One strategy is fine-tuning, which involves using a pre-trained network on a base dataset and training all layers on the target dataset. During pre-training, the model will likely be exposed to datasets similar to the task. Fine-tuning can stimulate the knowledge acquired by the model during the pre-training process. The other is to freeze and retrain, which involves freezing all layers except the last layer (the weights are not updated) and training only the last layer. Transfer learning is not limited to deep learning, but there are indeed many applications in deep learning. A novel CSLR approach that utilized transfer learning based on AlexNet was designed in the paper [86], which combined the Adam optimizer and provided four special configurations. The highest accuracy of 91.48% was yielded for the identification of Chinese fingerspelling.

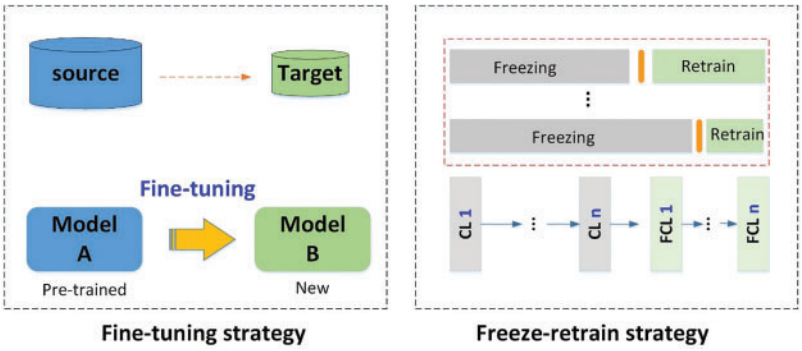

Figure 14: The strategy of transfer learning

The original intention of transfer learning is to transfer knowledge and representation between two tasks or domains to improve performance. Deep transfer learning has become an important method for effective knowledge transfer in recent years. Common network models include AlexNet, VGG, GoogLeNet, ResNet, etc. What they have in common is that they all utilize deep neural networks to accomplish transfer between tasks or domains.

Various advanced network models and technologies continue to emerge with the continuous development of artificial intelligence and neural network technology. In most cases, a variety of

mainstream technologies and advanced methods are often combined to achieve efficient network models. Integrating multiple technologies can compensate for the limitations of individual technologies and enhance overall performance. The integration and innovation of these technologies and methods make image recognition more practical and advanced. For instance, the CNN-based Chinese sign language recognition in [59] and [60] achieved an average accuracy of 88.10% and 89.32%, respectively. The accuracy of CSLR adopting only the LSTM method in [87] is 86.20%. The accuracy of sign language recognition has been improved to 98.11% [88] and 98.40% [89] through the fusion of these two technologies (CNN-LSTM). Additionally, some specialized networks, such as CGNet [90] and GFNet [91], have been developed for image recognition and detection and have achieved effectiveness. Therefore, the hybrid model offers new ideas and additional solutions for recognizing Chinese sign language.

5 Analysis and Discussions

We reviewed the relevant literature on Chinese Sign Language Recognition (CSLR) in the past 20 years. It was found that HMM, SVM, and DTW are the most widely employed techniques among traditional recognition methods. Deep neural networks (DNN) and their derived models are essential to modern artificial intelligence recognition methods. Meanwhile, there are also hybrid models and experimenter-defined identification methods.

5.1 Analysis of CSLR

Fig. 15 shows that the number of research papers on Chinese Sign Language Recognition has exhibited a consistent upward trend. It was in a slow growth stage before 2012, and research papers on sign language recognition have significantly increased since 2013. Especially since 2014, the publication of literature has grown exponentially, primarily due to the rapid advancements in computer vision and artificial intelligence technology. During the same period, Chinese Sign Language Recognition transitioned from traditional research methods to new methods and technologies based on vision, particularly deep neural networks. This trend has been confirmed more clearly since 2019.

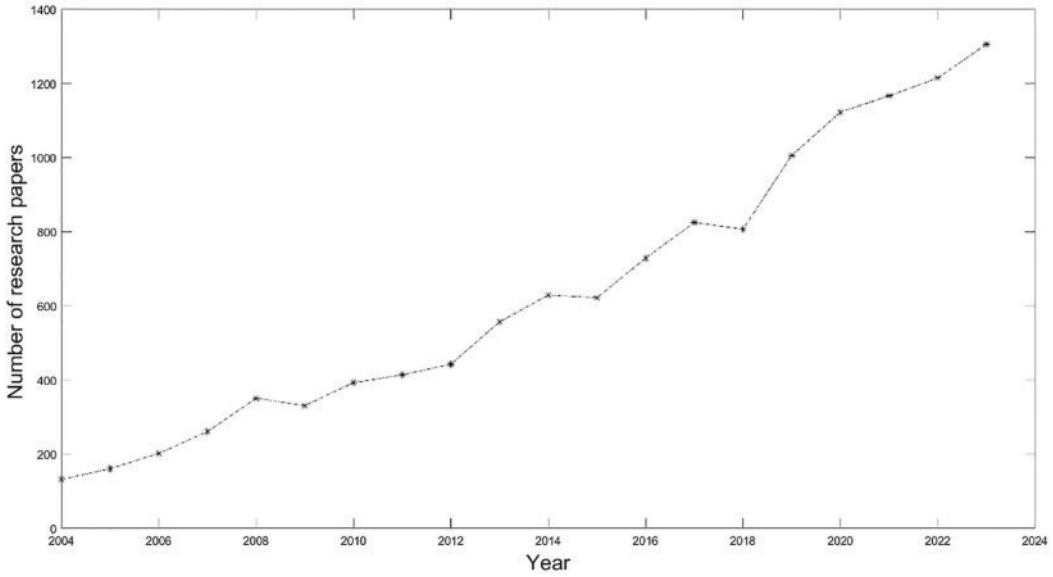

Figure 15: Research trend of CSLR with major technologies

As shown in Table 3, Chinese Sign Language Recognition technologies and methods can be divided into two stages: sign language recognition based on traditional technology and sign language recognition based on modern artificial intelligence technology. In the first stage (approximately from 2000 to 2011), HMM, SVM, and DTW were the mainstream technologies. Among them, HMM, widely used in speech recognition and handwritten font recognition, has been introduced into the field of sign language recognition. It is mainly used for time series modeling of sign language and has achieved good results. During this period, research on sign language recognition mainly focused on recognizing fingerspelling sign language and isolated static sign language (gesture). The datasets were obtained from data gloves. In the second stage (approximately from 2012 to the present), CNN, 3D-CNN, YOLO, and various deep neural networks (such as ResNet, VGG-Nets, Faster R-CNN, CapsNet, etc.) have sprung up. At this stage, research on sign language recognition mainly focuses on large-scale sign language and real-time, continuous sign language recognition. The datasets are obtained from data sensors with higher data collection quality, such as Kinect and Leap Motion, as well as high-definition photography. At the same time, facial expression recognition, complex background processing, and 3D sign language recognition have also attracted the research interest of scholars. In addition, technologies such as HMM and SVM, widely utilized in the early stages, have also been integrated and applied to some hybrid models. Overall, these two stages can be seen as the transformation of sign language recognition from traditional technology to computer vision and artificial intelligence and from a single to a hybrid model.

