Contents lists available at ScienceDirect

Intelligent Systems with Applications

Machine learning methods for sign language recognition: A critical review and analysis

I.A. Adeyanju [a], O.O. Bello [b], M.A. Adegboye [a,c,*]

[a] Department of Computer Engineering, Federal University, Oye-Ekiti, Nigeria
[b] Department of Computer Engineering, Ekiti State University, Ado-Ekiti, Nigeria
[c] Communications and Autonomous Systems Group, School of Engineering, Robert Gordon University, Aberdeen AB107GJ, United Kingdom

ARTICLE INFO

ABSTRACT

Sign language is an essential tool to bridge the communication gap between normal and hearing-impaired people. However, the diversity of over 7000 present-day sign languages with variability in motion position, hand shape, and position of body parts making automatic sign language recognition (ASLR) a complex system. In order to overcome such complexity, researchers are investigating better ways of developing ASLR systems to seek intelligent solutions and have demonstrated remarkable success. This paper aims to analyse the research published on intelligent systems in sign language recognition over the past two decades. A total of 649 publications related to decision support and intelligent systems on sign language recognition (SLR) are extracted from the Scopus database and analysed. The extracted publications are analysed using bibliometric VOSViewer software to (1) obtain the publications temporal and regional distributions, (2) create the cooperation networks between affiliations and authors and identify productive institutions in this context. Moreover, reviews of techniques for vision-based sign language recognition are presented. Various features extraction and classification techniques used in SLR to achieve good results are discussed. The literature review presented in this paper shows the importance of incorporating intelligent solutions into the sign language recognition systems and reveals that perfect intelligent systems for sign language recognition are still an open problem. Overall, it is expected that this study will facilitate knowledge accumulation and creation of intelligent-based SLR and provide readers, researchers, and practitioners a roadmap to guide future direction.

1. Introduction

Communication is an essential tool in human existence. It is a fundamental and effective way of sharing thoughts, feelings and opinions. However, a substantial fraction of the world's population lacks this ability (El-Din & El-Ghany, 2020). Many people are suffering from hearing loss, speaking impairment or both. A partial or complete inability to hear in one or both ears is known as hearing loss. On the other hand, mute is a disability that impairs speaking and makes the affected people unable to speak. If deaf-mute happens during childhood, their language learning ability can be hindered and results in language impairment, also known as hearing mutism. These ailments are part of the most common disabilities worldwide (Hasan et al., 2020). Statistical report of physically challenged children during the past decade reveals an increase in the number of neonates born with a defect of hearing impairment and creates a communication barrier between them and the rest of the world (Krishnaveni et al., 2019).

According to the World Health Organization (WHO) report, the number of people affected by hearing disability in 2005 was approximately 278 million worldwide (Savur & Sahin, 2016). Ten (10) years later, this number jumped to 360 million, a roughly 14% increment (Savur & Sahin, 2016). Since then, the number has been increasing exponentially. The latest report of WHO revealed that 466 million people were suffering from hearing loss in 2019, which amount to 5% of the world population with 432 million (or 83%) of them being adults, and 34 million (17%) of them are children (Bin et al., 2019; Hisham & Hamouda, 2019; Saleh & Issa, 2020). The WHO also estimated that the number would double (i.e. 900 million people) by 2050 (El-Din & El-Ghany, 2020). In these fast-growing deaf-mute people, there is a need to break the commu-

* Corresponding author at: Communications and Autonomous Systems Group, School of Engineering, Robert Gordon University, Aberdeen AB107GJ, United Kingdom.
E-mail addresses: bello.oluwaseyi@eksu.edu.ng (O.O. Bello), m.adegboye@rgu.ac.uk (M.A. Adegboye).

nication barrier that adversely affects the lives and social relationships of deaf-mute people.

Sign languages are used as a primary means of communication by deaf and hard of hearing people worldwide (Izzah & Suciati, 2014). It is the most potent and effective way to bridge the communication gap and social interaction between them and the able people. Sign language interpreters help solve the communication gap with the hearing impaired by translating sign language into spoken words and vice versa. However, the challenges of employing interpreters are the flexible structure of sign languages combined with insufficient numbers of expert sign language interpreters across the globe (Kudrinko et al., 2021). According to the World Federation of Deaf, more than 300 sign languages are using by more than 70 million worldwide (Rastgoo et al., 2021b). Therefore, the need for a technology-based system that can complement conventional sign language interpreters.

Sign language involves the usage of the upper part of the body, such as hand gestures (Gupta & Rajan, 2020), facial expression (Chowdhry et al., 2013), lip-reading (Cheok et al., 2019), head nodding and body postures to disseminate information (Butt et al., 2019); (Rastgoo et al., 2021b; Lee et al., 2021a). The key techniques for sign language recognition are vision-based and wearable sensing modalities such as sensory gloves. Sign language recognition systems based on these approaches have been proposed by several researchers (Ionescu et al., 2005; Yu et al., 2010; Li et al., 2015; Sonkusare et al., 2015; Bobic et al., 2016; Islam et al., 2017; Islam et al., 2017; Saha et al., 2018; Rastgoo et al., 2021a; Xu et al., 2021). The glove-based employs the mechanical or optical sensors attached to the glove worn by the user and converts finger movements into electrical signals for determining the hand posture for recognition. In a vision-based approach, the features corresponding to the palms, finger position and joint angles are estimated, which are then used to perform recognition. This method requires acquiring images or videos of the signs through a camera and processing using image processing techniques.

The recent advancement in Artificial Intelligent (AI) on sign language recognition has paved the way for research communities to apply AI in sign interpreting operations. There are several excellent references on intelligent systems for sign language recognition (Admasu & Raimond, 2010; Deriche et al., 2019; Zapata et al., 2019; Song et al., 2021; Lee et al., 2021a, 2021b; Gao et al., 2021). More recently, further attention has been given to the intelligent based SLR system as it is now being applied in many applications. This includes robotics (Ryumin et al., 2019), interpreting services, real-time multi-person recognition systems, games, virtual reality environments, natural language communications, online hand tracking of human communication in desktop environments, and human-computer interaction (Deng et al., 2017; Supančič et al., 2018; Wadhawan & Kumar, 2020; Rastgoo et al., 2021b). Despite historical research progress and remarkable achievements made in intelligent sign language recognition systems, there is still great potential in creating intelligent solutions to sign language recognition systems. Therefore, this study aims to provide a systematic and comprehensive review of research papers published in the field of intelligent sign language recognition systems to gain insights into the application of decision support and intelligent systems in this context. The major objectives of this study include 1. Analysis of the research published on intelligent systems in sign language recognition using bibliometric analysis on 649 publications extracted from the Scopus database for the period of two decades (2001–2021). 2. Provides comprehensive review in the context of sign language recognition systems with an explicit focus on decision support and expert system technologies over the past two decades. 3. Highlight open issues and possible research areas for future consideration.

The remainder of the paper is organised as follows. Section 2 presents the methodology for the systematic review and bibliometric analysis of the selected articles for studies. Section 3 provides a review of the techniques used in sign language recognition systems. This includes techniques used for data acquisitions, pre-processing, segmentation, feature extraction, and classification. In Section 4, a comprehensive review of decision support and intelligent system algorithms used in sign language recognition is presented. The conclusions and discussion on future needs are presented in Section 5.

2. Bibliometric analysis

This section employs a bibliometric analysis methodology to analyse the research published on intelligent systems in sign language recognition over the past two decades. This is done to discern and understand historical trends and publication patterns over time by journals, regions and cooperation between institutions and organizations. Over the years, there has been increasing research interest in ASLR. Therefore, investigating the overall research trend and the new research direction will be compelling in this field. A comprehensive bibliometric analysis of the state of research trend related to decision support and intelligent systems on sign language recognition from 2001 to 2021 was searched and analysed. This is to identify the potential research gap and highlight the boundaries of knowledge during this time frame. The choice of two decades of research was to give a broader view of how research has progressed during these periods to enable researchers to understand the research pattern activities and their characteristics. This information is helpful as it elucidates the scientific activities regarding publications trends across the countries, institutions, journals and authors.

The keywords used for data collection are "Sign Language recognition", "Intelligent Systems", "Artificial intelligence", "Decision support system", "Machine learning", "Neural network", "Expert system", "Fuzzy system", and "Knowledge-based systems". The search strategy in retrieving publication during the selected time frame is as follows:

(("sign AND language AND recognition") AND (machine AND learning) OR (artificial AND intelligent)) AND **DOCUMENT TYPE:** *(research article) AND* **PUBLISHED YEAR:** *(>2000)* **INDEX:** *(Scopus database).* The introduction of quotation marks (" ") in the search field aimed at retrieving exact keyword or phrase search. This is to identify the articles focusing on the subject of the study topic. The analysis was based on countries, institutions, publication distribution sources, citations and co-occurrence of search keywords. These indicators were selected as a benchmark to highlight the countries, institutions, and leading experts at the forefront of ASLR. It also guides in exploring collaborations networks across these indicators. The two leading tools for carryout bibliometric analyses are VOSviewer and CitNetExplorer software. The VOSviewer was employed in this study because it focuses on the analysis at the level of the aggregate publication, while the CitNetExplorer emphasises individual publications. The VOSviewer enables creation mapping from the network dataset to explore and visualize links in scientific publications, journals, citations, countries, institutions, and authorship.

2.1. Bibliometric analysis procedure

The procedure for bibliometric analysis is shown in Fig. 1. According to the process, the first two steps involve collecting extensive literature from the Scopus database on March 20, 2021. The rationale for selecting the Scopus database was that its indexed journals are globally recognised as influential and as top places for modern and eminent research outputs on sign language recognition systems. Therefore, it is possible to search for and obtained a significant proportion of published articles in this context.
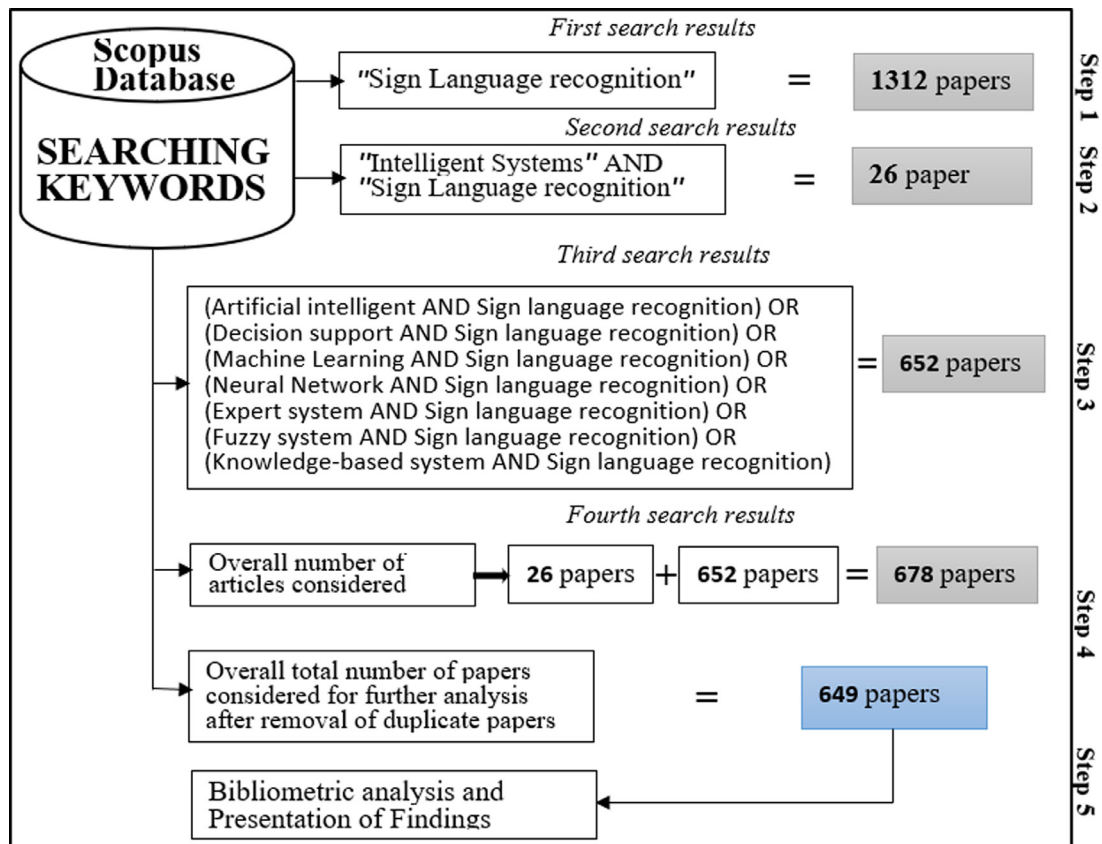
Fig. 1. Flowchart of publication gathering and analysis research output in Scopus database.

Different keywords were used collectively to extract research articles published on intelligent sign language recognition systems throughout the search process. The authors firstly searched the Scopus database using the keyword "Sign Language recognition", and the search resulted in **1312** journal articles published between 2001 and 2021.

Since the focus of this study is on intelligent systems in sign language recognition. After conducting the first search step on general sign language recognition, the authors repeated this process by refining the search using keywords in step 2 ("*Intelligent Systems*" AND "*Sign Language recognition*"). This search resulted in **26** journal articles that are focused on intelligent-based sign recognition systems. However, the search result of 26 articles is likely not sufficient to determine research trends in intelligent-based sign language recognition systems over the last two decades. Therefore, the authors further explored the literature and identified the different intelligent system types employed in sign language recognition systems. The identified intelligent system types include *Artificial intelligence, Decision support system, Machine learning, Neural network, Expert system, Fuzzy system, and Knowledge-based systems*. These search terms were combined into a search string using the conjunction *AND* and disjunction *OR* operators. The query string's output was **652** papers and combined with initial **26** journals for further analysis. To avoid any confusion and duplicates in the final articles selected for analysis, the authors further filtered out the identical papers in step 4. Finally, **649** published articles were considered for bibliometric analysis to determine research trends on intelligent-based sign language recognition systems. The VOSViewer, which is a software tool for constructing and visualising bibliometric networks, is used for step 5. The obtained result is presented in Section 2.2.

2.2. Findings and discussion

The findings of the bibliometric evaluation of the research published on intelligent systems in sign language recognition over the past two decades are presented under different sections. Each of the three sections presents the findings in connection to a specific variable.

2.2.1. Timescale distribution of publications

Fig. 2 shows the number of publications indexed in the Scopus database on intelligent systems sign language recognition between two decades (2001 to 2021). As shown in Fig. 2, the largest number of publications were published in 2020 with a publications figure of 114 articles, followed by the year 2019 recorded **91** papers. The least number of papers was published in the year 2003. The annual rate of research output in 2019 and 2020 is more than double the yearly publications produced since 2001. Please note that this research was conducted on March 20 2021. Only **28** papers were recorded in the Scopus database for the year 2021 when conducting this study. Therefore, the manuscript submitted in the previous years or early year 2021 may be published anytime from now upward. As it can be observed in Fig. 2, the publication trend highlights an increase in the number of researches in the intelligent sign language recognition systems from 2011 upwards with **495** papers; whereas the 2001 decade featured **154** articles out of **649** recorded for the entire two decades. Therefore, this trend evidently indicates that intelligent systems are gaining attention in the research area of sign language recognition, most especially from 2011 upwards. The publication trend is consistent with the rapid increase in the number of people affected by hearing disability (Bin et al., 2019). Beside, the growth rate of the cumulative publications
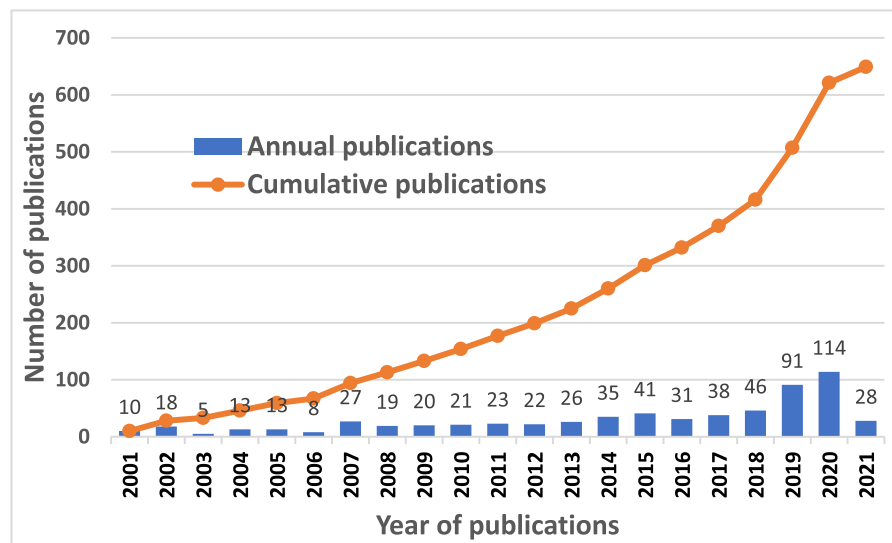
Fig. 2. Timescale distribution of the annual and cumulative publications intelligent-based sign language recognition in Scopus database from 2001 until 2021.

Table 1
The top 12 most active journals on Intelligent-based sign language recognition.