Table 3: Summary of Chinese Sign Language Recognition

Publication year	Typical methods and techniques	Characteristics	Performance evaluation/ Accuracy	Datasets/Sample size
2004	TMD-HMM + PCA [92]	Vision-based sign language recognition system	92.50%	CSL words (439)
2004	SOFM-HMM + SRN [37]	Chinese Sign Language Recognition system	82.90%	Sign vocabulary (5113)
2005	Boosted CHMM [93]	Recognition of sign language subwords	92.70%	Custom sample (510)
2005	NBC [52]	NBC	Over 80%	10 gestures
2006	DTW-HMMs + Re-sampling [94]	Expanding training set	85.35%	Gestures (2435)
2006	DTW-HMMs + TMMs [95]	Large-vocabulary continuous sign language recognition	91.90%	Test sentences (1500)
2008	SVM + Fourier descriptor-Hu moments [31]	Vision-based multi-features classifier	95.03%	Chinese manual alphabet (30)
2009	SVM + Co-Occurrence Matrix [3]	Gray-level co-occurrence matrix and other multi-features fusion	93.09%	Samples (5850)
2010	HMMs + ACC, sEMG [96]	Portable Accelerometer and EMG Sensors	95.78%	CSL subwords (121)
2011	HMMs + ACC, sEMG [97]	A framework for hand gesture recognition	95.30% (ACC) 96.30% (EMG)	CSL words (72)

(Continued)

Table 3 (continued)

Publication year	Typical methods and techniques	Characteristics	Performance evaluation/ Accuracy	Datasets/Sample size
2012	HMMs + ACC, sEMG [98]	A sign-component-based framework	96.50% 86.70%	Signs (120) Sentences (200)
2012	NN + Hu moment	Combining Hu moment feature and NN classifier	98.00%	High frequency words (201)
2013	I2C-DTW	Image-to-Class Dynamic Warping	98.44%	UESTC-DGL
2013	DICamShift + SLVW	Depth image camShift	96.21%	Chinese manual alphabet (30)
2013	Kinect + SURF	Speeded up robust features	97.70%	Chinese manual alphabet
2014	WDTW	Windowed dynamic time warping	85.00%	Gesture
2014	SVM + SURF [99]	Depth image information and SURF-BoW	96.24%	Sign language alphabet letters (30)
2014	ELM + SPC + hand shape [100]	3D Hand motion trajectories and depth images	82.79%	Instances (320)
2014	LC-KSVD + Hand trajectories and HOG	RGB-D sensor with sparse coding	92.36%	Chinese sign words (34)
2015	Camshift + HMM	Depth pre-segmentation combined with Camshift tracking	97.70%	Number “0” to “9”
2015	MEMS	Micro electro-mechanical systems	87.30%	Gesture
2015	fHMM	Framing hidden markov model	97.50 ± 1.60%	Chinese sign language words (30)
2015	Light-HMMs + HOG, RGB-D [101]	Key frame + Light-HMM	84.20%	Signs (1000)
2015	Multi-SVM DTW + HOD [4]	HOD + multi-SVM; DTW	85.20%	Phrases (450)
2016	Hausdorff distance template matching [1]	Hausdorff distance template matching	95.00%	Gestrure
2016	Hu + DTW	Hu + DTW	better recognition effect	Number “0” to “9”
2016	HMMs + HOG, PCA	HOG + PCA, HMMs framework	86.00%	Sign words (500)
2016	SVM-VHMM + HOD, RDF, HOG	SVM + VHMM + HOD	89.40%	Signs (500)
2016	SVM + HOG [102]	HOG + SVM based on Kinect	89.80%	Words (72)
2016	RF + ACC, sEMG [45]	Random forest; accelerometers and surface electromyographic sensors	98.25%	CSL subwords (121)
2016	End-to-end LSTM [87]	LSTM	86.20%	Isolated Chinese sign language vocabulary
2017	Statistical template matching	Skin color segmentation; statistical template matching	93.50%	Gesture (11)
2017	Tree-structure + sEMG, ACC, GYRO	sEMG + ACC + GYRO; optimized tree-structure framework	87.02%	Chinese sign language subwords (150)

(Continued)

Table 3 (continued)

Publication year	Typical methods and techniques	Characteristics	Performance evaluation/ Accuracy	Datasets/Sample size
2017	CNN + hand shape segmentation [103]	Convolutional neural network	99.00%	vocabularies (40)
2018	SURF + HMM	Rapid robust features and hidden markov model	93.00%	Gesture (8)
2018	CNN + SVM	CNN + SVM	98.60%	Jochen Triesch
2018	SVM + DNN [104]	Gesture recognition and facial expression recognition	90.70%	Static sign language
2018	3D-CNNs [105]	3D CNN for dynamic sign language recognition	89.20%	Vocabularies (20)
2018	Keyframe-centered clips (KCCs) [106]	Keyframe-centered clips	89.16%∼91.18%	Chinese sign language words (310)
2018	Attention-based 3D-CNNs [107]	3D-CNN	88.70%	500 categories
2019	YOLO V3	YOLO V3, K-Means	94.00%	Gesture (4)
2019	SAE-(HOG + LBP)-SVMs	SAE-(HOG + LBP)-SVMs	96.67%	JTD
2019	CNN + LSTM	Convolutional neural networks, long short-term memory	99.26%	Special video
2019	B3D ResNet [108]	BLSTM-3D residual networks	86.90%	Vocabularies (500)
2019	ANN [109]	ANN	88.70%	CSL gestures (15)
2019	3D-CNNs [110]	Attention-Based 3D-CNNs for large-vocabulary sign language recognition	88.70%	Categories CSL (500)
2019	6-layer CNN [59]	CNN; fingerspelling	88.10 ± 1.48%	Samples (1260)
2019	8-Layer CNN [60]	CNN; fingerspelling	89.32 ± 1.07%	Samples (1320)
2019	DBN [111]	Deep belief net, sEMG, ACC, GYRO	95.10%	150 CSL subwords
2020	CNN + BiLSTM [88]	Convolutional neural network and bidirectional long short-term memory	98.11%	Gesture (9)
2020	K-means + DTW [112]	improved K-means clustering pruning, DTW	90.00 ± 2.03%	Gesture (128)
2020	HMI-RBF-SVM [30],	HMI, RBF, SVM	86.47 ± 1.15%	Chinese fingerspelling
2020	RF-sEMG [44]	RF, sEMG	95.48%	30 alphabe
2021	HPSO-SVM	Hybrid particle swarm optimization, support vector machine	96.78%	Gesture (5)
2021	SSW	Sliding window segmentation	83.90%	Sentences (30)
2021	WE-RBF [23]	WE, RBF	88.76%	Chinese fingerspelling
2022	CNN [113]	CNN	99.50%	CSL
2022	HMM2 + Viterbi [2]	Second-order hidden markov model	88.60%	Sign video
2022	YOLO V5 [72]	YOLO V5	98.92%.	40 daily CSL
2022	3D-CNN [114]	3D-MobileNetv2	95.12%	CSL-500
2023	ACN + 3D-ResNet + LSTM	ACN, 3D-ResNet, LSTM	effective	CSL100
2023	Faster R-CNN	Faster R-CNN	85.00%	Gesture
2023	CNN + BLSTM [89]	CNN, BLSTM, CTC	98.40%	CSL
2023	YOLO V5	YOLO V5, labelimg	93.00%	Sign language pinyin

(Continued)

Table 3 (continued)

Publication year	Typical methods and techniques	Characteristics	Performance evaluation/ Accuracy	Datasets/Sample size
2023	Transfer learning	sEMG, IMU	85.10%	Sign language samples (60000)
2023	BLSTM [115]	Spatial-temporal graph attention network	98.41%	Chinese sign language dataset
2023	Transformer [8]	Transformer	96.30%	CSL
2023	Transformer [9]	Multimodal fusion framework (SeeSign)	93.17%	Isolated words

As shown in Fig. 16, Chinese Sign Language Recognition techniques and methods can be divided into three broad categories: vision-based, non-visual, and hybrid modes. In 2017, Yang et al. [103] proposed a vision-based sign language recognition method using a convolutional neural network and hand segmentation to verify 40 sign language vocabulary words and achieved a high recognition rate of 99.00%. In 2016, Su et al. [45] proposed a non-visual sign language recognition method based on ACC and sEMG, using random forest for analysis. The recognition rate was 98.25%, and the effect was also excellent. In 2018, Song et al. [104] used a hybrid model of SVM and DNN to recognize gestures and facial expressions. The recognition rate achieved was approximately 90.70%. At the same time, research on the frontier keywords of sign language recognition technology has found that CNN has the highest emergence intensity, followed by deep learning and machine learning. Since 2019, many scholars have used CNN technology to research sign language recognition. As shown in Fig. 17, sign language recognition technology is constantly updated and improved along with the rapid development of computer vision and artificial intelligence technology.