	Journals	TP (%)	TC	CiteScore 2021	Publishers
1	Sensors Switzerland	15	113,885	5.7	MDPI
2	Multimedia tools and Applications	14	25,844	4.5	Springer Nature
3	Neural Computing And Applications	13	19,996	7.1	Springer Nature
4	IEEE Sensors Journal	11	29,411	5.9	IEEE
5	IEEE Access	10	191,118	4.9	IEEE
6	Expert Systems With Applications	9	33,782	12.5	Elsevier Ltd
7	Neurocomputing	9	48,529	9.6	Elsevier Ltd
8	IEEE Transactions On Systems Man And Cybernetics Part B Cybernetics	8	–	–	IEEE
9	Journal Of Ambient Intelligence And Humanized Computing	7	5153	5.3	Springer Nature
10	Journal Of Theoretical And Applied Information Technology	7	2964	1.3	Little Lion Scientific
11	IEEE Transaction On Fuzzy Systems	6	16,853	18.1	IEEE
12	Journal of Biomedical Informatics	6	5545	7.9	Elsevier Ltd

signifies that the research on intelligent sign language recognition systems is still a hot research area. Therefore, it is expected that the annual publication will continue to increase.

2.2.2. Publications distribution sources

Our findings showed that 367 different journals were used to publish 649 articles recorded on the Scopus database from 2001 to 2021 in the area of intelligent-based sign recognition systems. To circumvent a lengthy table, most active journals as per the domain of this study are presented in Table 1.

The topmost journal was published by Multidisciplinary Digital Publishing Institute (MDPI), whereas IEEE journal tops the highest number of journals with four in total. The rest of the three journals were published by Springer Nature, Little lion Science, and Elsevier Ltd. Among which Springer and Elsevier journal has three journals each. Little Lion Scientific publisher has one journal that makes to 10 journals. Sensors Switzerland journal indubitably plays a dominant role in the domain with 15 publications and 113,885 total journal citations. This is followed by Multimedia Tools and Applications (14 publications). Journal of Ambient Intelligence and Journal of Theoretical and Applied Information Technology have seven journals each. IEEE Transaction on Fuzzy Systems and Journal of Biomedical Informatics share an equal number of papers (6 publications each). According to the CiteScore 2020 report, eight journals had CiteScore of 5 and above. The topmost leading journal scope is broad, covering different disciplines of science and technology about the sensor and its applications. The frequency of pub-

lication and being open access journal may also be attributed to the Sensor journal publishing the highest number of articles.

2.2.3. Geographical distribution of publications

We analysed the countries where the research was carried out based on a country's unique occurrence in the affiliations of each article using the VOSViewer tool. The presence of each country is then summed up to the total number of publications published per country. Note that a publication might sum to more than one country. That leads to 797 research output across 78 countries which published at least one full-length research article on intelligent systems in sign language recognition over the past two decades. Fig. 3 shows the geographical map of the top 16 most leading countries contributing to the growth of SLR systems research over the past two decades. India, China and U.S. contributed about 38.14% of the global publications.

The three countries play a key role in advancing sign language recognition research, with India leading worldwide. India led with 123 publications over the past two decades, covering 15.4%of the total global publications. Both China and U.S contributed 13.93% and 8.8%, respectively. In addition to Fig. 3, we also listed the most active institutions for the topmost 16 countries in Table 2. Maunala Azad National Institute of Technology, Bhopal top the list in India. University of Science and Technology of China, Hefei and Missouri University of Science and Technology, Rolla, Mo top in China and U.S., respectively.

Collaboration between institutions across different countries leads to a paper having more than one affiliation or country. A vi-
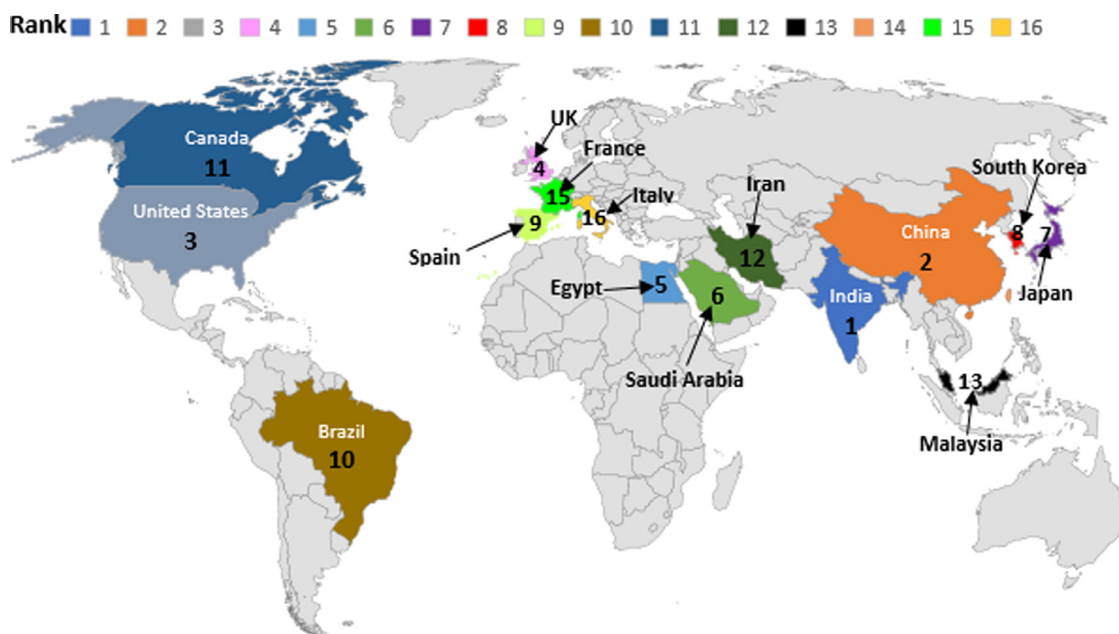
Fig. 3. Regional rank of the top 16 countries and academic institutions with the highest number of publications between 2001 and 2021.

Table 2
The 16 top most active country and academic institutions in each geographical location between 2001 and 2021. TPs: total publication of a given country; TCC: total citation of a given country.

Rank	Country	TPs	TCC	The most published academic institute	Rank	Country	TPs	TCC	The most published academic institute
1	India	123	2118	Maunala Azad National Institute of Technology, Bhopal, India.	9	Spain	19	360	Centro de Biotecnologíay Genómica de Plantas, Upm-inia, Spain.
2	China	111	1353	University of Science and Technology of China, Hefei, China.	10	Brazil	18	243	av. joão naves de ávila, 2160, Uberlândia, Mg, Brazil.
3	U.S.	70	3872	Missouri University of Science and Technology, Rolla, Mo, United States.	11	Canada	17	328	University of Ottawa, 800 king Edward, Ottawa, Canada.
4	U.K.	29	632	Loughborough University, Loughborough, United Kingdom.	12	Iran	17	186	Semnan University, Semnan, 3513119111, Iran.
5	Egypt	26	178	Al-azhar University, Cairo, Egypt.	13	Malaysia	17	203	Multimedia University, Jalan Ayer Keroh lama, Melaka, Malaysia.
6	Saudi Arabia	24	169	Majmaah University, Majmaah, Saudi Arabia.	14	Taiwan	17	588	China University of Technology, Taipei, Taiwan.
7	Japan	23	316	Daicel Corporation, Osaka, 530-0001, Japan.	15	France	16	288	Chru-nancy, Service de Neurologief-France.
8	South Korea	19	253	Keimyung University, Daegu, South Korea.	16	Italy	16	610	University of Trento, Italy.

sualization network is developed using co-authorship analysis of VOSViewer to explore the research cooperation between different affiliations. The distribution of countries or territories per region is illustrated in Fig. 4. The nodes (squares) in the network symbolize countries, and their size depends on the countries' cooperation. The colour in the visualization network characterises the collaboration network. The closer the countries located to each other in the network, the sturdier the link, the thicker the line and the stronger their relatedness. As shown in Fig. 4, the largest thickness is observed between the United States and the United Kingdom. Results of co-authorship collaboration showed that the United States is the most affiliated country, linked to 41 countries/territories. The list was followed by China (35 links), the United Kingdom (27 links), France (16 links), Saudi Arabia (13 links), Singapore (13 links), India (12 links), Turkey (12 links), Germany (11 links), Italy (11 links), Switzerland (10 links), and other 55 countries connected from a range of nine to one link. It was observed that about 17% of the listed countries had international collaborative publications with not less than ten countries. The possible contributory factors to the dynamic of international collaboration among these countries can be attributed to the diversity of research partners, substantial research funding, and many foreigners visiting or postgraduate schol-
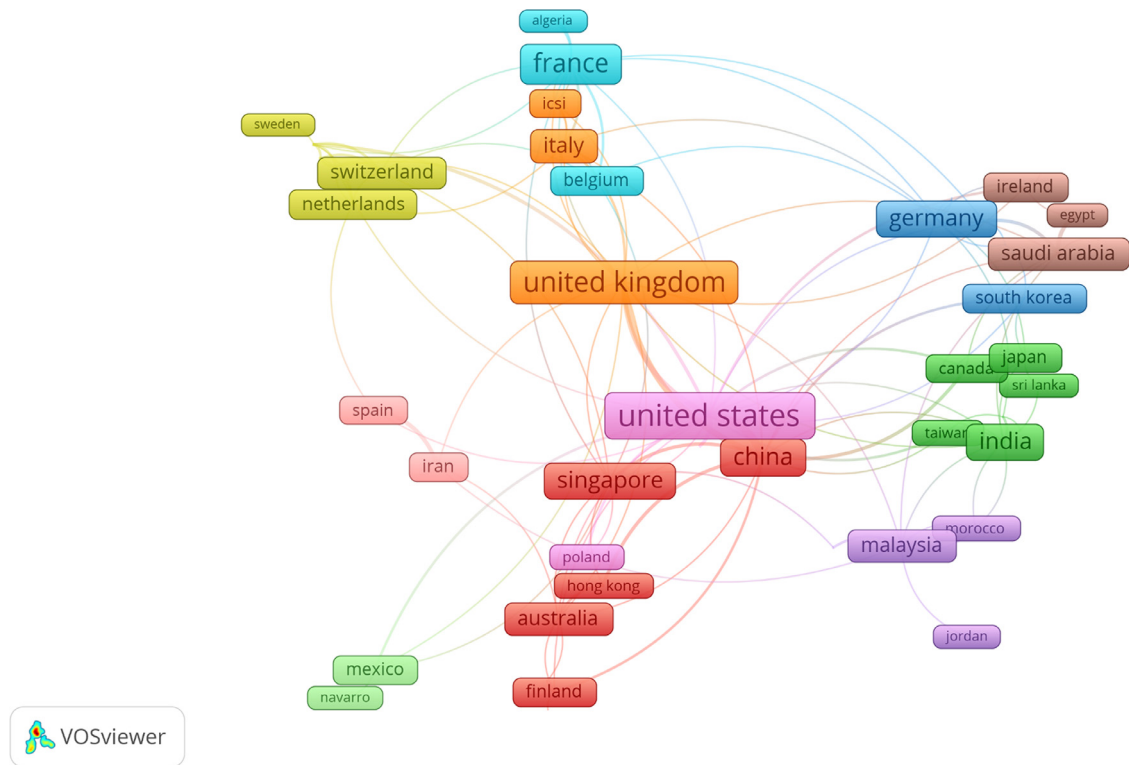
Fig. 4. A screenshot of cooperation network between different institutions on intelligent-based sign language recognition systems in Scopus database publications between 2001 and 2021.

ars. Flexible and stable research policy can also be attributed to the international collaboration in most high-linked countries.

3. Review of vision-based sign language recognition techniques

The stages involved in vision-based sign language recognition (SLR) can be categorised into five stages: image acquisition, image pre-processing, segmentation, feature extraction, and classification, as shown in Fig. 5. Image acquisition is the first stage in sign language recognition that can be acquired through self-created or available public datasets. The second stage is preprocessing to eliminate unwanted noise and enhanced the quality of the image. Next, after preprocessing step is to segment and extract the region of interest from the entire image. The fourth stage is feature extraction, which transforms the input image region into feature vectors for recognition. The last stage in vision-based SLR is classification, which involves matching the features of the new sign image with the stored features in the database for recognition of the given sign (Raj & Jasuja, 2018; Cheok et al., 2019).

3.1. Image acquisition devices

Researchers have used different image acquisition devices to acquire sign images for classifications. These devices include the camera or webcam, data glove, Kinect and leap motion controller (Kamal et al., 2019). Among these devices, a camera or webcam is the most widely used by many researchers because it provides better and natural human-computer interaction without additional devices, unlike data glove based. Data glove has proven to be more accurate in data acquisition but very costly and inconvenient for the users. Kinect is widely used and effective. It provides both colour video and depth video stream simultaneously. It easily separates the background from the actual sign image and extracts 3D trajectories of hand motions (Kamal et al., 2019). The shortcoming of Kinect, however, is the cost implication as it is very costly.

The leap motion controller can operate in a limited range, but it is a low-cost device with better accuracy than Kinect (Suharjito et al., 2017). The uses of a camera for sign images acquisition can be found in the literature (Wang et al., 2010; Mekala et al., 2011; Kumarage et al., 2011; Pansare & Ingle, 2016; Athira et al., 2019; Sharma et al., 2021). Data glove was used to acquire sign images in hand gesture recognition studies proposed by Mehdi and Khan (2002), Gao et al. (2004), Phi et al. (2015) and Pan et al. (2020). Kinect was used to acquire sign images by Jiang et al. (2015), Wang et al. (2015a, 2015b, 2016), Raheja et al. (2016), Carneiro et al. (2017) and Escobedo and Camara (2017). Similar studies employed a leap motion controller for sign image acquisition (Kiselev et al., 2019; Alnahhas et al., 2020; Enikeev & Mustafina, 2021). Table 3 shows the advantages and disadvantages of different data acquisition devices for classifications. These studies revealed that the images acquired were either static or dynamic captured under different positions, background and lighting conditions as frames of images.

3.2. Image preprocessing techniques

Preprocessing techniques are applied to an input image to remove unwanted noise and also enhance the quality. This can be accomplished by resizing, colour conversion, removing unwanted noise, or a combination of several of these techniques from the original image. The output of this process can greatly affect the accuracy with a good selection of preprocessing techniques. Image preprocessing techniques can be broadly classified into image enhancement and image restoration. Image enhancement techniques include Histogram Equalization (HE), Adaptive Histogram Equalization (AHE), Contrast Limited Adaptive Histogram Equalization (CLAHE) and logarithmic transformation. Image restoration includes median filter, mean filter, Gaussian filter, adaptive filter and wiener filter. Fig. 6 shows a detailed outline of the image preprocessing techniques.
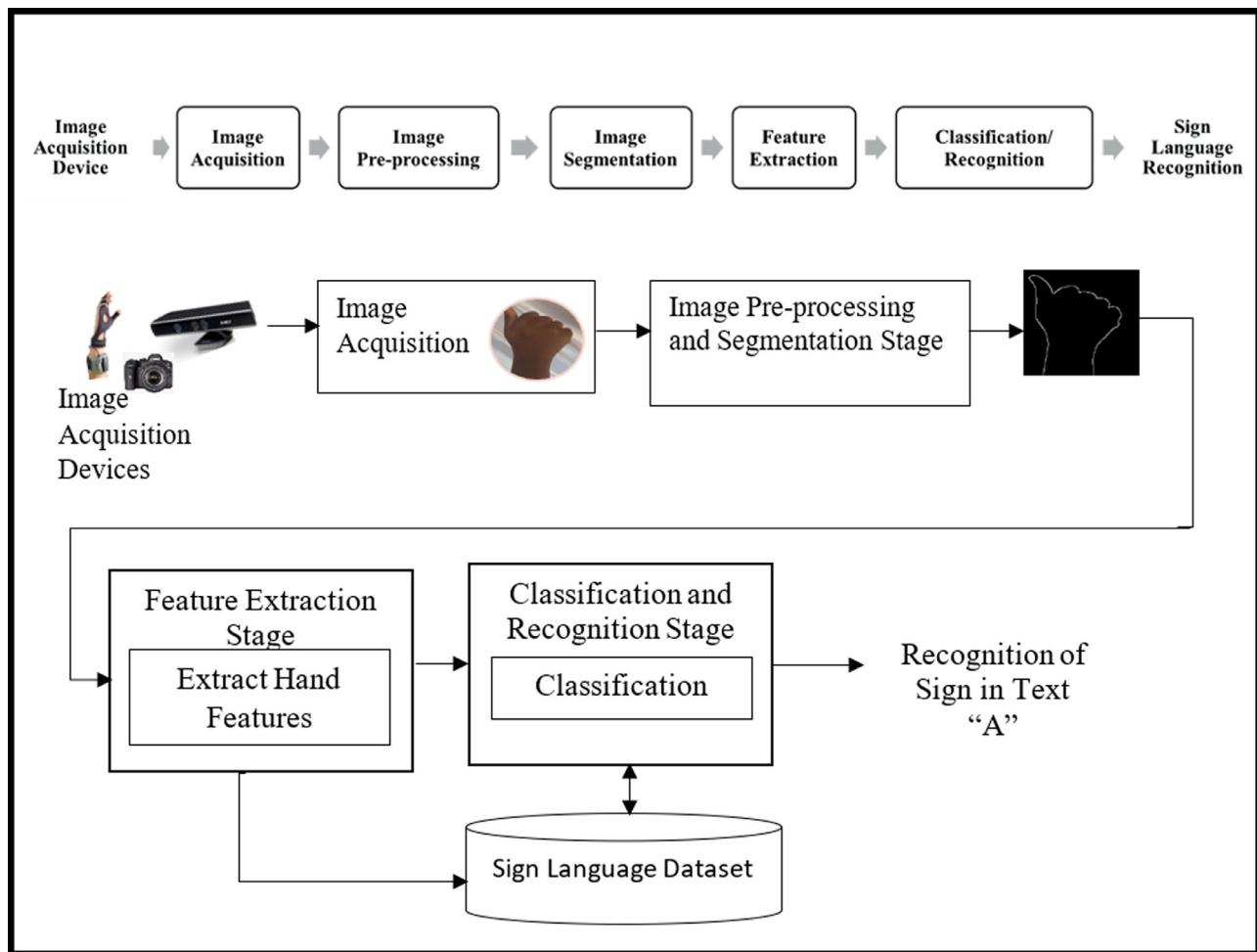
Fig. 5. Architecture of vision-based sign language recognition.