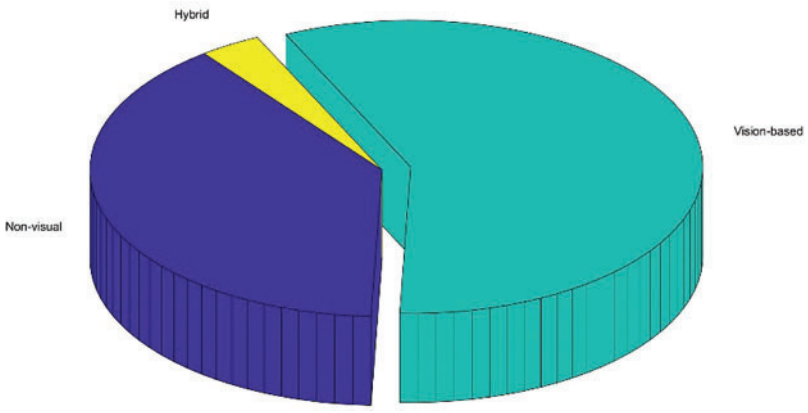

Figure 16: Major methods of classification and feature extraction in CSLR

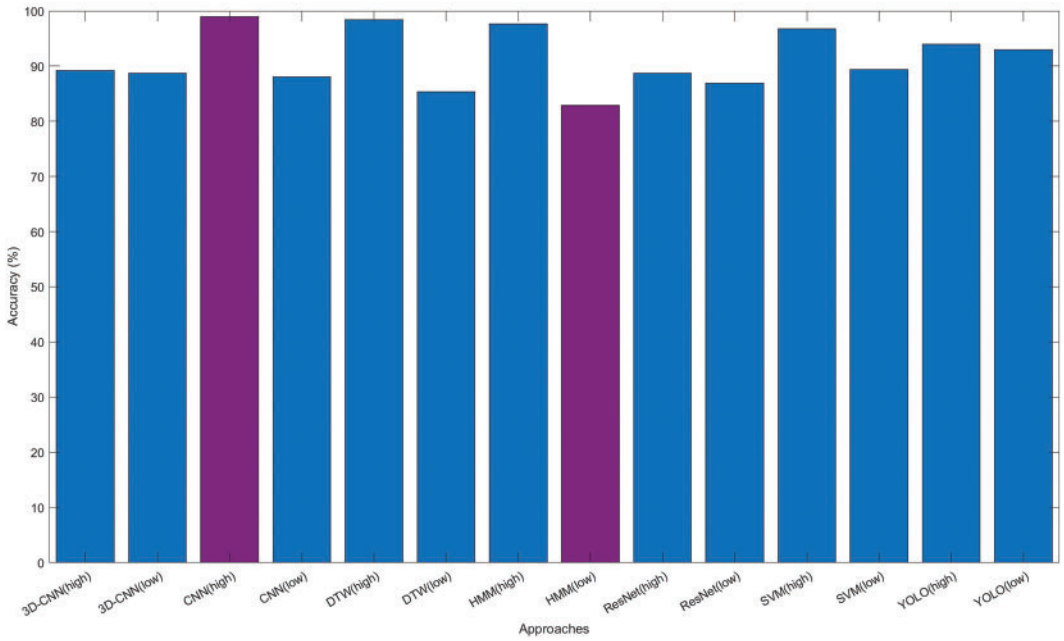

Figure 17: Comparison of accuracy with major methods in CSLR

5.2 Discussions of CSLR

Analysis indicated that various technologies are suitable for recognizing sign language based on different categories and characteristics. For instance, HMM and SVM are frequently employed to recognize finger language. Because the content of finger language is limited and belongs to static image recognition, the background environment is relatively controllable. Thus, the recognition accuracy is high, almost exceeding 90%. Additionally, CNN, 3DCNN, RNN, variants (such as LSTM, GRU, etc.), and Transformer models, are more suitable for continuous sign language recognition. Due to the temporal dynamics and contextual information involved in continuous sign languages, the relative recognition accuracy is lower and more challenging. At the same time, there is a growing need for handling massive data sets and powerful algorithm loads. Isolated word recognition falls between finger language recognition and continuous sign language recognition. Traditional and modern intelligence methods are mentioned, and the recognition performance is relatively satisfactory.

When it comes to sign language datasets, each one has its own unique characteristics and purpose for being created. In other words, the current compilation of resources for sign language data sets is primarily tailored to specific research requirements and utilizes customized specifications. As a result, they do not have relatively unified standards and cannot be easily generalized and promoted. They are limited to specific local applications. Based on various sign language classification methods, the dataset's characteristics, advantages, and disadvantages also vary. The finger language dataset allows for controlled data collection and can better accommodate factors such as background and lighting effects. As the dataset size increases, the accuracy of identifying isolated word data sets decreases. Continuous sign language recognition lacks large-scale and diverse data sets, which significantly impacts the practical requirements for real-time and online recognition. In addition, the continuous sign language dataset also needs to consider sentence segmentation and grammar, and include relevant supplementary information. It is evident that the recognition accuracy, data set size, creation cost, and cost-effectiveness corresponding to each data set are different. Therefore, the absence of appropriate corpora and datasets impedes further in-depth exploration of sign language research.

Sign language data collection can be categorized into two methods: contact and non-contact. Contact equipment was previously used for gesture recognition, with typical examples including data gloves [116–118], myoelectric signal armbands, inertial measurement units (IMU) [119], WiFi [120], radar, smartphones, Leap Motion controllers [121], and Kinect [122,123], etc. The equipment can directly detect the spatial information of the human hand and each joint and process it into input data. In contactless collection, the camera is the primary tool for acquiring input data and is used to capture sign language images and videos. The non-contact method has a low acquisition cost, minimal device dependence, and convenient acquisition. However, extracting features from video streams and keyframes may result in additional computational overhead. At the same time, the accuracy of vision-based recognition may decrease due to factors such as skin color, angle, and light. These problems typically require the use of high-performance computers.

Therefore, some suggestions and strategies for addressing CSLR challenges are as follows:

(1) Establish high-quality data sets and provide evaluation criteria. Most Chinese Sign Language data sets are too small, have insufficient samples, lack standardization, and cannot be generalized or compared horizontally. Additionally, they have a high proportion of experimental studies, which makes them unsuitable for application and promotion. Therefore, expanding the sample size, establishing a standardized and appropriate dataset in relevant fields, and addressing the challenges related to the shortage of evaluation resources and database standards is necessary.

(2) Develop efficient and accurate recognition systems by integrating multiple modalities and leveraging multi-perspective technologies. It is necessary to address the challenges of real-time processing, robustness, high accuracy, and user independence in sign language recognition. Address the issue of decreasing recognition accuracy as the dataset expands. In addition, a real-time system should be developed to properly handle changes in hand shapes against complex backgrounds and address the challenges of background interference, lighting, angle, and standardization of operations affecting sign language behavior. Meanwhile, in order to improve the accuracy of interpretations, it is necessary to incorporate sign language recognition that supplements the fusion of continuous sign language features with coordination information from lips and facial expressions. This will establish a comprehensive recognition model that includes main features as well as auxiliary information. Unfortunately, current sign language recognition primarily focuses on gestures, with very little research on collaborative recognition of facial expressions. In the future, researchers could attempt to incorporate the key aspects of micro-expression recognition into continuous sign language recognition as a supplementary aid.

(3) Try to utilize advanced models and algorithms. Algorithms and models are updated iteratively and rapidly, so staying current and exploring improved identification methods and models is essential. At the same time, attention should be given to the conflicting issues of balancing model accuracy and computational load. Address the challenge of handling large-scale and diverse data sets necessary for large models. Address the real-time and online requirements for recognizing sign language. The challenges of recognizing complex continuous sign language include interrupted sentence segmentation, grammar application, and supplementary auxiliary information.

(4) Furthermore, specific content and directional cues associated with sign language recognition require careful attention and study. For example, reinforcement learning and autonomous decision-making. Multimodal intelligence, which involves combining multiple modes of perception such as vision, hearing, and language, enables machines to have a more

comprehensive understanding and interaction capabilities. Personalized, customized services. Human-machine collaborative work, etc.

In addition, we also compared Chinese Sign Language Recognition with sign language recognition in other countries. Typical representatives include American Sign Language, Indian Sign Language, and Arabic Sign Language. As shown in Table 4, the comparison results indicate that a wider range of recognition technologies and methods have been introduced in other countries. Mainstream technologies such as SVM and CNN are mentioned and applied. Taking CNN as an example, Kasapbaşi et al. [124] proposed a CNN-based human-computer interface for American Sign Language recognition, which achieved high accuracy and demonstrated excellent prediction in tested datasets. Musthafa et al. [125] developed an innovative gesture-based sign language recognition system that can automatically detect sign language and recognize various complex gestures and actions. The model applies CNN and image processing methods to determine fingertip positions in static images and convert them to text, which can identify photos of signers taken in real time. Alani et al. [126] addressed the ArSL-CNN model to train a variety of Arabic sign languages and achieved an overall accuracy of 97.29%.

Table 4: Comparison of other sign language recognition

Types of sign language	Year	Methods or approaches	Accuracy
American Sign Language	2015 [127]	DTW	92.40%
American Sign Language	2016 [128]	Charge-transfer touch sensors	92.00%
American Sign Language	2016 [29]	SVM, SIFT, Hu-moments and FD, PCA and LDA, Skin color (YCbCr) with GMM	94.00%
American Sign Language	2019 [129]	ANN-SVM	Higher accuracy
American Sign Language	2019 [130]	Residual neural network	99.40%
American Sign Language	2022 [131]	Deep learning approach	98.69%
American Sign Language	2022 [124]	CNN	99.38%
American Sign Language	2023 [132]	Assistive data glove-neural network	98.00%
Indian Sign Language	2015 [133]	EFD and ANN	95.10%
Indian Sign Language	2016 [134]	SVM	97.50%
Indian Sign Language	2018 [135]	ROI-CNN	99.56%
Indian Sign Language	2019 [136]	CNN, wearable IMUs	94.20%
Indian Sign Language	2020 [137]	CNN	99.72%
Indian Sign Language	2022 [125]	CNN	Accurately
Indian Sign Language	2023 [138]	Multi-stream 3D CNN	92.80%
Arabic Sign Language	2014 [139]	Nave bayes classifier (NBC)–Leap motion controller (LMC)	98.30%
Arabic Sign Language	2015 [140]	Modified k-nearest neighbor (MKNN)	98.90%
Arabic Sign Language	2015 [27]	SIFT, LDA and SVM-kNN	99.00%
Arabic Sign Language	2019 [141]	HOG, HMM	99.33%
Arabic Sign Language	2019 [142]	A Pair of LMCs with GMM based classification	92.00%
Arabic Sign Language	2020 [143]	VGG16 and ResNet152	99.00%
Arabic Sign Language	2020 [144]	Deep convolutional neural network	97.60%

(Continued)

Table 4 (continued)

Types of sign language	Year	Methods or approaches	Accuracy
Arabic Sign Language	2021 [145]	Faster R-CNN	93.00%
Arabic Sign Language	2021 [126]	CNN	97.29%

Some of the same technologies and methods are earlier than domestic research. The expression in Chinese sign language contains the complex connotations of Chinese, involving many aspects such as semantics, grammar, sentence pattern, and ambiguity, unlike the expressions in the English series, which are concise and clear. Therefore, the Chinese Sign Language Recognition is relatively difficult. Meanwhile, most domestic hotspot research draws on the trends and experiences of foreign countries, so it is slightly behind in time.

6 Conclusion

This paper provides a comprehensive review and summary of the methods and technologies used for recognizing Chinese Sign Language over the past 20 years. Around 2014 was a pivotal moment when Chinese Sign Language Recognition methods transitioned from traditional methods to modern AI-based approaches. In the early research, the mainstream technologies were HMM, SVM, and DTW. With the rapid development of modern artificial intelligence technology, various recognition methods based on deep neural networks play an increasingly important role. It is undergoing changes from traditional methods to modern methods based on artificial intelligence. Meanwhile, architectures are transitioning from single models to mixed models. Besides, suitable datasets and evaluation criteria are worth pursuing. Currently, most Chinese sign language datasets are too small and non-standard. Meanwhile, the proportion of experiments is high, and the promotion of applications is insufficient. All of these aspects need improvement. Furthermore, based on the integration of multiple modalities and the intersection of multi-perspective technologies, there is an urgent need to develop systems with efficient and accurate recognition.

As a whole, the Chinese Sign Language Recognition model has achieved favorable overall evaluation indicators. However, it still has a gap compared with advanced sign language recognition models. In particular, due to the uniqueness and complexity of the CSL and the sign language dataset, there are still some issues worthy of further research:

1. The fusion of continuous sign language features.
2. The coordination of lips and facial expressions involved in some gestures.
3. Applications and enabling technologies that can be available to the general public [146].
4. To reduce influencing factors such as background interference, lighting, angles, and non-standardized operations.
5. The challenges of high precision, robustness, real-time performance, and user independence.

In the future, the continuous development of new technologies and the cross-integration of scientific fields will catalyze the progress of Chinese Sign Language Recognition. Hybrid network models, recurrent neural networks, deep learning, and artificial intelligence technologies will further promote theoretical research and algorithm innovation related to sign language recognition, and sign language recognition will achieve more remarkable development.

Acknowledgement: The authors thank the support by National Social Science Foundation Annual Project "Research on Evaluation and Improvement Paths of Integrated Development of Disabled Persons" the National Language Commission's "14th Five-Year Plan" Scientific Research Plan 2023 Project "Domain Digital Language Service Resource Construction and Key Technology Research" and the National Philosophy and Social Sciences Foundation.

Funding Statement: This work was supported by National Social Science Foundation Annual Project "Research on Evaluation and Improvement Paths of Integrated Development of Disabled Persons" (Grant No. 20BRK029), the National Language Commission's "14th Five-Year Plan" Scientific Research Plan 2023 Project "Domain Digital Language Service Resource Construction and Key Technology Research" (YB145-72), and the National Philosophy and Social Sciences Foundation (Grant No. 20BTQ065).

Author Contributions: The authors confirm contribution to the paper as follows: study conception and design: Xianwei Jiang, Yudong Zhang; data collection: Yanqiong Zhang, Juan Lei; analysis and interpretation of results: Xianwei Jiang, Yanqiong Zhang; draft manuscript preparation: Xianwei Jiang, Yudong Zhang. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: All the reviewed research literature and used data in this manuscript include scholarly articles, conference proceedings, books, and reports that are publicly available. The references and citations can be found in the reference list of this manuscript and are accessible through online databases, academic libraries, or by contacting the publishers directly.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

1. Yang, X., Feng, Z., Huang, Z. (2016). Gesture recognition combining gesture main direction and Hausdorff-like distance. *Journal of Computer-Aided Design and Graphics, 28(01),* 75–81.
2. Mei, J., Wang, W., Dai, X. (2021). Continuous sign language recognition based on second-order hidden markov model. *Computer System Applications, 31(4),* 375–380.
3. Li, Y., Yang, Q., Peng, J. (2009). Chinese sign language recognition based on gray-level co-occurrence matrix and other multi-features fusion. *4th IEEE Conference on Industrial Electronics and Applications*, pp. 1569–1572. Xi'an, China.
4. Zhang, J., Zhou, W., Li, H. (2015). A new system for Chinese sign language recognition. *2015 IEEE China Summit and International Conference on Signal and Information Processing (ChinaSIP)*, pp. 534–538. Chengdu, China.
5. Yang, G., Ding, X., Gao, Y. (2023). Complex background continuous sign language recognition based on attention mechanism. *Journal of Wuhan University (Science Edition), 69(1),* 97–105.
6. Zhang, Z., Wu, B., Jiang, Y. (2022). Gesture recognition system based on improved YOLO v3. *7th International Conference on Intelligent Computing and Signal Processing (ICSP)*, pp. 1540–1543. Xi'an, China.
7. Xie, P., Zhao, M., Hu, X. (2022). PiSLTRc: Position-informed sign language transformer with content-aware convolution. *IEEE Transactions on Multimedia, 24,* 3908–3919.