Table 3
Advantages and disadvantages of using different acquisition device controllers.

Image Acquisition device	Acquirement procedure	Advantages	Disadvantages
Camera/webcam	The device is used to capture or record the image/video from a fixed camera.	The device does not require wearing other external devices, and users only need to use their hands within the camera collection range. Low cost. Convenient and easy to use.	It is greatly affected by environmental factors such as light, skin colour, and occlusion. It requires several image processing techniques which might affect the recognition accuracy.
Data glove	The glove is attached to the sensors to extracts features needed for the recognition of sign. The features can be finger bending, orientation of the hand, movement, rotation, and position. These are transformed into electrical signals to determine the hand posture.	These devices are not Affected by the external environment when collecting data. It provides an improved recognition accuracy. Extraction of features with sensor based is relatively easier.	It reduces the naturalness of interaction. Inconvenient to use by user. Very expensive
Kinect	Kinect operate by provides colour and depth with trajectory information in an image simultaneously.	It is useful for many human computer interaction applications. The depth of distance detection is limited.	It can be affected by lighting conditions, hand and face segmentation, complex background, and noise. Kinect not suitable for outdoor applications and sensitivity to sunlight.
Leap Motion Controller	It is a sensor-based device connect to personal computer which detects hand movement and converts the signals into computer commands for data acquisition.	It has high recognition accuracy and faster processing speed around 200 frames per second. It can detect and track hands, fingers, and finger-like objects.	Due to it highly sensitive, accuracy of the recognition might be affected with small movement in sign position.

Table 4

Comparison of image enhancement techniques.

Enhancement techniques	Advantages	Disadvantages
Histogram equalization (HE)	It's simple to implement and highly effective for grayscale images.	It may increase the contrast of background noise. It is difficult to distinguish between noise and the desired features. It changes the brightness of an image.
Adaptive Histogram Equalization (AHE)	It is suitable to enhance local contrast and edges in every region of an image. It outperforms the histogram equalization technique.	It has an adverse effect on desired output due to its noise-amplification behavior. It fails to retain the Brightness on the input image.
Contrast Limited Adaptive Histogram Equalization (CLAHE):	It has a reduced noise compared to AHE and HE. It provides local output response and avoids brightness saturation.	It produces an unsatisfactory result when the input image has an unbalanced contrast ratio and increased brightness.
Logarithmic transformation	It is used to reduce higher intensities pixel values into lower intensities pixel values.	Applying the technique to a higher pixel value will enhance the image more and cause loss of actual information in the image. It does not apply to all kinds of images.
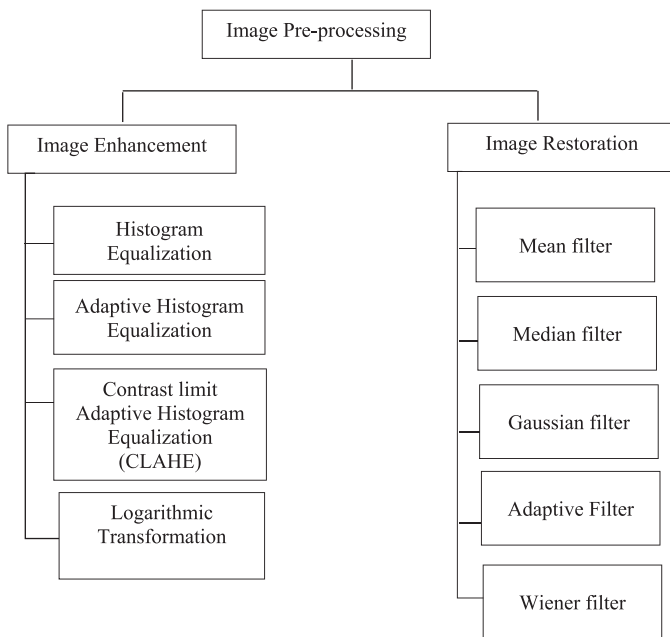
Fig. 6. Image preprocessing techniques.

3.2.1. Image enhancement techniques

In the aspect of image processing, image enhancement is one of the most challenging problems. It is an important process for restoring an image's visual appearance. The main goal of image enhancement is to improve input image quality, which can be better suited for human or machine analysis. The choice of technique depends on the area of application. These techniques can be used to refine the boundary and improve the accuracy of an input image (Majeed & Isa, 2021). Table 4 summary the advantages and disadvantages of various image enhancement techniques.

Histogram equalization (HE): Histogram equalization is used as an image preprocessing technique to strengthen the colour and increase the contrast (Gonzalez, 2002). As part of its operation, the histogram equalization technique remaps the grey level of a particular image based on its probability distribution.

This method is applied by distributing the grey value level in an image. It increases the lower limit of the range of colours to the darkest point and decreases the upper limit of the content of colours to the brightest point. Using this technique enhances the edges and boundaries of different images and reduces the images' local details (Verma & Dutta, 2017). In determine the histogram equalization, the probability density function (pdf) and cumulative distribution function (cdf) are computed using Eqs. (1) and (2), respectively (Kalhor et al., 2019):

$$\text{pdf}/p(A_k) = \frac{n_k}{n}; k = 0, 1, \ldots L - 1 \tag{1}$$

$$cdf/p(A_k) = \sum_{j=0}^{k} p(A_j) = \sum_{j=0}^{k} \left(\frac{n_k}{n} \right) \tag{2}$$

where n is the total number of pixels in a sample, k is the range of grey value, and n_k is the number of pixels with grey level with the grey level of A_k within the image A. The histogram equalization is operated on an image in three steps as stated by Bagade and Shandilya (2011):

i Formation of histogram.
ii New intensity values are calculated for each intensity levels.
iii Replace the previous Intensity values with the new intensity values.

Histogram equalization is the prevailing method for image enhancement, and it increases the contrast of the image, improving contrast and obtaining a uniform histogram.

It was used to improve the contrast of the input image in a different location and makes the brightness and illumination of the image uniform (Mahmud et al., 2019; Nelson et al., 2019; Sethi et al., 2012; Suharjito et al., 2019).

Adaptive Histogram Equalization (AHE): Adaptive Histogram Equalization (AHE) is a well-known and efficient algorithm for improving image contrast. However, it is time-consuming and computationally expensive. It has been widely applied to the various image processing application. Adaptive histogram equalization varies from ordinary histogram equalization. T adaptive approach calculates many histograms, each corresponding to a different section of the image, and uses these to redistribute the image's lightness values (Sund & Møystad, 2016). It is suitable for improving local contrast and edge in every region of an image but tends to increase the noise in relative homogenous regions (Sudhakar, 2017). Adaptive histogram equalization is a contrast enhancement method applicable to grayscale and colour images with very good efficiency. The steps in the AHE algorithm is given (Longkumer et al., 2014) as:

Step 1: Start the program.

Step 2: Obtain all the input images with the number of regions, dynamic range and clip limit.

Step 3: Pre-process the input image.

Step 4: Process each contextual region producing grey level mapping.

Step 5: Interpolate grey level mapping to assemble the final image.

Step 6: Stop.

Adaptive histogram equalization is used to improve local contrast and edge in every image region (Kaluri & Reddy, 2017; Kamal et al., 2019; Majeed & Isa, 2021; Nelson et al., 2019).

Contrast limit adaptive histogram equalization (CLAHE): Contrast Limited Adaptive Histogram Equalization (CLAHE) provides an improvement over adaptive histogram equalization, and it operates on small regions in the image rather than the entire image. The enhancement function is applied over all neighbourhood pixels, and the transformation function is derived. To achieve CLAHE technique, the image is divided into the contextual region known as tiles, while histogram equalization is applied to each tile to obtain a desired output histogram distribution (Aurangzeb et al., 2021). The CLAHE algorithm is given as (Rubini & Pavithra, 2019):

Algorithm for CLAHEs

Step 1: Read the input image.

Step 2: Apply mean and median filter on the input image.

Step 3: Find the frequency counts for each pixel value.

Step 4: Determine the probability of each occurrence using the probability function.

Step 5: Calculate the cumulative distribution probability for each pixel value.

Step 6: Perform equalization mapping for all pixels.

Step 7: Display the enhanced image.

Suharjito et al. (2019) used contrast limit adaptive histogram equalization to enhance image edges and skin detection from the images. The contrast and brightness are enhanced such that the original information is not lost, and the brightness is preserved (Sykora et al., 2014). Three image enhancement techniques, namely CLAHE, HE and AHE, were compared by Nelson et al. (2019).

Logarithmic transformation: Logarithmic transformation is used when the grey level input values are huge. The technique is used to reduce the skewness of highly skewed distributions in an image by spread out the dark pixels of the image while compressing the higher values (Mahmud et al., 2019). To achieve the transformation technique, an input image is first converted to grayscale before performing a logarithmic transformation so that the transformation's effects can be seen (Chourasiya et al., 2019; Maini & Aggarwal, 2010). Logarithmic transformation equation is given as:

$$s = c \log (1 + r) \tag{3}$$

where s is the logarithmic transformation, c is the factor by which the image is enhanced (it is usually set as 1), and r is the current pixel in the image. The comparison of image enhancement techniques in terms of merits and demerits is presented in Table 4.

Other preprocessing techniques: Other preprocessing techniques include image scaling and grayscale conversion, which are also referred to as image resizing. In the image preprocessing technique, they have an important role in enlarging and reducing the given image size in a pixel format (Perumal & Velmurugan, 2018). Image resizing is one of the basic operations for image preprocessing. It helps gain a lot of importance in improving low-resolution images and can be used to resample the image to decrease or increase the resolution (Fadnavis, 2014). The input images are scaled or cropped into a uniform size since the input images might be of different sizes collected from different sources. The resulting output of this technique produces either reduce or enlarge the size of an input image. The research community has used this technique to reduce the computational time and storage size (Wang et al., 2010; Jin et al., 2016; Islam et al., 2017; Ramos et al., 2019).

Grayscale conversion is one of the preprocessing techniques. It is one of the simplest enhancement techniques used in image processing. It is done by converting the colour space image such as Red, Green and Blue (RGB) to a grayscale image. This colour model consists of grey tones of colours, which have only 256 grey colours. It is composed exclusively of shades of grey, varying from black at the weakest intensity to white at the strongest (Buyuksahin, 2014). The advantages of using grayscale image over RGB coloured image include its simplified algorithm and reduces computational requirements while preserving the salient features of the colour images. However, a grayscale image has the advantage of losing colour information in an image (Güneş et al., 2016). The equation for the conversion of RGB colour model into a weighted grayscale image is given (Biswas et al., 2011) in Eq. (4):

$$GY = 0.56\,G + 0.33\,R + 0.11\,B \tag{4}$$

where GY denote the resulting grey level (grayscale) for the computed pixel, R is red, G is Green and B is Blue in a given image.

In Mekala et al. (2011), Karami et al. (2011), Oyewumi Jimoh et al. (2018) and Sharma et al. (2020a, 2020b), the colour image was converted into a grayscale image of two dimensions with two possible intensity values of white and black.

3.2.2. Image restoration

Image restoration is the process of restoring a degraded image caused by noise and blur (Reeves, 2014). The image restoration process is determined by the type of noise and corruption present in the image. Denoising techniques must be chosen based on how much noise an image contains. There are various image restoration techniques available to remove unwanted noise and blur in the image, these include; median filter, mean filter, Gaussian filter, adaptive filter and wiener filter.

Mean filter: Mean filter is a spatial filtering method based on sliding windows. It is used for image smoothing and reducing or eliminating noise in an image. The mean filter technique finds the value of the center pixel in the window by averaging the values of the neighbouring pixels (Aksoy & Salman, 2020). This filtering technique performs well with salt and pepper noise and Gaussian noise (Singh & Shree, 2016). The equation used to compute the mean filter is given as:

$$A[i,\ j] = \frac{1}{M} \sum_{k=i=1}^{i} \sum_{i=j=1}^{j+1} fk,\ l \tag{5}$$

where M is a value of the number of pixels used in the calculation, k and l are values that represent the location of these pixels. $A[i,\ j]$ is the mean filter at any point of the image by shifting the window on neighbour pixels. Mean filter was used by Kasmin et al. (2020) to remove noise from the sign images. There are limited numbers of researchers who have used mean filters in sign language recognition. This review work finds out that most of the research papers used median filter and Gaussian filter in the preprocessing stage to remove noise in the image, as reported in the review work (Cheok et al., 2019).

Median filter: The median filter is a non-linear method, which is most commonly used as a simple way to reduce noise in an image while preserving edges (Dhanushree et al., 2019). It achieves better results in removing salt and pepper noise. The pixel being considered is replaced by the median pixel value of the neighbouring pixel calculated from the pixel values of all the surrounding neighbouring pixels sorted in numerical order and then replacing the pixel being considered with the middle pixel value

(Ahmad et al., 2019). The average of the two middle pixel values is used when the neighbourhood under consideration contains an even number of pixels. Its performances are better than the mean filter in preserving edges and make it sharper while removing noise in the image and easy to implement.

The arithmetic means filter equation used is given as (Gonzalez, 2002):

$$\hat{f}_{(x,y)} = \frac{1}{mn} \sum_{(s,t) \in S_{xy}} g(s,t) \tag{6}$$

where S_{xy} represents the set of coordinates in a rectangular subimage window (kernel size) of $m \times n$ centered at any point (x,y) in the original image, s and t are the rows and column coordinates of the pixels whose coordinates are members of the set S_{xy}.

The technique is very good only for removing salt and pepper noise. It was used to reduce noise for the image and preserve edges by various researchers (Islam et al., 2017; Lahiani et al., 2016; Pansare et al., 2012; Pansare & Ingle, 2016; Zhang et al., 2011).

Gaussian filter: A Gaussian filter is a linear filter and a nonuniform low pass filter resulting from blurring an image by a Gaussian function. It is the most used to reduce noise and smoothen filter edges in an image (Basu, 2002). The Gaussian filter is generally utilized as a smoother and widely employed in image preprocessing to reduce the image noise by blurring an image. It acts as a smoothing operator and is mainly used by several researchers in sign language recognition. Gaussian blur is a filter used to smooth or remove blurs a digital image (Umamaheswari & Karthikeyan, 2019). Gaussian filter was used to remove noise and image smoothening in (Oliveira et al., 2017; Pansare et al., 2012; Yusnita et al., 2017). Equation of two-dimension (2D) Gaussian filter is given by:

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{7}$$

where $G(x,y)$ denotes gaussian filter value, x and y denote row and columns values, and σ is the standard deviation of the gaussian distribution.

Adaptive Filter: An adaptive filter is applied to the noisy image to remove noise from the image while detailed information about the image is retained. It preserves edges and other high-frequency parts of an image more than a similar linear filter. The mean and variance are the two statistical measures used to determine adaptive filters. The algorithm used to achieve adaptive filter is given as (Kaluri and Reddy, 2016a, 2016b):

Step 1: Read the input colour image.
Step 2: Convert the input colour image to a grayscale image.
Step 3: Increase Gaussian noise to the image so that the image size is large.
Step 4: Eliminate noise from the image by using wiener2 function $K = weiner2$ (J, [x.x])

An adaptive filter was used to remove noise from the input sign image in Kaluri and Reddy (2016a, 2016b).

Wiener filter: Wiener filter is primarily used to remove noise from the image and minimizes the mean square error (MSE) between the estimated random process and the desired process. It optimizes the trade-off between smoothing the image discontinuities and removal of the noise (Tania & Rowaida, 2016). The technique blurs image significantly due to the use of a fixed filter throughout the entire image. The restoration function of wiener filter includes both the degradation function and the statistical characteristic of noise. It uses a high-pass filter for deconvolution and a low-pass filter for noise reduction during compression (Maru & Parikh, 2017). This technique can be used to remove various kind of noise such as salt and pepper, gaussian and speckle noise in from

the image (Kaur, 2015). The equation to compute wiener filter is given as:

$$G(u,v) = \frac{H^*(u,v)P_s(u,v)}{|H(u,v)|^2 P_s(u,v) + P_n(u,v)} \tag{8}$$

where u and v are the values, represent the location of these pixels, $H^*(u,v)$ denote degradation function, $H^*(u,v)$ represent complex conjugate of degradation function, $P_n(u,v)$ is the power spectral density of noise, $P_s(u,v)$ is the power spectral density of the original image.

Various filtering techniques have been used in sign language recognition. A Winer filter was used to eliminate noise from the sign image (Kaluri & Reddy, 2017). Some researchers combined two filtering techniques in other to remove noise from the image. Pansare et al. (2012) combined both median and Gaussian filters to remove noise and to smooth the input image, respectively. Table 5 summary different image filtering techniques with their advantages and disadvantages.