8. Jiang, S., Liu, Y., Jia, H., Lin, P., He, Z. et al. (2023). Research on end-to-end continuous sign language sentence recognition based on transformer. *15th International Conference on Computer Research and Development (ICCRD)*, pp. 220–226. Hangzhou, China.
9. Zhang, J., Wang, Q., Wang, Q., Zheng, Z. (2023). Multimodal fusion framework based on statistical attention and contrastive attention for sign language recognition. *IEEE Transactions on Mobile Computing, 23,* 1431–1443.
10. Solís, F., Martínez, D., Espinoza, O. (2016). Automatic mexican sign language recognition using normalized moments and artificial neural networks. *Engineering, 8(10),* 733–740.
11. Koller, O., Forster, J., Ney, H. (2015). Continuous sign language recognition: Towards large vocabulary statistical recognition systems handling multiple signers-ScienceDirect. *Computer Vision and Image Understanding, 141,* 108–125.
12. Huang, J., Zhou, W., Zhang, Q., Li, H., Li, W. (2018). Video-based sign language recognition without temporal segmentation. *Proceedings of the AAAI Conference on Artificial Intelligence*, New Orleans, Lousiana, USA.
13. Dreuw, P., Neidle, C., Athitsos, V., Sclaroff, S., Ney, H. (2008). *Benchmark databases for video-based automatic sign language recognition*. Marrakech, Morocco: LREC.
14. Oszust, M., Wysocki, M. (2013). Polish sign language words recognition with Kinect. *2013 6th International Conference on Human System Interactions (HSI)*, pp. 219–226. Sopot, Poland.
15. Aliyu, S., Mohandes, M., Deriche, M. (2017). Dual LMCs fusion for recognition of isolated Arabic sign language words. *14th International Multi-Conference on Systems, Signals & Devices (SSD)*, pp. 611–614. Marrakech, Morocco.
16. Baró, X., Gonzalez, J., Fabian, J., Bautista, M. A., Oliu, M. et al. (2015). Chalearn looking at people 2015 challenges: Action spotting and cultural event recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 1–9. Boston, Massachusetts.
17. Yang, S., Jung, S., Kang, H., Kim, C. (2019). The Korean sign language dataset for action recognition. *International Conference on Multimedia Modeling*, pp. 532–542. Daejeon, South Korea.
18. Ronchetti, F., Quiroga, F., Estrebou, C., Lanzarini, L., Rosete, A. (2016). Sign languague recognition without frame-sequencing constraints: A proof of concept on the Argentinian sign language. In: *Advances in artificial intelligence-IBERAMIA 2016*, pp. 338–349. San José, Costa Rica.
19. Haralick, R. M., Shanmugam, K., Dinstein, I. H. (1973). Textural features for image classification. *IEEE Transactions on Systems, Man, and Cybernetics, 6,* 610–621.
20. Hamed, A., Belal, N. A., Mahar, K. M. (2016). Arabic sign language alphabet recognition based on HOG-PCA using microsoft kinect in complex backgrounds. *2016 IEEE 6th International Conference on Advanced Computing (IACC)*, pp. 451–458. Bhimavaram, India.
21. Mahmud, I., Tabassum, T., Uddin, M. P., Ali, E., Nitu, A. M. et al. (2018). Efficient noise reduction and HOG feature extraction for sign language recognition. *2018 International Conference on Advancement in Electrical and Electronic Engineering (ICAEEE)*, pp. 1–4. Gazipur, Bangladesh.
22. Kosmidou, V. E., Hadjileontiadis, L. J. (2009). Sign language recognition using intrinsic-mode sample entropy on sEMG and accelerometer data. *IEEE Transactions on Biomedical Engineering, 56(12),* 2879–2890.
23. Zhu, Z., Zhang, M., Jiang, X. (2021). Fingerspelling identification for Chinese sign language via wavelet entropy and kernel support vector machine. In: *Intelligent data engineering and analytics: Frontiers in intelligent computing: Theory and applications (FICTA 2020),* pp. 539–549, India: NIT Surathkal.
24. Saxena, A., Jain, D. K., Singhal, A. (2014). Sign language recognition using principal component analysis. *2014 Fourth International Conference on Communication Systems and Network Technologies*, pp. 810–813. Bhopal, India.

25. Ghandehari, A., Safabakhsh, R. (2011). A comparison of principal component analysis and adaptive principal component extraction for palmprint recognition. *2011 International Conference on Hand-Based Biometrics*, pp. 1–6. Hong Kong, China.
26. Gweth, Y. L., Plahl, C., Ney, H. (2012). Enhanced continuous sign language recognition using PCA and neural network features. *2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pp. 55–60. Providence, RI, USA.
27. Tharwat, A., Gaber, T., Hassanien, A. E., Shahin, M. K., Refaat, B. (2015). Sift-based arabic sign language recognition system. *Afro-European Conference for Industrial Advancement: Proceedings of the First International Afro-European Conference for Industrial Advancement AECIA 2014*, pp. 359–370. Addis Ababa, Ethiopia.
28. Dardas, N. H., Georganas, N. D. (2011). Real-time hand gesture detection and recognition using bag-of-features and support vector machine techniques. *IEEE Transactions on Instrumentation and Measurement, 60(11),* 3592–3607.
29. Pan, T. Y., Lo, L. Y., Yeh, C. W., Li, J. W., Liu, H. T. et al. (2016). Real-time sign language recognition in complex background scene based on a hierarchical clustering classification method. *2016 IEEE Second International Conference on Multimedia Big Data (BigMM)*, pp. 64–67. Taipei, Taiwan.
30. Gao, Y., Wang, R., Xue, C., Gao, Y., Qiao, Y. et al. (2020). Chinese fingerspelling recognition via Hu moment invariant and RBF support vector machine. *Multimedia Technology and Enhanced Learning*, pp. 382–392. Leicester, UK.
31. Quan, Y., Peng, J. Y. (2008). Chinese sign language recognition for a vision-based multi-features classifier. *2008 International Symposium on Computer Science and Computational Technology*, pp. 194–197. Shanghai, China.
32. Sokic, E., Konjicija, S. (2016). Phase preserving Fourier descriptor for shape-based image retrieval. *Signal Processing: Image Communication, 40,* 82–96.
33. Chanda, P., Auephanwiriyakul, S., Theera-Umpon, N. (2012). Thai sign language translation system using upright speed-up robust feature and c-means clustering. *2012 IEEE International Conference on Fuzzy Systems*, pp. 1–6. Brisbane, QLD, Australia.
34. Monteiro, C. D. D., Shipman, F., Gutierrez-Osuna, R. (2018). Comparing visual, textual, and multimodal features for detecting sign language in video sharing sites. *2018 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, pp. 7–12. Miami, FL, USA.
35. Gao, W., Ma, J., Wu, J., Wang, C. (2000). Sign language recognition based on HMM/ANN/DP. *International Journal of Pattern Recognition and Artificial Intelligence, 14(5),* 587–602.
36. Zhang, J., Zhou, W., Xie, C., Pu, J., Li, H. (2016). Chinese sign language recognition with adaptive HMM. *2016 IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1–6. Seattle, WA, USA.
37. Gao, W., Fang, G., Zhao, D., Chen, Y. (2004). A Chinese sign language recognition system based on SOFM/SRN/HMM. *Pattern Recognition, 37(12),* 2389–2402.
38. Travieso, C. M., Alonso, J. B., Ferrer, M. A. (2003). Sign Language to text by SVM. *Seventh International Symposium on Signal Processing and Its Applications*, pp. 435–438. Paris, France.
39. Ye, J., Yao, H., Jiang, F. (2004). Based on HMM and SVM multilayer architecture classifier for Chinese sign language recognition with large vocabulary. *Third International Conference on Image and Graphics (ICIG'04)*, pp. 377–380. Hong Kong, China.
40. Pu, J., Zhou, W., Li, H. (2016). Sign language recognition with multi-modal features. *Advances in multimedia information processing-PCM 2016*, pp. 252–261. Xi'an, China.
41. Lichtenauer, J. F., Hendriks, E. A., Reinders, M. J. (2008). Sign language recognition by combining statistical DTW and independent classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 30(11),* 2040–2046.
42. Xia, Z., Xing, J., Wang, C., Li, X. (2021). Gesture recognition algorithm of human motion target based on deep neural network. *Mobile Information Systems, 2021,* 1–12.