3.3. Image segmentation techniques

Image segmentation is the process of partitioning an image into meaningful regions called segments (Egmont-Petersen et al., 2002). The images are segmented to obtain the region of interest. There are two basic approaches used for segmentation; contextual and non-contextual segmentation (Enikeev & Mustafina, 2020; Jin et al., 2016; Al-Shamayleh et al., 2020). In contextual segmentation, it exploits the relationships between the image features, such as edges, similar intensities and spatial proximity. A non-contextual segmentation ignores spatial relationships between image features but groups pixels based on global attributes value (Pal & Pal, 1993; Sharma et al., 2021). Image segmentation techniques are classified into edge detection-based, thresholding, region-based, clustering-based, and artificial neural network-based. A detailed outline of the image segmentation techniques is shown in Fig. 7.

3.3.1. Thresholding techniques

Thresholding is the simplest and commonly used segmentation technique to separate objects from the background (Lee et al., 1990; Cheng et al., 2002; Dong et al., 2008; Xu et al., 2013). The techniques separate the image pixels according to their degree of strength and the range of values in which a pixel lies—examples of thresholding techniques are global thresholding, local adaptive thresholding and multilevel thresholding.

Global thresholding: The global thresholding technique uses a single threshold value for the whole image to separate foreground from background. This technique assumes that the image has a bimodal histogram. Thus the image can be extracted from the background using a simple operation that compares image values with a threshold T (Rogowska, 2009).

The threshold technique is defined by Kaur and Chand (2018) as:

$$g(x,y) = \begin{cases} 1, & if \ f(x,y) > T \\ 0, & if \ f(x,y) \leq T \end{cases} \tag{9}$$

where $g(x,y)$ is the output image of the original images. T is the threshold value constant for entire images.

Otsu thresholding technique is the most widely used global thresholding technique. The technique converts the multi-level image into a binary image through a given threshold value to separate the foreground from the image's background. It is based on discriminant analysis by maximizing the between-class variance of the Gray levels in the image and background portions. The weighted sum of the variance of the two classes is given as:

$$\sigma_w^2(t) = w_0(t)\sigma_0^2(t) + w_1(t)\sigma_1^2(t) \tag{10}$$

Table 5

Advantages and disadvantages of image filtering techniques.

Filter techniques	Advantages	Disadvantages
Mean filter	Easy to implement	A single wrongly represented pixel value can significantly impact the mean value of all pixels in their immediate neighbourhood. It blurs an edge when the filter neighbourhood crosses a boundary.
Median filter	It preserves thin edges and sharpness from an input image. Both of the problems of the mean filter are tackled by the median filter.	It is relatively expensive and Complex to compute. It is good only for removing salt and Pepper noise. It is less effective at removing the gaussian type of noise from the image.
Gaussian filter	It is effective for removing the gaussian type of noise. The weights give higher significance to pixels near the edge.	It has high computational time and sometimes removes edges details in an image.
Adaptive filter	It preserves edges and other high-frequency parts better than a similar linear filter.	It is computational complexity. There are still some visible distortions available in the image using an adaptive filter.
Weiner filter	It is a popular filter used for image restoration. It is not sensitive to noise. Suitable to exploit the statistical properties of the image. The small window size can be used to prevent blurring of edges	Prior knowledge of the power spectral density of the original image is unavailable in practice. It is comparatively slow to apply because it works in the frequency domain. The output image is very blurred.
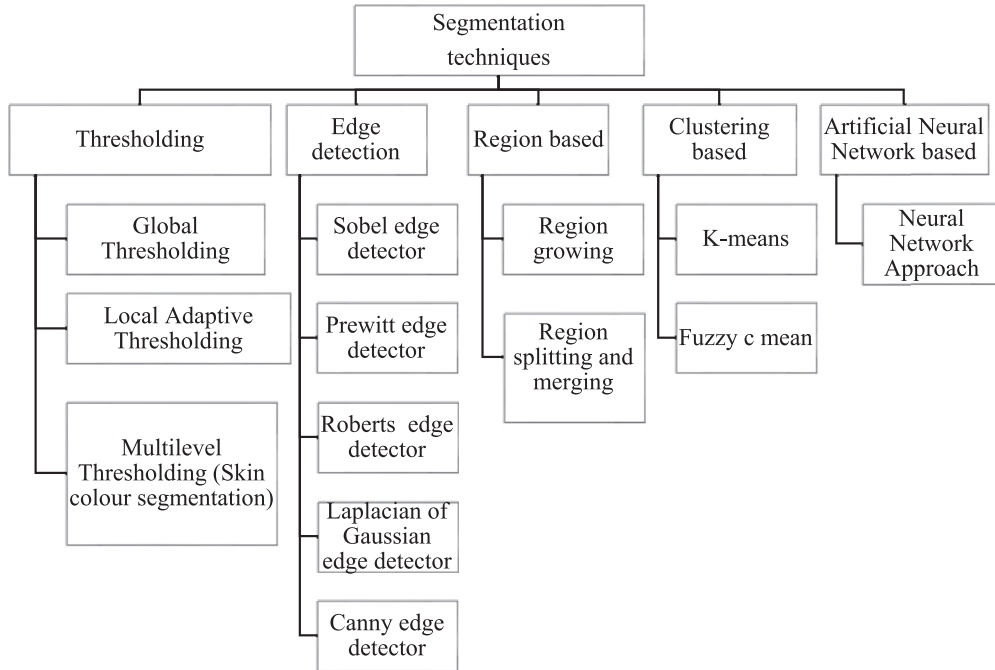
Fig. 7. Classification of image segmentation techniques.

where;

w_0 and w_1 are the probabilities of the two classes separated by a threshold t (with a value range from 0 to 255), σ_0^2 and σ_1^2 are variances of these two classes. The class probabilities w_0 and w_1 for each pixel value, i are computed from the bins of the histogram as:

$$w_0(t) = \sum_{i=1}^{t} p(i) \tag{11}$$

$$w_1(t) = \sum_{i=t+1}^{I} p(i) \tag{12}$$

where i is the maximum pixel value (255)

The total variance is given as:

$$\sigma_T^2 = \sigma_w^2(t) + \sigma_b^2(t) \tag{13}$$

where variance between-class is determined by:

$$\sigma_b^2(t) = w_0(t) w_1(t)[\mu_1(t) - \mu_2(t)]^2 \tag{14}$$

Research presented in Rahim et al. (2020), Pansare et al. (2015), Islam et al. (2017) and Tan et al. (2021) used Otsu's thresholding algorithm based on global thresholding to segment the hand region from its background region using the computed threshold value. Otsu's thresholding was fused with the canny edge and Discrete Wavelet Transforms (DWT) to segment the region of interest from the video sequence (Kishore & Kumar, 2012). Due to the use of a single threshold value, this technique does not produce effective segmentation regarding illumination over an entire image.

Local adaptive thresholding: Local adaptive thresholding is employed to address the problem of global thresholding based techniques by dividing an image into sub-images and calculating thresholds for each sub-image (Korzynska et al., 2013). This thresholding technique uses the mean value of the local intensity distribution or other statistics metrics, like mean plus standard deviation, to separate foreground from background in each sub-image (Senthilkumaran & Vaithegi, 2016). The most basic approach to

adaptive thresholding approach proposed by Niblack (1986), which local threshold is calculated based on the mean (m) and standard deviation (s) of the local neighbourhood of pixel w. Adaptive thresholding value based on Niblack's techniques is given as:

$$T_{niblack} = m + k * s \tag{15}$$

where $T_{niblack}$ is the adaptive threshold, m is the mean of the window of size w, s is the standard deviation, and k is the fixed value introduced depending on the remaining noise level in the image's background. The adaptive threshold technique helps separate images from varying backgrounds and extract tiny and sparse regions. The major drawback of adaptive thresholding is that it is computationally expensive than global thresholding. Adaptive thresholding was used to segment the region of interest in sign language recognition for further processing (Dudhal et al., 2019; Rao & Kishore, 2018).

Multilevel thresholding: Multilevel thresholding technique is employed to extract homogenous regions in an image. This technique determines multiple thresholds for the given image and divides the image into distinct regions. The method yields adequate results for images with coloured or complex backgrounds, where bilevel thresholding has failed. Skin colour segmentation is one of the approaches used for multilevel thresholding and is widely used in different areas of application, including human-computer interaction (HCI), image recognition, traffic control system, video surveillance and hand segmentation (Sallam et al., 2021; Lee et al., 2021a, 2021b; Razmjooy et al., 2021). The technique used the colour model to separate the skin region from the image in hand segmentation. The colour models used include Red, Blue, and Green (RGB) colour space, Hue Saturation Value (HSV) and Luminance (Y) and Chromaticity (CbCr) (YCbCr) colour space model. These colour models were extensively discussed in Garcia-Lamont et al. (2018) and Zarit et al. (1999). YCbCr and HSV are the most widely used skin colour segmentation techniques in sign language and the equations to transform the input RGB image into YCbCr using Eqs. (16) to (18).

$$H = \cos^{-1} \frac{\frac{1}{2}(2R - G - B)}{\sqrt{(R - G)^2 - (R - B)(G - B)}} \tag{16}$$

$$S = \frac{\max(R, G, B) - \min(R, G, B)}{\max(R, G, B)} \tag{17}$$

$$V = \max(R, G, B) \tag{18}$$

and that of the HSV colour model using Eq. (19).

$$\begin{bmatrix} Y \\ C_b \\ Cr \end{bmatrix} = \begin{bmatrix} 0 \\ 128 \\ 128 \end{bmatrix} + \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.169 & -0.331 & 0.500 \\ 0.500 & -0.419 & -0.081 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{19}$$

The value of H, S, V, Y, Cb and Cr is determined and used as threshold values to obtain the required colour model.

The RGB colour space is the least preferred for colour-based detection and colour analysis because it is challenging to identify and establish human skin from RGB colour tone due to the variation in human skin (Shaik et al., 2015). The HSV model is an effective mechanism for determining human skin based on hue and saturation. The other efficient models are YUV and YIQ colour space models (Tabassum et al., 2010).

Research in Hartanto et al. (2014) and Huong et al. (2016) converted RGB colour space into HSV colour space for identifying skin regions. Tariq et al. (2012) transformed the input RGB video into to YCbCr colour model. The value obtained against each frame pixel is compared with a specific threshold value that isolates the hand region from the whole image. The YCbCr colour model was hybridised with the Gaussian Mixture Model (GMM) and Morphological operation to obtain skin colour region in Pan et al. (2016) re-

search. Athira et al. (2019) converted single-handed dynamic gestures from the input video to the YCbCr model and later eliminated the face region with only the hand region. The efficiency of CIELAB colour space was studied by Mahmud et al. (2019). Similar research conducted by Shaik et al. (2015) and Suharjito et al. (2019) found that YCbCr is more robust and gave more accurate results than HSV and other skin colour segmentation techniques in different illumination conditions.

3.3.2. Edge detection techniques

The edge detection approach is one of the essential image processing techniques. The technique relies on the quick change in intensity value in an image. The Edge detection algorithms detect edges where either the first derivative of intensity is larger than a certain threshold or the second derivative contains zero crossings. A good edge-based segmentation requires three critical steps: detecting edges, eliminating irrelevant edges, and joining the edges. Edge detection techniques reviewed in this paper are Robert edge detector, Sobel edge detector, Prewitt edge detector, Laplacian of Gaussian edge detector and Canny edge detector. Various edge detector techniques were discussed extensively in Bhardwaj and Mittal (2012); Chinu & Chhabra, 2014; Maini & Aggarwal, 2009; Rashmi and Saxena, 2013; Shrivakshan & Chandrasekar, 2012).

Robert edge detector: Robert edge detector is a gradient based operator that computes the sum of the squares of the difference between diagonally adjacent pixels through discrete differentiation and then calculates approximate gradient of the image. The input image is convolved with 2×2 kernels operator with gradient magnitude and directions computed (Maini & Aggarwal, 2009). To obtain gradient component in each direction (Gx and Gy), 2×2 kernels operator are applied independently to the input image. The equation used to compute gradient magnitude $|G|$ and directions θ are given as:

$$|G| = \sqrt{G_x{}^2 + G_y{}^2} \tag{20}$$

$$\theta = \tan^{-1}\left(\frac{G_x}{G_y}\right) - \frac{3\pi}{4} \tag{21}$$

where G_x and G_y are the gradients in the x and y directions, respectively.

Sobel edge detector: The Sobel edge detector is a discrete differentiation operator of the first-order derivative used to approximate the gradient of image intensity function for edge detection. The technique convolves the input image with a kernel size of 3×3 to obtain gradient magnitude and emphasises pixels closer to the kernel's center. This technique is sensitive to noise compared with the Robert edge technique but has a fast computation time due to kernel size (Rashmi and Saxena, 2013; Sujatha & Sudha, 2015). The direction of the gradient is calculated using Eq. (22). The direction of the gradient for θ Sobel edge detector is given as:

$$\theta = \tan^{-1}\left(\frac{G_x}{G_y}\right) \tag{22}$$

Prewitt edge detector: Prewitt edge detector operation is very similar to the Sobel edge detector with a kernel of 3×3 and is widely used to detect an image's vertical and horizontal edges (Maini & Aggarwal, 2009). In the Prewitt edge detector, the maximum response of all eight kernels for a pixel location is used to calculate the local edge gradient magnitude. At the same time, emphases are not placed on pixels closer to the kernel's center (Shrivakshan & Chandrasekar, 2012). The technique is less computational expensive but detects many false edges during detection, which results in a noisy image. It is susceptible to noise and more effective with noiseless images (Muthukrishnan & Radha, 2011).

The equation to compute the gradient magnitude of the Prewitt edge detector is given as (Sujatha & Sudha, 2015):

$$|G| = \max\big(|G|_i : i = 1 \; to \; n\big) \tag{23}$$

where $|G|_i$ is the kernel i response at the particular pixel position, and n is the number of convolution kernels.

Laplacian of Gaussian (LoG) Edge Detector: This technique uses second-order derivatives of pixel intensities to locate edges in an image. An image's Laplacian is used to highlight areas of rapid intensity change in order to detect edges. Before applying the Laplacian function, the image is subjected to a Gaussian smoothing filter to reduce noise levels. It takes a single grey level image as input and produces another grey level image as output (Maini & Aggarwal, 2009).

The Laplacian L(*x,y*) of an image with pixel intensity values I(*x,y*) is given as:

$$L(x, y) = \frac{\partial^2 I}{\partial x^2} + \frac{\partial^2 I}{\partial y^2} \tag{24}$$

where I is the image intensity, x, y are the coordinates of the point.

The Gaussian smoothing filter LoG used to reduce noise in an image before the Laplacian technique is applied and is given as:

$$LoG(x, \; y) = -\frac{1}{\pi \sigma^4}\left[1 - \frac{x^2 + y^2}{2\sigma^2}\right]e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{25}$$

where σ denote Gaussian standard deviation.

The standard deviation of the Gaussian smoothing filter employed in the LoG filter dramatically influences the behavior of the LoG edge detector. An increase in the value of σ resulting in a wider Gaussian filter and more smoothing, which may be difficult to distinguish edges in an image.

Canny edge detector: The Canny edge detection technique (Canny, 1986) has been suggested as an optimal edge detector (Aliyu et al., 2020). It is one of the standard edge detection techniques and was first created by John Canny at MIT in 1983 (Muthukrishnan & Radha, 2011). The goal of this technique is to detect edges in an image while simultaneously suppressing noise. It finds edges by separating noise from the image before find edges of the image. The basic algorithm used to detect Canny edge is given as:

Step 1: Read the input image.

Step 2: Smooth the image with a Gaussian filter of chosen kernel size to reduce noise and unwanted details (sometimes median filter can be used because it preserved edges more than gaussian filter).

Step 3: The gradient of the image is used to determine edge strength. For this, a Robert mask or a Sobel mask can be utilized. The equation used to compute the magnitude of the gradient $|G|$ is given as:

$$|G| = \sqrt{G_x{}^2 + G_y{}^2} \tag{26}$$

G_x and G_y are the gradients in the x and y directions, respectively.

Step 4: Find the edge direction by using the gradient in the x and y directions.

The direction of the gradient θ for Robert edge detector is given as:

$$\theta = \tan^{-1}\left(\frac{G_x}{G_y}\right) - \frac{3\pi}{4} \tag{27}$$

Step 5: Computed edge directions are resolved into horizontal, positive, vertical, negative directions.

Step 6: Apply non-maxima suppression to trace the edge in the edge direction and suppress any pixel value below the set value (that is not considered to be an edge). This gives a thin line in the output image.

Step 7: Hysteresis thresholding is applied to eliminate streaking.

Different researchers have used various edge detection techniques in sign language recognition. Canny edge performs better than many edge detection techniques that have been developed. It is an essential and better method without disturbing the features of the edges in the image afterwards. However, the method is faced with challenges of extracted noise with edges, and sometimes disjoint edges were obtained by manually selecting edges when working with multiple images. Lionnie et al. (2012) employed Sobel edge detection in their study. They compared the performance of Sobel edge detection with skin colour segmentation in HSI colour space, low pass filtering, histogram equalization, and desaturation. The desaturation technique obtained the best performance, which converts the image into grayscale to remove the chromatic channel and preserve only the intensity channel. Jayashree et al. (2012) used Sobel edge detection to extract the image's region of interest. Other research by Thepade et al. (2013) performed Sobel edge detection on image datasets. The result of the proposed system was compared with four other edge detection techniques, namely, canny, Robert, Prewitt and Laplace. The results show that Sobel performed better, but Canny edge detection had the best overall result compare with others. Also, Prasad et al. (2016a, 2016b) used canny edge with the discrete wavelet transform to detect boundary pixels of the sign video image. The technique helps extract satisfactory edges, preserves the hand, and makes the video image exactly. In recent research, canny edge detection was used to obtain the object of interest from the image and achieve optimal performance on overall accuracy (Jin et al., 2016; Mahmud et al., 2019; Singh et al., 2019).