43. Ajay, S., Potluri, A., George, S. M., Gaurav, R., Anusri, S. (2021). Indian sign language recognition using random forest classifier. *2021 IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT)*, pp. 1–6. Bangalore, India.
44. Yuan, S., Wang, Y., Wang, X., Deng, H., Sun, S. et al. (2020). Chinese sign language alphabet recognition based on random forest algorithm. *2020 IEEE International Workshop on Metrology for Industry 4.0 & IoT*, pp. 340–344. Roma, Italy.
45. Su, R., Chen, X., Cao, S., Zhang, X. (2016). Random forest-based recognition of isolated sign language subwords using data from accelerometers and surface electromyographic sensors. *Sensors, 16(1),* 100–105.
46. Hochreiter, S., Schmidhuber, J. (1997). Long short-term memory. *Neural Computation, 9(8),* 1735–1780.
47. Aly, S., Aly, W. (2020). DeepArSLR: A novel signer-independent deep learning framework for isolated arabic sign language gestures recognition. *IEEE Access, 8,* 83199–83212.
48. Liu, T., Zhou, W., Li, H. (2016). Sign language recognition with long short-term memory. *2016 IEEE International Conference on Image Processing (ICIP)*, pp. 2871–2875. Phoenix, AZ, USA.
49. Yang, S., Zhu, Q. (2017). Continuous Chinese sign language recognition with CNN-LSTM. *Ninth International Conference on Digital Image Processing (ICDIP 2017)*, pp. 83–89. Hong Kong, China.
50. Xiao, Q., Qin, M., Guo, P., Zhao, Y. (2019). Multimodal fusion based on LSTM and a couple conditional hidden Markov model for Chinese sign language recognition. *IEEE Access, 7,* 112258–112268.
51. Pattanaworapan, K., Chamnongthai, K., Guo, J. M. (2016). Signer-independence finger alphabet recognition using discrete wavelet transform and area level run lengths. *Journal of Visual Communication and Image Representation, 38,* 658–677.
52. Pramunanto, E., Sumpeno, S., Legowo, R. S. (2017). Classification of hand gesture in Indonesian sign language system using Naive Bayes. *2017 International Seminar on Sensors, Instrumentation, Measurement and Metrology (ISSIMM)*, pp. 187–191. Surabaya, Indonesia.
53. Wong, S. F., Cipolla, R. (2005). Real-time adaptive hand motion recognition using a sparse bayesian classifier. *Computer Vision in Human-Computer Interaction: ICCV 2005 Workshop on HCI*, pp. 170–179. Beijing, China.
54. McCulloch, W. S., Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics, 5,* 115–133.
55. Barbhuiya, A. A., Karsh, R. K., Jain, R. (2021). CNN based feature extraction and classification for sign language. *Multimedia Tools and Applications, 80(2),* 3051–3069.
56. Masood, S., Srivastava, A., Thuwal, H. C., Ahmad, M. (2018). Real-time sign language gesture (word) recognition from video sequences using CNN and RNN. *Intelligent Engineering Informatics: Proceedings of the 6th International Conference on FICTA*, pp. 623–632. Bhubaneswar, Odisha.
57. Alaria, S. K., Raj, A., Sharma, V., Kumar, V. (2022). Simulation and analysis of hand gesture recognition for indian sign language using CNN. *International Journal on Recent and Innovation Trends in Computing and Communication, 10(4),* 10–14.
58. Rao, G. A., Syamala, K., Kishore, P., Sastry, A. (2018). Deep convolutional neural networks for sign language recognition. *Conference on Signal Processing and Communication Engineering Systems (SPACES)*, pp. 194–197. Vijayawada, India.
59. Jiang, X., Zhang, Y. D. (2019). Chinese sign language fingerspelling via six-layer convolutional neural network with leaky rectified linear units for therapy and rehabilitation. *Journal of Medical Imaging and Health Informatics, 9(9),* 2031–2090.
60. Jiang, X., Lu, M., Wang, S. H. (2020). An eight-layer convolutional neural network with stochastic pooling, batch normalization and dropout for fingerspelling recognition of Chinese sign language. *Multimedia Tools and Applications, 79,* 15697–15715.

61. Koller, O., Bowden, R., Ney, H. (2016). Automatic alignment of hamnosys subunits for continuous sign language recognition. *LREC 2016: 10th Edition of the Language Resources and Evaluation Conference*, pp. 121–128. Portorož, Slovenia.
62. Zhang, S., Zhang, Q., Li, H. (2020). Review of sign language recognition based on deep learning. *Journal of Electronics and Information, 42(4),* 1021–1032.
63. Singh, D. K. (2021). 3D-CNN based dynamic gesture recognition for Indian sign language modeling. *Procedia Computer Science, 189,* 76–83.
64. Huang, J., Zhou, W., Li, H., Li, W. (2015). Sign language recognition using 3D convolutional neural networks. *2015 IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1–6. Turin.
65. ElBadawy, M., Elons, A., Shedeed, H. A., Tolba, M. (2017). Arabic sign language recognition with 3D convolutional neural networks. *2017 Eighth International Conference on Intelligent Computing and Information Systems (ICICIS)*, pp. 66–71. Cairo, Egypt.
66. Chéron, G., Laptev, I., Schmid, C. (2015). P-CNN: Pose-based cnn features for action recognition. *Proceedings of the IEEE International Conference on Computer Vision*, pp. 3218–3226. Santiago, Chile.
67. Molchanov, P., Gupta, S., Kim, K., Kautz, J. (2015). Hand gesture recognition with 3D convolutional neural networks. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 1–7. Boston, Massachusetts.
68. Liang, Z. J, Liao, S. B, Hu, B. Z (2018). 3D convolutional neural networks for dynamic sign language recognition. *The Computer Journal, 61(11),* 1724–1736.
69. Asri, M., Ahmad, Z., Mohtar, I. A., Ibrahim, S. (2019). A real time Malaysian sign language detection algorithm based on YOLOv3. *International Journal of Recent Technology and Engineering, 8(2),* 651–656.
70. Redmon, J., Divvala, S., Girshick, R., Farhadi, A. (2016). You only look once: Unified, real-time object detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 779–788. Las Vegas, Nevada.
71. Li, T., Yan, Y., Du, W. (2022). Sign language recognition based on computer vision. *2022 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA)*, pp. 927–931. Dalian, China.
72. Zhang, Y., Long, L., Shi, D., He, H., Liu, X. (2022). Research and improvement of chinese sign language detection algorithm based on YOLOv5s. *2022 2nd International Conference on Networking, Communications and Information Technology (NetCIT)*, pp. 577–581. Manchester, UK.
73. Xiao, H., Yang, Y., Yu, K., Tian, J., Cai, X. et al. (2022). Sign language digits and alphabets recognition by capsule networks. *Journal of Ambient Intelligence and Humanized Computing, 13,* 2131–2141.
74. Bousbai, K., Merah, M. (2022). Hand gesture recognition using capabilities of capsule network and data augmentation. *2022 7th International Conference on Image and Signal Processing and their Applications (ISPA)*, pp. 1–5. Mostaganem, Algeria.
75. Suri, K., Gupta, R. (2019). Continuous sign language recognition from wearable IMUs using deep capsule networks and game theory. *Computers & Electrical Engineering, 78,* 493–503.
76. Devlin, J., Chang, M. W., Lee, K., Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.
77. Radford, A., Narasimhan, K., Salimans, T., Sutskever, I. (2018). Improving language understanding by generative pre-training. https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf. (accessed on 11/06/2018).
78. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L. et al. (2017). Attention is all you need. *Advances in Neural Information Processing Systems, 30,* 1–11
79. de Coster, M., van Herreweghe, M., Dambre, J. (2020). Sign language recognition with transformer networks. *12th International Conference on Language Resources and Evaluation*, pp. 6018–6024. Marseille, France.