3.3.3. Region-based techniques

Region-based techniques grouped the pixels exhibiting similarly closed boundaries based on predefined criteria (Garcia-Lamont et al., 2018). This technique is also known as Similarity-Based Segmentation (Yogamangalam & Karthikeyan, 2013) and requires the use of appropriate thresholding techniques to group its similarities. The similarity between pixels can either be in the form of intensity, colour, shape or texture. Region-Based techniques are further classified into Region growing and Region splitting and merging methods (Divya & Ganesh Babu, 2020; Kaur' & Kaur, 2014; Yogamangalam & Karthikeyan, 2013).

Region growing technique: The region growing technique clusters the pixels that represent similar areas in an image. This is done by grouping pixels whose properties, such as intensity, colour and shape, differ by less than some specified amount. Each grown region is assigned a unique integer label in the output image. Region growing is capable of correctly segmenting regions that have the same properties and are spatially separated. However, it is time-consuming and sometimes produces undesirable results. The output of region growth depends strongly on the selection of a similar criterion. If it is not chosen correctly, the regions leak into adjoining areas or merge with areas that do not belong to the object of interest (Rogowska, 2009). Region growing technique was extensively reviewed by Ikonomakis et al. (2000), Mehnert and Jackway (1997) and Verma et al. (2011) Assuming $p(x, y)$ is the original image that is to be segmented, $s(x, y)$ is the binary image where the seeds are located and T is any predicate which is to be tested for each (x, y) location. The region growing algorithm based on 8-connectivity is given by Kaur' and Kaur (2014) as follows:

Step 1: All the connected components of s are eroded.

Step 2: Compute a binary image P_T where P_T (x, y) = 1, *if* T(x, y) = True

Step 3: Compute a binary image q, where q(x, y) = 1, if P_T (x, y) = 1 and (x, y) is 8-connected to seed in s.

The connected components in q are segmented regions used as the output of the region growing technique.

Region splitting and merging technique: Region splitting and merging based segmentation technique uses two basic approaches; splitting and merging for segmenting an image into various regions. Splitting is done by iteratively dividing an image into maximum regions having similar characteristics, and merging contributes to combining the similar adjacent regions to form an excellent segmented image of the original image (Kaur' & Kaur, 2014). A combination of these two techniques gives better performance. The basic algorithm for region growing and merging is given as follows:

Let p be the original image, R_i represent subdivide region and *T* be the particular predicate.

Step 1: All the R_i is equal to *p*.

Step 2: Each region is divided into quadrants for which T (R_i) = False.

Step 3: If for every region, T (R_j) = True, then merge adjacent regions R_i and R_j such that T R_i U R_j = True.

Step 4: Repeat step 3 until merging is impossible.

Ikonomakis et al. (2000) summarise the procedure for the region as follows:

i Split into four disjointed quadrants any region where a homogeneity criterion does not hold.

ii Merge any adjacent regions that satisfy a homogeneity criterion.

iii Stop when no further merging or splitting is possible.

3.3.4. Clustering based segmentation techniques

Clustering-based segmentation is an unsupervised learning technique that divides a set of elements into uniform groups. It is highly used for the segmentation of images into clusters having pixels with similar characteristics. Several clustering-based segmentations exists. The two major techniques most widely used include K-means and Fuzzy C-means clustering (Cebeci & Yildiz, 2015; Ghosh & Dubey, 2013; Khan, 2013).

K means technique: K means algorithm is a clustering-based technique, and it has been extensively reviewed in Duggirala (2020) and Panwar et al. (2016). It takes the distance from the data point to the prototype as the primary function of optimization. The adjustment rules of iterative operation are obtained by the method of finding extreme values of functions. The K-means algorithm makes use of Euclidean distance as the similarity measure that finds the optimal classification of an initial cluster center vector so that the evaluation index is minimum. The error square sum criterion function is used as a clustering criterion function. Although the algorithm of K means is efficient, the value of K should be given in advance, and the selection of K value is very difficult to estimate (Zheng et al., 2018).

K means algorithm by Ghosh and Dubey (2013) is given as follows:

Step 1: Set desired clusters, K and Initialize to choose k starting points which are used as initial estimates of the cluster centroids.

Step 2: To classify all the data or images:

(a) Calculate the distance between clusters centroids and the data.

(b) Move the data closer to the cluster that has less distance as compared to others.

Step 3: Centroid calculation

– When each point in the dataset is assigned to a cluster, it is needed to recalculate the new k centroids.

Step 3: Convergence criteria

– Step (2) to step (3) are repeated until no point changes its cluster assignment or until the centroids no longer move.

Fuzzy C means (FCM) Technique: Fuzzy C means (FCM) technique assigns membership levels and uses them to assign data elements to one or more clusters. It provides a more precise computation of the cluster membership and has been used successfully for many image clustering applications (Dhanachandra & Chanu, 2017). FCM is an iterative approach that generates a fuzzy partition matrix and requires a cluster center and an objective function. The cluster center and objective function values are updated for every iteration, and the process is terminated when the difference between two successive object function values is smaller than a predetermined threshold value (Khan, 2013). The major difference with the K means algorithm is that instead of making a hard decision about clusters, which cluster belongs, it assigns a value between 0 and 1 that describes every cluster. It requires much computation time, unlike K-means clustering that produces nearly the same results with less computation time. The FCM algorithm is given as follows (Khan, 2013):

Step 1: Assign the values for c (number of clusters; $2 = c < n$), q (weighting exponent of each fuzzy member) and threshold value ε

Step 2: Initialize the partition matrix $U = [v_{ik}]$ degree of membership of x_k in i^{th} *cluster* .

Step 3: Initialize the cluster centers and a counter p.

Step 4: Calculate the membership values and store them in an array.

Step 5: For each iteration, calculate the parameters a_i^p and b_i^p till all pixels are processed, where

$$a_i^p = a_i^p + v_i x_k \tag{28}$$

$$b_i^p = a_i^p + v_i \tag{29}$$

Step 6: After each iteration, update the cluster center and compare it with the previous value $(U^b - U^{b-1})$

Step 7: If the comparison difference is less than the defined threshold value, stop iteration else and repeat the procedure.

3.3.5. Artificial neural network based segmentation technique

Artificial neural network-based segmentation techniques replicated the learning mechanisms of the human brain. It is made up of a large number of connected nodes, each with its weight, and the neuron corresponds to a pixel in an image (Khan, 2014). The image is trained as a neural network using training samples, and then the connections between neurons with pixels are created. The newly created images are segmented from the trained image. There are two essential steps to this technique; extracting features relevant to the image and segmentation using a neural network. The technique has performed very well in a difficult image segmentation problem. Backpropagation neural network (BPNN), Feedforward neural network (FFNN), Multilayer feedforward neural network (MLFF), Multilayer Perceptron (MLP), Convolution neural network (CNN) (Kanezaki, 2018) and Self organization Map (SOM),

Table 6
Summary of advantages and disadvantages of various segmentation techniques.

Segmentation Technique	Advantages	Disadvantages
Thresholding Method	It is a fast and straightforward approach. It does not require prior information to operate. It has a low computation cost.	It is highly dependent on peaks, while spatial details are not considered. Sensitive to noise. Selection of an optimal threshold value is difficult.
Edge Based Method	Suitable for images having better contrast between objects.	It is not suitable for images with too much noise or too many edges.
Region-Based Method	It is less susceptible to noise and more useful when defining similarity criteria is easy.	It is quite expensive in terms of computation time and memory consumption.
Clustering Method	It's more useful for real-world challenges due to the fuzzy partial membership employed.	Determining membership functions is not easy.
Artificial Neural Network-Based Method	It does not require a complex program to work. Less prone to noise.	Computational time in training is higher.

are among the most often used neural networks for image segmentation (Amza, 2012; Kanezaki, 2018; Moghaddam & Soltanian-Zadeh, 2011). The algorithm used to separate the region of interest from its background was proposed in An and Liu (2019) and Zhao et al. (2010). In a neural network, an image can be segmented based on pixel classification or edge detection. Sections 3.3.1 to 3.3.5 explained various segmentation techniques, including thresholding, edge-based, region-based, clustering-based, and artificial neural network-based. Table 6 summary the advantages and advantages of the segmentation techniques.

Table 7 illustrates a summary of the vision-based SLR systems. Presented in the table are data acquisition devices, data preprocessing techniques, and segmentation techniques used.

3.4. Feature extraction techniques

Feature extraction is a technique used to obtain the most relevant features from the input image. It aims at finding the most distinctive features in the acquired image (Patil & Sinha, 2017). It is a form of dimensional reduction that effectively represents the interesting parts of an image. The compact feature vector is extracted by removing an irrelevant part to increase learning accuracy and enhance the result's visibility (Khalid et al., 2014; Kumar & Bhatia, 2014). The feature extraction output supports the classification stage by checking for features that can effectively be distinguished between classes and help achieve high recognition accuracy. The features extracted from the interested region are characterised into colour, texture and shape features (Patel & Gamit, 2016). The important feature extraction techniques used in SLR that have achieved a good performance include principal component analysis (PCA), Fourier descriptor (FD), histogram of oriented gradient (HOG), shift-invariant feature transform (SIFT), and speed up robust feature (SURF).

3.4.1. Principal component analysis (PCA)

Principal Component Analysis (PCA) is a technique widely used to extract features in image processing (Aliyu et al., 2020). PCA requires a statistical procedure that uses an orthogonal transformation to transform a series of observations of potentially correlated variables into a set of values for non-correlated variables (Kumar & Bhatia, 2014). It operates by calculating new variables called principal components and use the new variable to create a linear combination of the initial variables. The PCA operation is as follows: first, the principal components produce the greatest potential variance, while the larger portion is extracted. This is followed by the computation of the eigenvectors and corresponding eigenvalues. The computed eigenvectors are stored by decreasing the order of

the eigenvalues, which form the dimensionality reduction of data in PCA. The algorithm used to calculate PCA is given as follows (Cheok et al., 2019; Karamizadeh et al., 2013):

Step 1: Column or row of vector size represents a given training set of images, M with an *S*-dimensional vector.

Step 2: The mean, μ of all images in the training set is computed using Eq. (30):

$$\mu = \frac{1}{M} \sum_{i=1}^{M} x_i \tag{30}$$

with x_i as the *i*th image through its columns concatenated in a vector.

Step 3: PCA basis vectors which are eigenvectors of the Scatter matrix, S_T are computed using Eq. (31):

$$S_T = \sum_{I=1}^{M} (x_i - \mu) . (x_i - \mu)^2 \tag{31}$$

Step 4: The eigenvectors and corresponding eigenvalues were calculated. The eigenvectors values are stored by decreasing eigenvalues order. The eigenvectors with lower eigenvalues contain less information on the distribution of data. These are filtered to reduce the dimensionality of data.

Huong et al. (2016) employed PCA to extract the features needed to recognise 25 Vietnamese sign language (VSL) alphabets under uniform background. Research in Zaki and Shaheen (2011) introduced PCA with Kurtosis position and Motion chain code (MCC). Kurtosis was fused with PCA to calculate the edges denote articulation, and MCC extracted the vectors representing the hand movement. The research finding shows that combining these three feature extraction techniques improved recognition accuracy by 89.90% compared to separate use or combination of two methods. In a similar study presented by Li et al. (2016), PCA was combined with the entropy-based K-means algorithm and applied to Hidden Markov Model (HMM). The input image used for feature extraction was acquired from glove-based data using an Attitude Heading Reference System (AHRS) sensor. Simultaneously, the noise was removed from an unclear image using a low pass filter (LPF). The developed model demonstrated better performance than the result from the Kalman filter. Prasad et al. (2016a, 2016b) proposed a study that combined PCA and Elliptical Fourier Descriptors for feature extraction on a video dataset. The Elliptical Fourier Descriptors was used to optimize and preserve the shape details without any changes in rotation. Simultaneously, using PCA, the feature vector output for a given sign from multiple frames is treated to form a single vector.

Table 7

Summary of vision-based sign language recognition architecture.

References	Data Acquisition Device	Data Acquired/ Feature set	Dataset Collection	Preprocessing Technique	Segmentation Technique
Kishore and Kumar (2012)	Camera	Hand gesture and head movement.	Self-acquired	Grayscale image and Gaussian filtering.	Otsu's thresholding and discrete wavelet transform with canny edge detection.
Pansare et al. (2012)	Camera	Handshapes.	Self-acquired	Grayscale image, Median filter, Gaussian filtering and Morphological operations.	Thresholding, blob and crop, and Sobel edge detection.
Thepade et al. (2013)	Camera	Manual (handshapes).	Self-acquired	-	Sobel edge detection compared with (canny, Robert, Prewitt and Laplace).
Jin et al. (2016)	Camera	Handshapes	Self-acquired	RGB colour image resized to the resolution of 320×240 pixels.	Canny edge detector and seeded region growing.
Carneiro et al. (2017)	Kinect	Hand gestures	Self-acquired	Mean filter, Grayscale, Contrast stretch.	Background subtraction.
Prasad et al., 2016)	Camera	Handshapes and head position	Self-acquired	Grayscale, Morphological operations and average filter.	Canny edge with the discrete wavelet transform (DWT).
Raheja et al. (2016)	Kinect and Camera	Handshapes	Self-acquired	Skin filter	Skin colour (HSV colour space).
Islam et al. (2017)	Camera	Handshapes	Self-acquired	-	Otsu's method, Median filtering and Morphological.
Mahmud et al. (2019)	Camera	Handshapes	Public	Logarithmic Transformation and Histogram equalization (HE).	CIELAB colour space and Canny Edge detection.

3.4.2. Fourier descriptor (FD)

Fourier descriptors are used to characterise shape complexity and have been used for identifying different sign shapes (Agrawal et al., 2012; Kishore et al., 2015). The Fourier transformed coefficients form the Fourier descriptors of the image and represent the shape in the frequency domain, lower and higher frequency descriptors (Cheok et al., 2019). The lower frequency descriptors contain information about the general features of the shape, while the higher frequency descriptors contain information about the relevant part of the shape.

Complex coordinate of the boundary pixels is given (Shukla et al., 2016) as:

$$I(K) = [x_k + jy_k] \qquad (32)$$

The complex coefficient of the Fourier descriptors of the boundary coordinates $a(u)$ is given as:

$$a(u) = \sum_{k=0}^{k-1} I(K) e^{-\frac{j2\pi uk}{K}} \qquad (33)$$

where K is the total number of pixels in the image, k = 0, 1, 2, ..., K-1, u = 0, 1, 2, ..., K-1 and (x, y) are the coordinates of the point.

Fourier descriptor coefficient to achieve rotation, scale and translation invariance is given as:

$$a(u) = \frac{a(u)}{a(0)}, \text{ where } u = 0, 1, \ldots\ldots K - 1 \qquad (34)$$

For shape like recognition, Fourier descriptors are particularly effective due to their invariance in terms of scale, rotation, and translation. In research introduced by Kumar and Bhatia (2014), centroid distance Fourier descriptors (CeFD) was used for feature extraction. Izzah and Suciati (2014) employed a generic Fourier descriptor (GFD) to extract feature vectors for real-time sign language. Their findings showed that GFD with the K-Nearest Neighbour (KNN) classifier performed better than the PCA result with KNN. In similar research by Kishore et al. (2015), elliptical Fourier descriptors of a closed curve were used to obtain feature vectors in the image. Simultaneously, classification was performed using an artificial neural network (ANN) algorithm. The system achieved better recognition accuracy of 95.10%. Shukla et al. (2016) employed Fourier descriptor (FD) and dynamic time warping (DTW)

to obtain features needed for gesture recognition of 26 Indian sign language (ISL) alphabets. Their system performed better than other studies (Rekha et al., 2011a, 2011b; Agrawal et al., 2012; Kishore et al., 2015). A combination of SIFT, Hu moments and FD was employed by Pan et al. (2016) to extract features from given images. The features in the image are reduced by applying PCA and LDA techniques and performed for both Chinese Sign Language (CSL) and ASL alphabets.

3.4.3. Histogram of oriented gradient

Histogram of Oriented Gradient (HOG) is a feature descriptor used to identify an object in image processing. The features obtained from HOG offer a concise and efficient image representation for image classification. It is one of the most robust feature extraction techniques used in recent times to identify shapes or structures within an image (Torrione et al., 2014). The HOG features are used to measure the degree of the input image gradient and its gradient path. The central concept behind HOG features shows that object entry and outline are considered by the circulation of edge directions (Mahmud et al., 2019). Tavari and Deorankar (2014) introduced HOG as a feature extraction technique. In their study, the features extracted are calculated by counting gradient orientation occurrences in localized portions of an image. The algorithm used for the HOG descriptor is given as follows:

Step 1: The gradient of the image i is computed by sifting it with horizontal and vertical one-dimensional distinctive derivative mask given as:

$$D_X = [-1, \ 0, \ 1], \ \text{and} \ D_Y = \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix} \tag{35}$$

where, D_x and D_y are horizontal and vertical masks respectively and obtained from X and Y derivatives using the following convolution operation:

$$I_X = I * D_X \tag{36}$$

$$I_Y = I * D_Y \tag{37}$$

The magnitude of the gradient is obtained as:

$$|G| = \sqrt{I_X{}^2 + I_Y{}^2} \tag{38}$$

The orientation of the gradient is given as:

$$\theta = \tan^{-1} \frac{I_X}{I_Y} \tag{39}$$

Step 2: Create the cell histograms. Each pixel calculates a biased vote for an orientation-based histogram channel based on the values found in the gradient computation. The cells themselves were rectangular, and the histogram channels were evenly spread over 0° to 180° or 0° to 360°, depending on whether the gradient was unsigned or signed.
Step 3: In order to change the illumination and contrast, the gradient strength is regionally normalized, which needs grouping the cells together into larger spatially connected blocks.
Step 4: The final feature vectors are obtained.