80. Du, Y., Xie, P., Wang, M., Hu, X., Zhao, Z. et al. (2022). Full transformer network with masking future for word-level sign language recognition. *Neurocomputing, 500,* 115–123.
81. Cui, Z., Zhang, W., Li, Z., Wang, Z. (2023). Spatial–temporal transformer for end-to-end sign language recognition. *Complex & Intelligent Systems, 9,* 4645–4656.
82. Farhadi, A., Forsyth, D., White, R. (2007). Transfer learning in sign language. *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8. Minneapolis, MN, USA.
83. Pan, S. J., Tsang, I. W., Kwok, J. T., Yang, Q. (2010). Domain adaptation via transfer component analysis. *IEEE Transactions on Neural Networks, 22(2),* 199–210.
84. Lin, Y. P., Jung, T. P. (2017). Improving EEG-based emotion classification using conditional transfer learning. *Frontiers in Human Neuroscience, 11,* 334.
85. Tan, C., Sun, F., Kong, T., Zhang, W., Yang, C. et al. (2018). A survey on deep transfer learning. *Artificial Neural Networks and Machine Learning-ICANN 2018: 27th International Conference on Artificial Neural Networks*, pp. 270–279. Rhodes, Greece.
86. Jiang, X., Hu, B., Chandra Satapathy, S., Wang, S. H., Zhang, Y. D. (2020). Fingerspelling identification for Chinese sign language via AlexNet-based transfer learning and Adam optimizer. *Scientific Programming, 2020,* 1–13.
87. Liu, T., Zhou, W., Li, H. (2016). Sign language recognition with long short-term memory. *2016 IEEE International Conference on Image Processing (ICIP)*, pp. 2871–2875. Phoenix, AZ.
88. Zhong, H. (2020). Dynamic gesture recognition method based on deep learning. *Electronic Measurement Technology, 43(2),* 128–132.
89. Yin, L., Ying, H., Meng-hao, Y. (2023). Chinese sign language recognition based on two-stream CNN and LSTM network. *International Journal of Advanced Networking and Applications, 14(6),* 5666–5671.
90. Yu, X., Wang, S. H., Zhang, Y. D. (2021). CGNet: A graph-knowledge embedded convolutional neural network for detection of pneumonia. *Information Processing & Management, 58(1),* 102411.
91. Zhou, W., Chen, Y., Liu, C., Yu, L. (2020). GFNet: Gate fusion network with Res2Net for detecting salient objects in RGB-D images. *IEEE Signal Processing Letters, 27,* 800–804.
92. Zhang, L. G., Chen, Y., Fang, G., Chen, X., Gao, W. (2004). A vision-based sign language recognition system using tied-mixture density HMM. *Proceedings of the 6th International Conference on Multimodal Interfaces*, pp. 198–204. State College, PA, USA.
93. Zhang, L. G., Chen, X., Wang, C., Chen, Y., Gao, W. (2005). Recognition of sign language subwords based on boosted hidden markov models. *Proceedings of the 7th International Conference on Multimodal Interfaces*, pp. 282–287. Torento, Italy.
94. Wang, C., Chen, X., Gao, W. (2006). Expanding training set for chinese sign language recognition. *7th International Conference on Automatic Face and Gesture Recognition (FGR06)*, pp. 323–328. Southampton, UK.
95. Fang, G., Gao, W., Zhao, D. (2006). Large-vocabulary continuous sign language recognition based on transition-movement models. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans, 37(1),* 1–9.
96. Li, Y., Chen, X., Tian, J., Zhang, X., Wang, K. et al. (2010). Automatic recognition of sign language subwords based on portable accelerometer and EMG sensors. *International Conference on Multimodal Interfaces and the Workshop on Machine Learning for Multimodal Interaction*, pp. 1–7. Beijing, China.
97. Zhang, X., Chen, X., Li, Y., Lantz, V., Wang, K. et al. (2011). A framework for hand gesture recognition based on accelerometer and EMG sensors. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans, 41(6),* 1064–1076.
98. Li, Y., Chen, X., Zhang, X., Wang, K., Wang, Z. (2012). A sign-component-based framework for Chinese sign language recognition using accelerometer and sEMG data. *IEEE Transactions on Biomedical Engineering, 59(10),* 2695–2704.

99. Yang, Q., Peng, J. Y. (2014). Chinese sign language recognition method based on depth image information and SURF-BoW. *Journal of Pattern Recognition and Artificial Intelligence, 8,* 1–10.
100. Geng, L., Ma, X., Wang, H., Gu, J., Li, Y. (2014). Chinese sign language recognition with 3D hand motion trajectories and depth images. *Proceedings of the 11th World Congress on Intelligent Control and Automation*, pp. 1457–1461. Shenyang, China.
101. Wang, H., Chai, X., Zhou, Y., Chen, X. (2015). Fast sign language recognition benefited from low rank approximation. *2015 11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*, pp. 1–6. Ljubljana, Slovenia.
102. Chen, Y., Zhang, W. (2016). Research and implementation of sign language recognition method based on Kinect. *2016 2nd IEEE International Conference on Computer and Communications (ICCC),* Chengdu, China.
103. Yang, S., Zhu, Q. (2017). Video-based Chinese sign language recognition using convolutional neural network. *2017 IEEE 9th International Conference on Communication Software and Networks (ICCSN)*, pp. 929–934. Guangzhou, China.
104. Song, N., Yang, H., Wu, P. (2018). A gesture-to-emotional speech conversion by combining gesture recognition and facial expression recognition. *2018 First Asian Conference on Affective Computing and Intelligent Interaction (ACII Asia)*, pp. 1–6. Beijing, China.
105. Liang, Z. J., Liao, S. B., Hu, B. Z. (2018). 3D convolutional neural networks for dynamic sign language recognition. *The Computer Journal, 61(11),* 1724–1736.
106. Huang, S., Mao, C., Tao, J., Ye, Z. (2018). A novel Chinese sign language recognition method based on keyframe-centered clips. *IEEE Signal Processing Letters, 25(3),* 442–446.
107. Huang, J., Zhou, W., Li, H., Li, W. (2018). Attention-based 3D-CNNs for large-vocabulary sign language recognition. *IEEE Transactions on Circuits and Systems for Video Technology, 29(9),* 2822–2832.
108. Liao, Y., Xiong, P., Min, W., Min, W., Lu, J. (2019). Dynamic sign language recognition based on video sequence with BLSTM-3D residual networks. *IEEE Access, 7,* 38044–38054.
109. Zhang, Z., Su, Z., Yang, G. (2019). Real-time Chinese Sign Language Recognition based on artificial neural networks. *2019 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, 1413–1417, Dali, China.
110. Huang, J., Zhou, W., Li, H., Li, W. (2019). Attention-based 3D-CNNs for large-vocabulary sign language recognition. *IEEE Transactions on Circuits and Systems for Video Technology, 29(9),* 2822–2832.
111. Yu, Y., Chen, X., Cao, S., Zhang, X., Chen, X. (2019). Exploration of Chinese sign language recognition using wearable sensors based on deep belief net. *IEEE Journal of Biomedical and Health Informatics, 24(5),* 1310–1320.
112. Ni, Q., Qiao, J., Lian, Z. (2020). DTW dynamic gesture recognition method with improved K-means clustering pruning. *Modern Computer, 699(27),* 20–25.
113. Zhang, Y., Xu, W., Zhang, X., Li, L. (2022). Sign annotation generation to alphabets via integrating visual data with somatosensory data from flexible strain sensor-based data glove. *Measurement, 202,* 111700.
114. Han, X., Lu, F., Tian, G. (2022). Efficient 3D CNNs with knowledge transfer for sign language recognition. *Multimedia Tools and Applications, 81(7),* 10071–10090.
115. Guo, Q., Zhang, S., Li, H. (2023). Continuous sign language recognition based on spatial-temporal graph attention network. *Computer Modeling in Engineering & Sciences, 134(3),* 1653–1670. doi: 10.32604/cmes.2022.021784.
116. Saeed, Z. R., Zainol, Z. B., Zaidan, B., Alamoodi, A. (2022). A systematic review on systems-based sensory gloves for sign language pattern recognition: An update from 2017 to 2022. *IEEE Access, 10,* 123358–123377
117. Kim, H. J., Baek, S. W. (2023). Application of wearable gloves for assisted learning of sign language using artificial neural networks. *Processes, 11(4),* 1065.