Mahmud et al. (2019) used HOG to segment the image into 64 blocks, while every block constitutes 2×2 cells. The histogram edge orientation is manipulated for each pixel in a created cell to obtained gradient direction, orientation binning and descriptor blocks which are the feature extracted and stored in the feature matrix. Their system performed with K-NN classifier better than the result obtained using Bag of features with a support vector machine in the same experiment. Raj and Jasuja (2018) used

HOG as a feature extraction technique for British sign language alphabets. The features are obtained by constituting an 8×8 pixel size window, which glides over the image. Similar research (Butt et al., 2019) used HOG with Local Binary Pattern (LBP) and statistical features to extract essential images. LBP is used to enhance the output of the extracted features. Despite the efficiency of HOG in many studies on sign language recognition, it was observed that the choice of HOG parameters affects the feature vector size—this results in computation time and accuracy issues. Joshi et al. (2020) proposed a multi-level HOG feature vector to address HOG parameters selection challenges. The study adopts a combined Taguchi and Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS) based decision-making method to determine the optimal set of multi-level HOG feature vector parameters. The proposed system demonstrates better performance than the state-of-the-art method on complex background ISL dataset.

3.4.4. Shift-invariant feature transform (SIFT)

SIFT extracts features from an image with a different variant, such as translation, scaling, and rotation. It offers a robust framework to detect unique invariant image features that robustly match different views of an image, such as translation, rotation and scaling, and some invariant changes in lighting and camera perspective. The four SIFT computational steps (Lowe, 2004) include:

Step 1: Determine approximate location and scale of salient feature points

Difference-of-Gaussian (DoG) function is used to identify potential interest points improve the computation speed, which are invariant to locations and scales (Mistry & Banerjee, 2017).

The scale-space of an image $L(x, y, \sigma)$ is computed by performing convolution between Gaussian function $G(x, y, \sigma)$ with an input image $I(x, y)$.

The scale-space function is given as:

$$L(x, y, \sigma) = G(x, y, \sigma) \bullet I(x, y) \tag{40}$$

The Gaussian function is given as:

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2} \tag{41}$$

$D(x, y, \sigma)$ can be computed from the difference of two nearby scales separated by a constant multiplicative factor k.

The difference of Gaussian equation is given as:

$$D(x, y, \sigma) = (G(x, y, k\sigma) - G(x,y,\sigma)) \bullet I(x,y) = L(x, y, k\sigma) - L(x, y, \sigma) \tag{42}$$

Step 2: Keypoint localization

The keypoint localization is determined by eliminating low contrast (edge response) and selected based on their stability measures. The key point location, $\theta(x, y)$ and scale, $m(x, y)$ is computed as given in Eqs. (43) and (44), respectively:

$$m(x,y) = \sqrt{(L(x+1, y) - L(x-1, y))^2 + (L(x, y+1) - L(x, y-1))^2} \tag{43}$$

$$\theta(x,y) = \tan^{-1}\left(\frac{(L(x, y+1) - L(x, y-1))}{(L(x+1, y) - L(x-1, y))}\right) \tag{44}$$

Step: 3 Determine orientation(s) for each keypoint.

In the third step (Orientation Computation), orientations are assigned based on the image gradient at each keypoint location

Step 4: Obtain a descriptor for each keypoint.

The key points are transformed into a representation that allows for significant levels of local shape distortion and change in illumination.

Several studies on sign recognition systems in which SIFT is employed for future extraction can be found in the literature (Agrawal et al., 2012; Tharwat et al., 2015; Yasir et al., 2016; Patil & Sinha, 2017; Shanta et al., 2018). Patil and Sinha (2017) developed a framework that focused on the time required to implement various phases of the SIFT algorithm. A system that hybridized SIFT with adaptive thresholding and gaussian blur image smoothing for feature extraction was proposed by Dudhal et al. (2019). Agrawal et al. (2012) combined shape descriptor with HOG and SIFT methods. A shape descriptor was used to analyse the overall shape of the segmented image, while the HOG descriptors were employed for invariance to illumination change, orientation and the articulated or occluded gestures associated with two-handed gestures. The key points for each sign image were determined sing SIFT tecnique and stored as a feature vector for classification. Their findings show a better performance with the combination of three techniques.

3.4.5. Speed Up Robust Feature (SURF)

Speed Up Robust Feature (SURF) is an effective technique for feature points extraction (Raj & Joseph, 2016). It is a newly-developed framework to improve the performance of an object recognition system. It is designed as an efficient alternative to SIFT, which is much faster and robust than SIFT (Sykora et al., 2014). The descriptors are derived from the pixel surrounding an interesting point. The SURF can detect objects in images taken under different extrinsic and intrinsic settings (Bhosale et al., 2014). SURF uses integral images instead of the Difference of Gaussians (DoG) employed in SIFT. Integral image is the sum of intensity value for points in the image with a location less than or equal to the original image (Cheok et al., 2019).

The integral image $I(X)$ equation presented by Mistry and Banerjee (2017) is given as:

$$I(X) = \sum_{i=0}^{i \leq x} \sum_{j=0}^{j \leq y} I(i, j) \tag{45}$$

To obtain points of interest, SURF uses a hessian blob detector. Hessian matrix determinant defines the magnitude of the response. Integral image is convoluted with box filter while box filter is the approximate filter of Gaussian filter (Mistry & Banerjee, 2017)

The Hessian matrix is given as:

$$H(X, \sigma) = \begin{bmatrix} L_{xx}(X, \sigma) & L_{xy}(X, \sigma) \\ L_{xy}(X, \sigma) & L_{yy}(X, \sigma) \end{bmatrix} \tag{46}$$

where $H(X, \sigma)$ is Hessian matrix, $X = (x, y)$ of an image I at scale σ, $L_{xx}(X, \sigma)$ (Laplacian of Gaussian) is the convolution of the Gaussian second-order derivative; $\frac{\partial^2}{\partial x^2} g(\sigma)$ with the image I in point X and similarly for $L_{xy}(X, \sigma)$ and $L_{yy}(X, \sigma)$.

SURF is a well-suited feature extraction technique for the image because of its efficient attributes including scale invariance, translation invariance, and rotation invariance. SURF has attracted a lot of research on sign language recognition systems (Rekha et al., 2011a; Yang & Peng, 2014; Fang Hu et al., 2014; Wang et al., 2015a, 2015b; Jin et al., 2016; Ravi et al., 2018; Kaluri & Reddy, 2017; Narayanan et al., 2018).

Hartanto et al. (2014) conducted a study on alphabets recognition using hand gesture images. The SURF algorithm is employed to detect and extract keypoint features. Sykora et al. (2014) compared SIFT and SURF methods of ten gestures. The findings show that the overall accuracy of SURF is higher than SIFT method. In the study of Merin Joy and Rajesh (2017), the feature detector and

descriptors were extracted from surveillance video using the SURF method. The descriptors obtained from different images for the SURF comparator were later used by finding the matching pairs for recognition. Rekha et al. (2011a, 2011b) hybridised SURF and Hu Moment Invariant methods to achieve a good recognition rate along with low time complexity. Elouariachi et al. (2021) proposed new feature extraction techniques called Quaternion Krawtchouk Moments (QKMs) for sign language recognition. The authors compared the QKMs technique with KNN classifier with various traditional feature extraction techniques (HOG, LBP, SIFT, SURF, and Gabor). Their finding shows better performance for the proposed method compared with conventional techniques.

The type of features extracted determines the performance of the recognition. Therefore, it is necessary to choose a good feature extraction technique to achieve good performance. Table 8 summarise the advantages and disadvantages of various feature extraction techniques used in vision-based sign language recognition systems.

3.5. Sign language categories

Sign languages are used to accomplish the same functions as spoken languages while simultaneously interpreting spoken words into sign language. The structure of sign language is different from spoken words and has its own phonology, syntax morphology, vocabulary, and grammar distinct from spoken languages (Agrawal et al., 2016; Sahoo et al., 2014). In achieving this, Signs are made in or near the signer's body with either one or two hands in a particular hand configuration (Schembri et al., 2013; Wadhawan & Kumar, 2021). The five basic parameters used in sign language communication include handshape, location of the hand, palm orientation, movement of the hand, and facial expression; these influence the meaning of a sign. The meaning of sign changes when either of these parameters changes (Wilcox & Occhino, 2016).

Sign languages are not universal like spoken languages, and it varies across countries due to different geographical locations, nationality, social boundaries and vocabularies (Lemaster & Monaghan, 2007). There are various sign languages all over the world; these include American Sign Language (ASL), Arabic Sign Language (ArSL), British Sign Language (BSL), Australian Sign Language (Auslan), Chinese Sign language (CSL), French Sign Language (FSL), Indian Sign Language (ISL), Danish Sign Language (DSL), German Sign Language (DGS), Taiwan Sign Language (TSL), Thai Sign Language (TLS), Finnish Sign Language (FinSL), Brazilian Sign Language (LSB), and many others that have evolved in hearing-impaired communities (Agrawal et al., 2016; Cheok et al., 2019; Kadhim & Khamees, 2020; Sahoo et al., 2014; Wadhawan & Kumar, 2021). It is important to know that most countries that share a spoken language do not share the same sign language. This happens because the deaf and hearing impaired in the two or more countries using the same spoken language were not in contact. Therefore, the sign language of these countries was developed independently. The challenges in developing a sign language recognition system include signing speed, which varies per signer. Segmentation of region of interest from entire images are also problematic due to the different environments in which the images are taken, illuminations, computational time, gesture tracking, nature of parameters used for communication such as hand shape, facial expression, orientation and so on (Agrawal et al., 2016).

3.6. Benchmark datasets

There are various publicly accessible benchmark datasets for evaluating the performance of static, isolated, and continuous sign language recognition systems. For the American sign language

Table 8
Summary of various advantages and disadvantages of feature extraction techniques.

Feature Extraction Technique	Advantages	Disadvantages
Principal Component Analysis (PCA)	Low noise sensitivity. Reduce the redundant features and large dimensionality of the data. Hence, it improves algorithm performance. It helps in overcoming data overfitting issues by decreasing the number of features.	Loss data interpretability PCA only takes into account pair-wise relationships between voxels of the brain images. It is difficult to evaluate the covariance matrix adequately. It is invariance to scaling.
Fourier Descriptor (FD)	Translation, Scale and rotation invariant. Computational complexity is more diminutive. A very robust technique	Affine transform and noise resistance is bad. Local features cannot be located because only the magnitudes of the frequencies are known instead of locations.
Scale Invariant Feature Transform (SIFT)	It is more accurate than any other descriptors. It is Rotation and scale-invariant. It is close to real-time performance.	SIFT is relatively slow compared to SURF. It is not good at illumination changes
Speed Up Robust Features (SURF)	It is more robust to scale and rotation changes. SURF is faster than SIFT in real-time applications. It has low dimensionality as compared to SIFT. It reduces the computation time.	High dimension of feature descriptor. High computational time. It is not stable to rotation, and it is not working correctly with illumination. It is not effective for low powered devices.
Histogram of Oriented Gradient (HOG)	It captures edge or gradient structure that is very characteristic of the local image. It is invariant to scaling and illumination transformations.	It is susceptible to image rotation. It is computationally expensive due to its in-depth scanning approach over the entire region of interest.

(ASL) dataset, the Purdue RVL-SLLL (Martínez et al., 2002) consists of gestures, movements, words and sentences signed by fourteen (14) signers. It comprises 2576 videos, which made up 184 videos per signer. Thirty-nine (39) of the videos are isolated motion primitives, 62 hand shapes and sentences. American Sign Language Lexicon Video Dataset (ASLLVD) (Athitsos et al., 2008) encompasses high-quality video sequences of about 3800 ASL signs corresponding to about 3000 signals signed by four native signers. RWTHBOSTON-104 (Zahedi et al., 2006) and RWTHBOSTON-400 (Dreuw et al., 2008) datasets consist of isolated and continuous ASL. RWTHBOSTON-104 dataset contains isolated sign language with a vocabulary of 104 signs and 201 sentences signed by three signers. The RWTHBOSTON-400 dataset is created to develop continuous ASL Recognition. It comprises 843 sentences with a vocabulary size of 406 words signed by four (4) signers. Also, the Massey University dataset (Barczak et al., 2011) consists of 36 classes of alphabets (A-Z) and numbers (0–9). The total numbers of images are 2160. For the Arabic Sign Language (ArSL) dataset, the Sign Language Database (Assaleh et al., 2012) contains 40 sentences, with each sentence repeated 19 times. It was acquired from eighty signers. The Signs World Atlas (Shohieb et al., 2015) has about 500 static gestures (finger spelling and hand motions) with dynamic gestures (non-manual signs) involve body language, lip reading and facial expressions. Brazilian Sign Language dataset (LIBRAS-HCRGBDS) consists of sixty-one hand configurations of the Libras (Porfirio et al., 2013). The Kinect sensor was used to collect the dataset of about 610 video clips obtained from five different signers. British sign language dataset was created by British Sign Language (BSL) Corpus (Schembri et al., 2013) and made up of videos that contained 249 people conversing in BSL with annotations of 6330 gestures from the conversation. RWTH-PHOENIX-Weather 2014 (Forster et al., 2014) and SIGNUM Database (Agris & Kraiss, 2007) were created for German sign language recognition. RWTH-

PHOENIX-Weather 2014 contains continuous sign language of 6861 sentences and 1558 vocabularies. SIGNUM Database contains a vocabulary size of 450 basic gestures and 780 sentences signed by 25 signers. Table 9 summarised benchmark datasets from different countries used by various researchers. These datasets were used as the basis for comparison or performance evaluation of the developed model.

4. Review of intelligent classification architectures Employed in sign language recognitions

After the pre-processing, segmentation, and extraction of features from the images have been completed, it is necessary to use a predictor algorithm to help gives valuable meaning to the extracted features. Just like humans learn by doing tasks repeatedly, machines are also trained to learn, and machine learning improves their performance. Machine learning is a subfield of computer science, and it is also classified as an artificial intelligence method (Voyant et al., 2017). The artificial intelligent techniques used for sign language recognition include supervised or unsupervised. Supervised machine learning took in a set of known training data and used it to infer a function from labelled training data, whereas; unsupervised machine learning is used to draw inferences from datasets with input data with no labelled response. After a comprehensive literature review, the commonly intelligent predictors utilized for recognition of sign language are k-nearest neighbour (KNN), artificial neural network (ANN), support vector machine (SVM), hidden Markov Model (HMM), Convolutional Neural Network (CNN), fuzzy logic and ensemble learning. This section briefs about machine learning techniques used for the recognition of sign language. Many papers that employed machine learning to recognise or classify sign language are reviewed and presented in the subsequent sections.

Table 9

Benchmark datasets for sign language in different countries.

Language	Dataset Name	Type of sign	Source
American Sign Language (ASL)	The Purdue RVL-SLLL Database.	Static, isolated and continuous.	(Martínez et al., 2002)
	American Sign Language Lexicon Video Dataset (ASLLVD).	Continuous	(Athitsos et al., 2008)
	RWTHBOSTON-104	Isolated and Continuous.	(Zahedi et al., 2006)
	RWTH BOSTON-400	Isolated and Continuous	(Dreuw et al., 2008)
	Massey Dataset	Static.	(Barczak et al., 2011)
Arabic Sign Language (ArSL)	Arabic Sign Language Database	Continuous	(Assaleh et al., 2012)
	Signs World Atlas, a benchmark Arabic Sign Language	Static and continuous	(Shohieb et al., 2015)
Brazilian Sign Language (Libras)	LIBRAS-HCRGBDS		(Porfirio et al., 2013)
British Sign Language (BSL)	British Sign Language Corpus	Continuous	(Schembri et al., 2013)
German Sign Language (DGS)	RWTH-PHOENIX-Weather 2014	Continuous	(Forster et al., 2014)
	The SIGNUM Database		(Agris & Kraiss, 2007)

4.1. K-Nearest Neighbour algorithm (K-NN)

K-Nearest Neighbour algorithm is also referred to as lazy learning. It is based on the principle that the instances within a dataset will generally exist in close proximity to other instances with similar properties (Selim et al., 2019). It predicts the class of a new object based on its k-nearest neighbours' classes by performing a simple majority voting to decide the class of the test instance (Alamelu et al., 2013). The 'k' parameter in K-NN refers to the number of nearest neighbours of a test data point to include in most voting processes (Kotsiantis et al., 2006). The procedure for design the K-NN classification algorithm presented in Mahmud et al. (2019) is given as follows:

Step 1: Training dataset and new input image dataset for testing were loaded.

Step 2: K value was chosen, which is the nearest data point.

Step 3: For each point in the testing data, do the following:

i Distance between each row of test data and each row of training data were computed using distance metrics. Euclidean distance equation is given as:

$$\text{Euclidean distance} = \sum_{i=1}^{n} \left(a_i^2 - b_i^2 \right) \qquad (47)$$

where a_i is the test data, b_i is the training data and k is the Numberof neighbour

i The distance value obtained were sorted in ascending order.

ii Top K rows values from the sorted array were chosen.

iii The class was assigned to test data point-based majority vote and a most frequent class of these rows.

Step 4: Repeat step 1 to step 3 for all testing images in the dataset.

K-NN is one of the machine learning algorithms that use distance measures as its core features. Some of the classification methods that are based on distance measurement used for sign language recognition includes Mahalanobis distance (Huong et al., 2016), Manhattan (YanuTara et al., 2012) and Euclidean distance (Pansare et al., 2012; Tara et al., 2012; Hartanto et al., 2014; Huong et al., 2016). Tara et al. (2012) used the Manhattan distance method to achieve a recognition accuracy of 95% with small computational latency. A similar study by Huong et al. (2016) investigated and compared Euclidean distance and Mahalanobis distance methods. The system performance revealed 90.4% and 91.5% recognition accuracy for the Euclidean distance and Mahalanobis distance, respectively.

Izzah and Suciati (2014) applied KNN to recognise 120 stored images of ASL and 120 images captured in a real-time using webcam. The system achieved an accuracy of 86% with images stored in the database and 69% recognition accuracy with the real-time captured image. Their system has a better performance compared to PCA with KNN. K-NN is used as a classifier to obtain an accuracy of the test image for static sign language recognition (Jasim & Hasanuzzaman, 2015; Tharwat et al., 2015; Gupta et al., 2016; Anand et al. (2016) developed a system for recognition of 13 alphabets of Indian sign language. The features extracted using Discrete Wavelet Transform (DWT) fed into the KNN classifier and achieved recognition of 99.23% accuracy. Also, Mahmud et al. (2019) employed K-NN for the recognition of ASL alphabets. The system used L*a*b colour space to segment the region of interest and canny edge with HOG to extract features. The system achieved recognition accuracy of 94.23%. In the same research, the system's performance was tested using a combination of a bag of features (BoF) with k-means for features extraction and SVM as classification with a recognition accuracy of 86%, which is lower than the result with K-NN. Butt et al. (2019) used K-NN for sign language recognition. They compared their model performance with the generalized linear model (GLM) and deep learning algorithms. The obtained accuracy for GLM, K-NN and deep learning are 100%, 98.03% and 99.9%, respectively. The authors reported that the performance of Naive Bayes, decision tree and other considered algorithms were much lesser than that of GLM, K-NN and deep learning for the same experiment. Table 10 summarises some of the vision-based SLR studies using various feature extraction techniques with KNN.

4.2. Artificial neural network (ANN)

Artificial Neural Network (ANN) is a computational analytical tool inspired by the biological nervous system of the brain in a bid to mimic human reasoning. It consists of highly interconnected networks that can compute input values and perform parallel computations for data processing and knowledge representation. It is a branch of artificial intelligence (AI) that helps build predictive models from large databases. Due to its robust and adaptive nature, ANN has been used to perform computations like pattern recognition, matching, classification (Jielai et al., 2015; Adegboye et al., 2020). It is generally defined by three parameters: the interconnection pattern between different layers of neurons, the weight of interconnections, and the activation function. A neuron has inputs x_1, x_2, x_3, …, x_p, each is labelled with a weight w_1, w_2, w_3, …, w_p, and measures the permeability, K is the activation function. Fig. 8 shows the structure of neuron network layers.

Output function (y) is given as:

$$y = K \bullet \sum_{j=1}^{p} x_i \cdot w_i = x_1 w_1 + x_2 w_2 + x_3 w_3 \ldots \ldots + x_n w_n \qquad (48)$$

Table 10
Summary of KNN using various feature extraction techniques.

References	Feature Extraction	Classification	Accuracy	Remark
Pansare et al. (2012)	Centroid and area of edge	Euclidean distance	90.19%.	Real-time recognition of 26 ASL alphabets.
Izzah and Suciati (2014)	Generic Fourier Descriptor (GFD) and PCA	K-NN	86 %	It achieved 69 % recognition accuracy for testing data captured in real-time.
Hartanto et al. 2014)	SURF	Euclidian distance	63.0%	Twenty-four alphabets of Indonesian sign language predicted. J and Z excluded.
Shukla et al. (2016)	Edge detection, Fourier Descriptor and Dynamic Time Warping.	K-NN	96.15%	Dynamic signs of ISL obtained from video input
Anand et al. (2016)	Dynamic Time Warping (DWT)	K-NN	99.23%	ISL gestures of 13 English alphabets.
Mahmud et al. (2019)	Histogram of Oriented Gradient (HOG)	K-NN	94.23%	26 ASL alphabets. It achieved better results compared to using Bag of Features (BoF) with k-means and SVM.
Butt et al. (2019)	HOG, Local Binary Pattern (LBP) and statistical features	KNN	98.03%	Weka-3–9-3 and Rapid Miner 9.1.0 were used to training and test data models using various classification algorithms.
Sharma et al. (2020)	ORB K-means clustering Bag of words (BoW)	K-NN	95.81% with KNN 96.96% for MLP	Static ASL gesture using Kaggle dataset. MLP perform better than other classification methods.
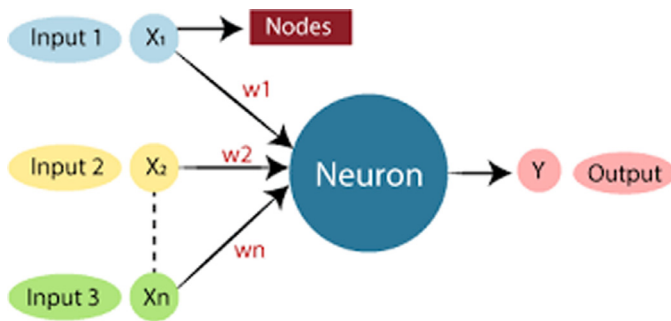
Fig. 8. Structure of a neuron and network layers (Vandeginste et al., 1998).

Neural Network algorithms used for gesture recognition includes; feed-forward neural network, backpropagation neural network algorithms and multilayer perceptron (MLP). Kishore et al. (2015) employed ANN with a backpropagation algorithm to recognise hand gestures of Indian sign language. The system obtained a feature vector using elliptical Fourier descriptors, and four cameras were used to enhance the result with a recognition accuracy of 95.10%. Prasad et al. (2016a, 2016b) employed the backpropagation NN algorithm to recognise static and dynamic gestures of ISL alphabets and numbers. The selected words are consisting 59 signs gestures with a recognition accuracy of 92.34%. The dataset used is video acquired data and extracted using a combination of PCA and Elliptical Fourier Descriptors. A similar approach using the feedforward backpropagation NN algorithm (Islam et al., 2017) was proposed for ASL alphabets and numbers and attained recognition accuracy of 94.32%. In recognition of the Bangla sign language of 20 Bangla alphabets, the feedforward backpropagation algorithm NN was employed by Hasan et al. (2017). The extracted features from a combination of canny edge and FCC are fed into the neural network. The system achieved recognition accuracy of 96.5%. Also, Raj and Jasuja (2018) proposed an ANN-based system to recognise British sign language alphabets. The testing dataset contained 780 sign images, and the system performance attained 99.01% accuracy, which is better than similar research (Liwicki & Everingham, 2009) for British sign language alphabets with a recognition accuracy of 98.9%. Shaik et al. (2015) hybridised backpropagation neural network with a genetic algorithm model. They reported that their system performed better than existing backpropagation-based hand gesture recognition. The extensive review we performed revealed several researchers (Tariq et al., 2012; Adithya et al., 2013; Yasir & Khan, 2014) had used ANN for sign language recognition. The proposed systems have demonstrated good performance. Recent research by Sharma et al. (2020a, 2020b) used several machine learning algorithms to recognise features extracted from the combination of ORB with K-means and Bag of words (BoW). Their finding shows better recognition accuracy of 96.96% using the multilayer perceptron model. The summary of SLR studies using various feature extraction techniques with ANN is presented in Table 11.

4.3. Support vector machine (SVM)

Support Vector Machine (SVM) is a supervised learning model with associated learning algorithms that are non-probabilistic. It is a popular pattern recognition learning technique for classification regression analysis (Xanthopoulos & Razzaghi, 2014). SVM can be used to solve both pattern-classification and nonlinear-regression problems (Adeyanju et al., 2015), but it is most useful in solving difficult pattern-classification problems (Martiskainen et al., 2009). Classification is performed in SVM by differentiating between two or more data classes. This is done by defining an optimal hyperplane that splits all categories, as shown in Fig. 9(a). However, suppose it is impossible to find a single line to separate the two

Table 11
Summary of ANN method using various feature extraction techniques.

References	Feature Extraction	Classification	Accuracy	Remark
Tariq et al. (2012)	Convexity defects	ANN (Feedforward Neural Network with Backpropagation algorithm)	62% on the testing dataset	Low recognition rate of "M" and "N" characters due to similarities in shape.
Kishore et al. (2015)	Elliptical Fourier Descriptor (EFD)	ANN with backpropagation algorithm	95.10%	The images were acquired using four cameras
Prasad et al. (2016)	Elliptical Fourier Descriptor (EFD) and PCA	ANN	92.34%	It achieved better results compared with a combination of Morphological operation, Sobel, DCT and ANN.
Islam et al. (2017)	K-curvature and convex hull algorithm	Artificial Neural Network (ANN)-Feedforward Backpropagation Neural Network	94.32%	The system achieved better recognition with a combination of all the features extracted.
Hasan et al. (2017)	Canny edge detector. Freeman Chain Code (FCC)	Artificial Neural Network (ANN)	96.50%.	Recognition accuracy is low under a dark environment and poor illumination.
Raj and Jasuja (2018)	Histogram of Oriented Gradient (HOG)	ANN (Feedforward backpropagation Algorithm)	99.0%	26 British sign language (BSL) alphabets signed with two hands
Sharma et al. (2020)	ORB K-means clustering Bag of words (BoW)	Multilayer Perceptron (MLP) Random Forests, SVM, Naïve Bayes, Logistic Regression, K-NN	96.96% for MLP	Static ASL gesture using Kaggle dataset. MLP perform better than other classification methods.
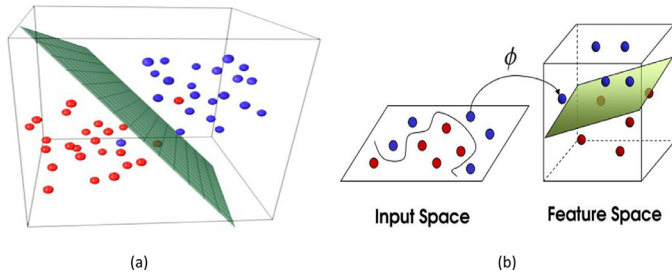
Fig. 9. (a) Hyperplane that separate two classes of data, (b) kernel trick (Merenda et al., 2020).

classes in the input space. In that case, a classification can be performed using the kernel trick, as shown in Fig. 9(b).

SVM operates according to the concept of margin calculation, and boundaries are drawn between classes. The margins are drawn so that the distance between the margin and the classes is maximum. Hence the classification error is reduced. Optimization techniques are employed in finding the optimal hyperplane (Cheok et al., 2019). The hyperplane equation that does the separation is given as:

$$\vec{w} \bullet \vec{x} + b = 0 \tag{49}$$

where w is the weight vector for $\vec{w}$, for training data $(\vec{x}_1, \vec{y}_1), \ldots, (\vec{x}_n, \vec{y}_n)$, y_i are either 1 or -1, indicating to which class the data $\vec{x}_l$ belong (target output). The weight vector decides the orientation of the decision boundary, whereas bias point b decides its location.

Distance, D_1 between the support vector 1 and plane is computed according to:

$$D1 = \frac{1}{\|\vec{w}\|} \tag{50}$$

Distance, D_2 between the support vector 2 and plane is computed according to:

$$D2 = \frac{1}{\|\vec{w}\|} \tag{51}$$

The Margin, M is the addition of distance between support vectors D_1 and D_2 is given by:

$$M = \frac{2}{\|\vec{w}\|} \tag{52}$$

SVM has been proposed for the recognition of sign language by many researchers (Rashid et al., 2009; Rekha et al., 2011b; Mohandes, 2013; Moreira Almeida et al., 2014; Kong & Ranganath, 2014; Dahmani & Larabi, 2014; Sun et al., 2015; (Jayshree R. Pansare & Ingle, 2016); Raheja et al., 2016; Lee et al., 2016; Chong & Lee, 2018; Rahim et al., 2019; Abiyev et al., 2020). Agrawal et al. (2012) developed a model for recognition of Indian sign language in real-time using SVM. The system combined shape descriptors with HOG and SIFT to extract features. The model achieved a recognition accuracy of 93%. The performance of SVM has been compared with K-NN in (Tharwat et al., 2015; Yasir et al., 2016). It reported that SVM has better performance compare to the KNN classifier. Also, Tharwat et al. (2015) used SMV to recognition Arabic sign language (ArSL). The model achieved a recognition accuracy of 99%. A similar research also used Kinect with the SVM classifier to develop Indian Sign Language Recognition (SLR) system.

Table 12
Summary of sign language recognition based on SVM.

References	Feature Extraction	Classification	Accuracy	Remark
(Agrawal et al., 2012)	Shape Descriptors, HOG and SIFT	SVM	93%	Low recognition rate with similar signs such as "M" and "N"
(Tharwat et al., 2015)	SIFT LDA	SVM K-NN (with k=1 and k=5)	99%	SVM achieved better results compared with K-NN
(Yasir et al., 2016)	SIFT, K-means clustering and Bag of words (BoW)	SVM K-NN	-	SVM achieved a better result with a large dataset than K-NN.
(Pan et al., 2016)	SIFT, Hu-moments, Fourier Descriptor (FD), PCA and LDA	SVM	99.8% (CSL) 94% (ASL)	It achieved the recognition accuracy under none complex background
(Raheja et al., 2016)	Hu Moments and motion rajectory	SVM	97.5%	Four signs gestures ("A", "B", "C" and "Hello") of Indian sign language.
(Rahim et al., 2019)	Feature fusion of CNN	SVM	97.28%	Low recognition under complex background, poor illumination and partial occlusion.
(Athira et al., 2019)	Zernike moments	SVM	Accuracy for the static and dynamic sign is 91% and 89%, respectively.	Low recognition with dataset captured under cluttered background and different illumination.
(Joshi et al., 2020)	Multi-level HOG using Taguchi and Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS).	SVM	92% for ISL and 99.2% for ASL	Sign images obtained from Kaggle datasets. Good performance under complex background.
Barbhuiya et al. (2021)	Convolution Neural Network (CNN) based on AlexNet and VGG16 models	SVM	99.82%	It requires high computational complexity when dealing with a large dataset.

The system demonstrated a recognition accuracy of 97.5% (Raheja et al., 2016). In Yasir et al. (2016), SVM was employed to predict Bangal sign language. A bag of words with k-means clustering was used to reduce feature vectors obtained from the video sequence of signs. SVM was used as a classifier, and Zernike's moments were employed to find the keyframe in a video sequence in Athira et al. (2019). The system attained an accuracy of 91% and 89% for static and dynamic, respectively. However, the performances of both systems are low under complex backgrounds and poor lighting. SVM was implemented on a mobile application to recognise 16 ASL alphabets and achieved a recognition accuracy of 97.13% in Jin et al. (2016). Other studies that employed SVM as a classifier with better performance include (Joshi et al., 2020; Barbhuiya et al., 2021; Sharma et al., 2020a, 2020b). The summary of SLR using various feature extraction techniques coupled with SVM is presented in Table 12.

4.4. Hidden Markov model (HMM)

One of the most effective sequence detection and recognition techniques is the Hidden Markov Model (HMM). It is a statistical model assumed to be a Markov process with an unknown set of hidden parameters. The hidden parameters can be obtained from related observation parameters (Lan et al., 2017). Several re-searchers have used this technique in sign language recognition, where the image to be processing might be in a video sequence and has achieved better recognition accuracy. In HMM, a new state is generated when input apply. The "hidden" in the Hidden Markov model implies that shifts from the old state to the new state are not clearly measurable, and the probability of transition depends on how the model train with the training sets. The technique works by train the model using training sets. The training has to be valid and get all classes classified because the model will only learn from what is trained. Some of the HMM used as a classifier for sign language recognition includes the continuous Hidden Markov Model (using Gaussian) and discrete Hidden Markov Model (using Multinomial) (Suharjito et al., 2019).

Aloysius and Geetha (2020a, 2020b) gave procedural steps for continuous sign language recognition (CSLR) using HMM. The HMM classifier is structured to be modelled per each sign label, with a predefined number of hidden states s as shown in Fig. 10. Given a video clip as a sequence of images $X_1^N = X_1, X_2 \ldots \ldots \ldots X_n$, a CSLR system finds a sequence of sign words $Gloss_1^m$, for which X_1^N best fit the learned models. For decoding purposes, a sliding window approach was considered for each video segment from the continuous utterance. At the same time, the Viterbi algorithm (Forney, 1973) was carried out for each HMM, to find the most likely gesture based on their scores. The
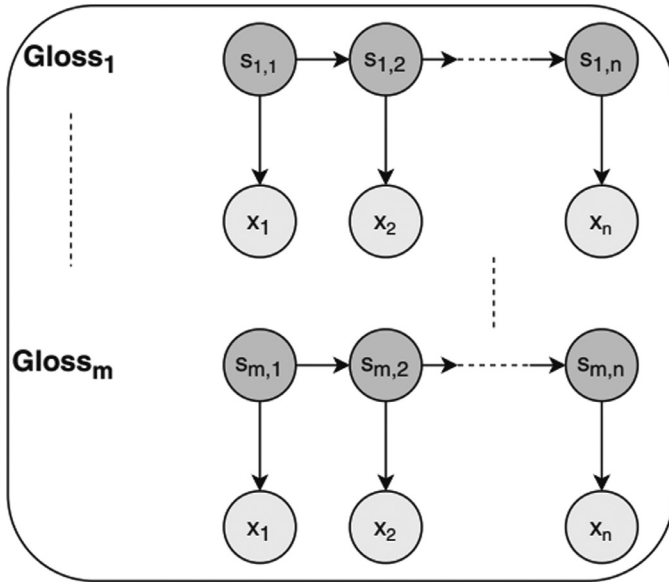
Fig. 10. HMM model for gesture recognition (Aloysius and Geetha, 2020b).

model with the best scores was considered as the sign (gloss or label).