118. Ahmed, M. A., Zaidan, B. B., Zaidan, A. A., Salih, M. M., Lakulu, M. M. B. (2018). A review on systems-based sensory gloves for sign language recognition state of the art between 2007 and 2017. *Sensors, 18(7),* 2208.
119. Wu, J., Sun, L., Jafari, R. (2016). A wearable system for recognizing American sign language in real-time using IMU and surface EMG sensors. *IEEE Journal of Biomedical and Health Informatics, 20(5),* 1281–1290.
120. Zhang, N., Zhang, J., Ying, Y., Luo, C., Li, J. (2022). Wi-Phrase: Deep residual-multihead model for wifi sign language phrase recognition. *IEEE Internet of Things Journal, 9(18),* 18015–18027.
121. Abdullahi, S. B., Chamnongthai, K. (2022). American sign language words recognition using spatio-temporal prosodic and angle features: A sequential learning approach. *IEEE Access, 10,* 15911–15923.
122. Lang, S., Block, M., Rojas, R. (2012). Sign language recognition using kinect. *International Conference on Artificial Intelligence and Soft Computing*, pp. 394–402. Berlin, Heidelberg.
123. Zafrulla, Z., Brashear, H., Starner, T., Hamilton, H., Presti, P. (2011). American sign language recognition with the kinect. *Proceedings of the 13th International Conference on Multimodal Interfaces*, pp. 279–286. Alicante, Spain.
124. Kasapbaşi, A., Elbushra, A. E. A., Omar, A. H., Yilmaz, A. (2022). DeepASLR: A CNN based human computer interface for American sign language recognition for hearing-impaired individuals. *Computer Methods and Programs in Biomedicine Update, 2,* 100048.
125. Musthafa, N., Raji, C. (2022). Real time Indian sign language recognition system. *Materials Today: Proceedings, 58,* 504–508.
126. Alani, A. A., Cosma, G. (2021). ArSL-CNN: A convolutional neural network for Arabic sign language gesture recognition. *Indonesian Journal of Electrical Engineering and Computer Science, 22,* 1096–1107.
127. Plouffe, G., Cretu, A. M. (2015). Static and dynamic hand gesture recognition in depth data using dynamic time warping. *IEEE Transactions on Instrumentation and Measurement, 65(2),* 305–316.
128. Abhishek, K. S., Qubeley, L. C. F., Ho, D. (2016). Glove-based hand gesture recognition sign language translator using capacitive touch sensor. *2016 IEEE International Conference on Electron Devices and Solid-State Circuits (EDSSC)*, pp. 334–337. Hong Kong, China.
129. Fatmi, R., Rashad, S., Integlia, R. (2019). Comparing ANN, SVM, and HMM based machine learning methods for American sign language recognition using wearable motion sensors. *2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC)*, pp. 0290–0297. Las Vegas, NV, USA.
130. Xie, M., Ma, X. (2019). End-to-end residual neural network with data augmentation for sign language recognition. *2019 IEEE 4th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, pp. 1629–1633. Chengdu, China.
131. Susa, J. A. B., Macalisang, J. R., Sevilla, R. V., Evangelista, R. S., Quismundo, A. Q. et al. (2022). Implementation of security access control using american sign language recognition via deep learning approach. *2022 International Conference on Emerging Technologies in Electronics, Computing and Communication (ICETECC)*, pp. 1–5. Jamshoro, Sindh, Pakistan.
132. Amin, M. S., Rizvi, S. T. H., Mazzei, A., Anselma, L. (2023). Assistive data glove for isolated static postures recognition in american sign language using neural network. *Electronics, 12(8),* 1904.
133. Kishore, P., Prasad, M. V., Prasad, C. R., Rahul, R. (2015). 4-Camera model for sign language recognition using elliptical fourier descriptors and ANN. *2015 International Conference on Signal Processing and Communication Engineering Systems*, pp. 34–38. Guntur, India.
134. Raheja, J., Mishra, A., Chaudhary, A. (2016). Indian sign language recognition using SVM. *Pattern Recognition and Image Analysis, 26(2),* 434–441.
135. Sajanraj, T., Beena, M. (2018). Indian sign language numeral recognition using region of interest convolutional neural network. *2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT)*, pp. 636–640. Coimbatore, India.

136. Suri, K., Gupta, R. (2019). Convolutional neural network array for sign language recognition using wearable IMUs. *2019 6th International Conference on Signal Processing and Integrated Networks (SPIN)*, pp. 483–488. Noida, India.

137. Wadhawan, A., Kumar, P. (2020). Deep learning-based sign language recognition system for static signs. *Neural Computing and Applications, 32,* 7957–7968.

138. de Castro, G. Z., Guerra, R. R., Guimarães, F. G. (2023). Automatic translation of sign language with multi-stream 3D CNN and generation of artificial depth maps. *Expert Systems with Applications, 215,* 119394.

139. Mohandes, M., Aliyu, S., Deriche, M. (2014). Arabic sign language recognition using the leap motion controller. *2014 IEEE 23rd International Symposium on Industrial Electronics (ISIE)*, pp. 960–965. Istanbul, Turkey.

140. Tubaiz, N., Shanableh, T., Assaleh, K. (2015). Glove-based continuous Arabic sign language recognition in user-dependent mode. *IEEE Transactions on Human-Machine Systems, 45(4),* 526–533.

141. Sidig, A. A. I., Luqman, H., Mahmoud, S. A. (2019). Arabic sign language recognition using vision and hand tracking features with HMM. *International Journal of Intelligent Systems Technologies and Applications, 18(5),* 430–447.

142. Deriche, M., Aliyu, S. O., Mohandes, M. (2019). An intelligent arabic sign language recognition system using a pair of LMCs with GMM based classification. *IEEE Sensors Journal, 19(18),* 8067–8078.

143. Saleh, Y., Issa, G. (2020). Arabic sign language recognition through deep neural networks fine-tuning. *iJOE*, *16(5)*, 71–83.

144. Latif, G., Mohammad, N., AlKhalaf, R., AlKhalaf, R., Alghazo, J. et al. (2020). An automatic Arabic sign language recognition system based on deep CNN: An assistive system for the deaf and hard of hearing. *International Journal of Computing and Digital Systems, 9(4),* 715–724.

145. Alawwad, R. A., Bchir, O., Ismail, M. M. B. (2021). Arabic sign language recognition using faster R-CNN. *International Journal of Advanced Computer Science and Applications, 12(3),* 692–700.

146. Boggaram, A., Boggaram, A., Sharma, A., Ramanujan, A. S., Bharathi, R. (2022). Sign language translation systems: A systematic literature review. *International Journal of Software Science and Computational Intelligence (IJSSCI), 14(1),* 1–33.