HMM equation was employed to maximize optimal sign and is given as:

$$[Gloss_1^m]_{optimal} = \arg\max\left\{\max_{s_i^n}\left\{\prod_n p(X_n/s_n, \; Gloss_1^m). \; p\left(\frac{s_n}{s_{n-1}}, \; Gloss_1^m\right)\right\}\right\} \tag{53}$$

For m labels and observation x of length n, x_j is an observation of input sequence x at position j and $s_{i,j}$ represents the state of HMM for sign label i ($Gloss_i$) and x_j.

Zaki and Shaheen (2011) have proposed the HMM-based sign language model using an appearance-based feature extraction technique. Hybridization of three feature extraction techniques was used to obtain features that define the point of articulation arrangement, hand orientation and movement. The system used skin colour thresholding and connected component identification to extract only the dominant hand, face and track it. Their system achieved an overall error rate of 10.9% on the RWTH-BOSTON-50 database. Also, Li et al. (2016) used HMM as a classifier to recognise 11 home-service-related Taiwan sign language words. The system combined PCA with entropy-based K-means algorithm for data transformation, eliminating redundant dimensions and evaluating the number of clusters in the dataset without losing the main information. The ABC algorithm and the Baum–Welch algorithm are integrated to resolve optimization problems and achieved recognition accuracy of 91.30%. Some researchers hybridize HMM with other techniques to improve SLR accuracy, such as CNN-HMM (Koller et al., 2018), HMM-SVM (Lee et al., 2016) with Kinect to create a 3D model of the acquired image and achieved recognition accuracy of 85.14% and Coupled-HMM (Kumar et al., 2017) used Kinect and Leap to acquire data with a recognition accuracy of 90.80%.

Kaluri and Reddy (2017) introduced a genetic algorithm to improve HMM for gesture recognition and compared the performance of the proposed model against SVM and Neural Network. The recognition accuracy for the proposed method, neural network and SVM are 83%, 72% and 79%, respectively. Suharjito et al. (2019) proposed models for predicting ten signs of Argentine sign language using sign Gaussian HMM and Multinomial HMM. The models are

compared with edge detection and skin detection techniques. Their result showed that Gaussian HMM performed better when using edge detection with a recognition accuracy of 83%. HMM-based algorithms for sign language recognition have been developed by other researchers, including Coupled HMM (CHMM) (Brand et al., 1997; Kumar et al., 2017), Tied-Mixture Density HMM (Zhang et al., 2004), Parallel HMM (PaHMM) (Vogler & Metaxas, 1999) and Parametric HMM (PHMM) (Wilson & Bobick, 1999).

4.5. Convolutional neural network (CNN)

Convolution neural network offers a wide range of applications, including face recognition, scene labelling, image classification, voice recognition and natural language processing (Gopika et al., 2020). It is a type of deep learning algorithm that takes in an input image, assigns values to various features in the image, and uses the value to differentiate various features (Saha, 2018). It normally has input, convolution, pooling, and fully connected layers with output (Nisha & Meeral, 2021). Fig. 11 shows the operation of a convolution neural network. The convolution layer extracts the input data by a convolution operation. The pooling layer realizes features dimensionality reduction and controls the calculation burden by selecting the most significant features to prevent overfitting. The extracted features are fed to the fully connected layer, consisting of an activation function (Jiang et al., 2020). Instead of building complex handmade features, CNNs can automate the process of feature extraction and perform better compared to other traditional image processing techniques (Ameen & Vadera, 2017).

A real-time ASL recognition system using CNN was proposed by Taskiran et al. (2018). The input image is segmented using a convex hull algorithm to obtain the hand region using YCbCr skin colour segmentation. The proposed CNN model consists of the input layer, two 2D convolution layers, pooling, flattening and two dense layers. An accuracy of 98.05% was attained when tested with the real-time data. Tolentino et al. (2019) developed a real-time system to learn sign language for beginners requiring hand recognition. This system is based on a skin-colour modelling technique to extract the hand region from the background and integrate it into Convolutional Neural Network (CNN) to classify images. The system achieved an accuracy of 90.04% for ASL alphabets, 93.44% for number recognition and 97.52% for static word recognition.

Dudhal et al. (2019) proposed CNN based approach for ISL recognition using SIFT to extract the features from an image with a different variant. The output of SIFT algorithm was fed into Convolution Neural Network (CNN) for classification with a recognition accuracy of 92.78%. The research also used adaptive thresholding on the sign image with CNN and achieved an accuracy of 91.84%. The study shows that better performance is achieved with the hybridization of SIFT, while hybridization of adaptive thresholding and gaussian blur is better than adaptive thresholding only. Similar research (Barbhuiya et al., 2021) proposed a similar approach for robust modelling of static signs language recognition using deep learning-based convolutional neural networks (CNN). The proposed sign language recognition system includes four major phases: image acquisition, image preprocessing, training and testing of the CNN classifier. The developed system achieved the highest training accuracy of 99.72% and 99.90% on coloured and grayscale images. More recently, Barbhuiya et al. (2021) designed convolutional neural networks (CNN) framework based on a modified pre-trained AlexNet and modified pre-trained VGG16 model for feature extraction with multiclass support vector machine (SVM) for the classifier. The system achieved recognition accuracy of 99.82%.

Another deep learning approach using a fine-tuned VGG16 based on CNN model Nelson et al. (2019) achieved recognition accuracy of 97% compared to a fine-tuned VGG19 (Khari et al., 2019) with a recognition accuracy of 94.8%. Adithya and Rajesh (2020) in-
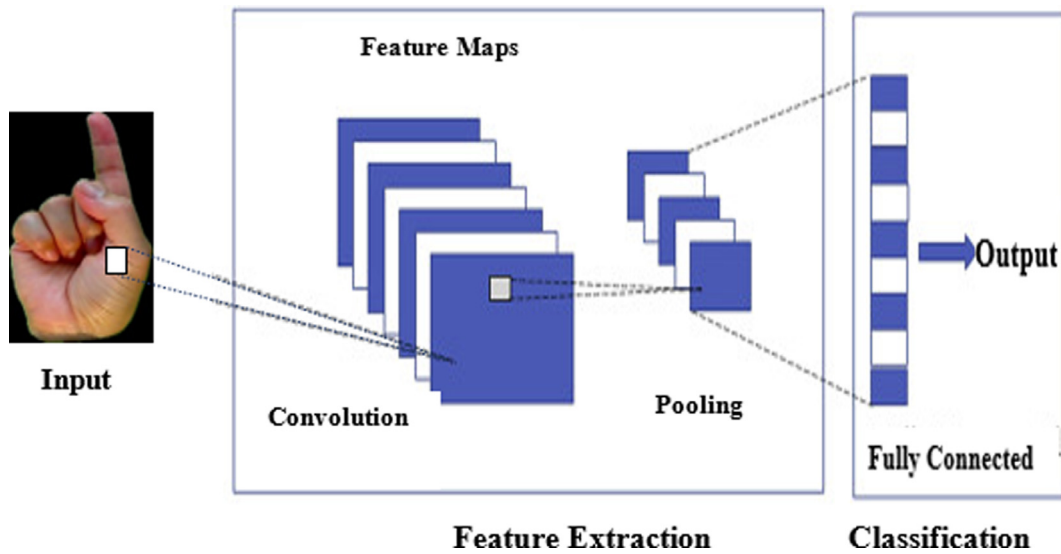
Fig. 11. Architecture of Convolutional neural network.

troduced a deep learning model based on CNN to recognise static ASL alphabets. The model was tested on the National University of Singapore (NUS) hand posture dataset and American fingerspelling A dataset which contains 24 letters of the ASL alphabet. The developed model performed well on the two datasets with 94.7% and 99.96% recognition accuracy on the NUS and American fingerspelling A datasets.

4.6. Fuzzy logic systems

Fuzzy logic is a form of multi-valued logic that uses the mathematical theory of fuzzy sets for reasoning rather than accuracy (Ngai et al., 2014). Zadeh presented a fuzzy logic algorithm in 1965 to solve the models that deal with logical reasoning, such as imprecise and vague (Song & Chissom, 1993). The approach mimics how humans make decisions involving all intermediate possibilities between digital values 0 and 1 (Scott et al., 2003; Nuhu et al., 2021). This classification technique is based on fuzzy set theory and is used in a different area to solve problems. It operates based on three stages which are Fuzzification, Inference and Defuzzification. In the fuzzification stage, the input feature vectors are converted into fuzzy values with the aid of a fuzzy membership function, which correlates with the score of each fuzzy value. The second stage is the fuzzy inference engine, where the mapping between input and output membership function is done based on the fuzzy rules. At the same time, defuzzification is the final phase in fuzzy logic which combines the output into a single numerical value for prediction (Kaluri and Reddy, 2016a, 2016b).

The fuzzy logic system includes Fuzzy Inference System (FIS), and Adaptive Neuro-Fuzzy Inference System (ANFIS) (Nedeljkovic et al., 2004) have been applied in sign language recognition system.

4.6.1. Fuzzy inference system (FIS)

Fuzzy inference uses fuzzy logic to create a mapping from a given input to an output. It entails assigning each rule a weight between 0 and 1, then multiplying the membership value assigned to the outcome of the rules (Lenard et al., 1999). The process of fuzzy inference involves membership functions, fuzzy logic operators and if-then rules. Fuzzy Inference System (FIS) implementation can be further classified into Mamdani and Sugeno (Wang, 2001).

Mamdani fuzzy inference system: Mamdani fuzzy inference system was first introduced by Mamdani and Assilian (1975). The Mamdani technique is one of the most extensively utilized fuzzy inference systems. It is employed because the design combines expert knowledge with more intuition in the form of IF-THEN rules expressed in natural language (Izquierdo & Izquierdo, 2018).

In Mamdani, the rules of fuzzy systems are given as:

$$Q_i : IF\ x_i\ is\ A_1^j\ and\ x_n\ is\ A_n^j\ THEN\ y\ is\ B_j \tag{54}$$

where Q_i denotes the ith rule, $i = 1, \ldots\ldots\ldots, N_r$, N_r is the number of rules, x_n is n-th input to the fuzzy system, A_n^j and B_n^j are fuzzy sets described by $MF\ \mu_{A_{i,j}}(x_i) :\rightarrow [0, 1]$ and $\mu_{B_{i,j}}(y) :\rightarrow [0, 1]$.

The propositions in the IF part of the rule are combined by applying minimum operators. Sometimes the product is calculated, but it mostly depends on the situation. The number of prepositions in the consequence part of the rule depends on the number of outputs of the fuzzy system (Garcia-Diaz et al., 2013).

4.7. Sugeno fuzzy inference system

Sugeno technique of fuzzy inference is similar to the Mamdani technique. The few steps of the fuzzy inference process, fuzzifying the inputs and applying the fuzzy operator, are exactly the same. In comparison, Sugeno fuzzy inference employs singleton output membership functions, which are either constant or linear functions of the input values. When compared to a Mamdani system, the Sugeno defuzzification procedure is more computationally efficient.

The rules in the functional Sugeno fuzzy inference system is given by:

$$Q_i : IF\ x_i\ is\ A_1^j\ and\ x_n\ is\ A_n^j\ THEN\ y\ is\ f_j(x) \tag{55}$$

where $f_j(x)$ is a crisp function of the input variables, rather than a fuzzy proposition.

The major difference between Mamdani-type FIS and Sugeno-type FIS is that Mamdani-type FIS uses the defuzzification technique of a fuzzy output. In contrast, Sugeno-type FIS uses a weighted average to compute the crisp output hereby bypassed the defuzzification stage. Mamdani FIS is well suited to human input with a better interpretable rule base (Kaur & Kaur, 2012; Shleeg & Ellabib, 2013).

4.7.1. Adaptive neuro-fuzzy inference system (ANFIS)

Adaptive Neuro-Fuzzy Inference System (ANFIS) proposed by (Jang, 1993) is a neural network-based integration system that optimizes the fuzzy inference system. It creates a set of fuzzy if–then
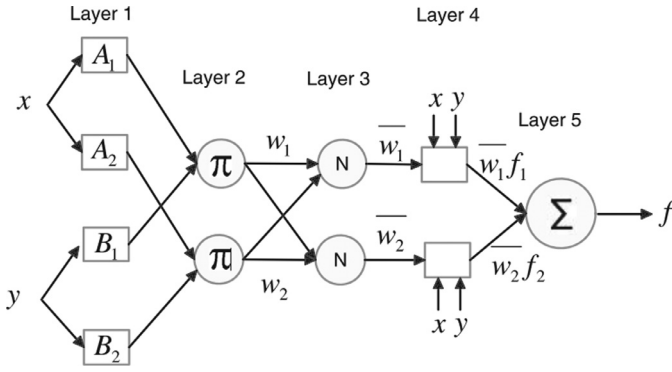
Fig. 12. ANFIS architecture (Al-Hmouz et al., 2012).

rules with appropriate membership functions. Human expertise regarding the outputs to be modelled is used to set the initial fuzzy rules and membership functions. ANFIS can modify these fuzzy if–then rules and membership functions based on an interconnected neural network to map the numerical inputs into a reduced output error (Ahmed & Shah, 2017; Al-Hmouz et al., 2012) as shown in ANFIS architecture of two inputs in Fig. 12. This technique refines fuzzy IF-THEN rules to describe the behavior of a complex system and enables a fast and accurate learning mechanism. It has successfully combined the advantages of fuzzy logic and neural network techniques into a single approach (Jang, 1993; Kassem et al., 2017).

Given Sugeno if-then rules as:

Rule 1: *If x is A*1 *and y is B*1, *then f*1 $= p1\,x + q1\,y + r1$

Rule 2: *If x is A*2 *and y is B*2, *then f*2 $= p2\,x + q2\,y + r2$

The overall output of the model is computed by:

$$\sum_i \overline{w_i} f_i = \frac{\sum_i w_i f_i}{\sum_i w_i} \tag{56}$$

where $\bar{w}_i f_i$ is the output of node i in Layer 4 as illustrated in Fig. 12, x and y are the inputs, Ai and Bi are the fuzzy sets, fi are the outputs within the fuzzy region specified by the fuzzy rule, and pi, qi and ri are the design parameters that are determined during the training process.

Al-Jarrah and Halawani (2001) developed an adaptive neuro-fuzzy inference system to recognise Arabic sign language. The system was applied on sign images with a bare hand and achieved a recognition accuracy of 93.55%. Al-Jarrah and Al-Omari (2007) also presented a similar model using an improved algorithm. The system achieved recognition accuracy of 97.5%, 100% for the 10 and 19 rules, respectively. Kausar et al. (2008) used colour marked gloves to extract fingertip and finger-joint features from the Pakistani sign language alphabets. The angle between fingertip and joint was distinguished using the Fuzzy Inference System (FIS). Lech and Kostek (2012) presented a fuzzy rule-based inference system for dynamic gesture recognition. Their system has shown better performance than the fixed speed thresholds method using Sugeno fuzzy inference as a classifier for the video dataset.

In the study of Kishore et al. (2016), the Fuzzy Inference Engine was proposed for Indian sign language recognition. Continuous gestures of 50 words were segmented and extracted using Horn Schunck Optical Flow (HSOF) algorithm and active contours, respectively. The system combined the most dominant features of data to form the features used for recognition. Mufarroha and Utaminingrum (2017) used an adaptive network-based fuzzy inference system to group features and a KNN classifier to speed up the recognition process. The system achieved recognition accuracy of 80.77% with the ten epochs. Elatawy et al. (2020) proposed a predictive model to recognize Arabic sign language alphabets. The

dataset used were first converted into the neutrosophic domain to add more information. The system used the fuzzy c-means technique and attained a recognition accuracy of 91%. Table 13 summarises the reviewed papers on the fuzzy logic systems used for sign language recognition.

4.8. Ensemble learning

Ensemble learning (EL) is a hybrid learning model that combines multiple classifiers algorithms to improve the model's prediction performance (Simske, 2019). It can make a strong classifier with low training errors from the combination of multiple weak classifiers. An ensemble learning method is a meta-algorithm that combines several machine learning techniques to solve many real-life problems and create a predictive model with improved accuracy (Polikar, 2012; Huang & Chen, 2020; Zhao et al., 2021). The algorithm has been widely used in various machine learning applications, including face and object recognition, error correction, object tracking, and feature selection. Examples of ensemble learning techniques are Adaptive Boosting (AdaBoost) and XGBoost (Extreme Gradient Boosting).

The commonly adopted ensemble learning approaches are bagging and boosting. Bagging combines the predicted classifications (prediction) from multiple models or the same type of model for different learning data. Bagging also addresses the inherent instability of results when applying complex models to relatively small data sets. In Boosting, different models' decisions, like bagging, by amalgamating the various outputs into a single prediction but derives the individual models in different ways. In the bagging approach, however, the models receive equal weight, whereas weighting is used to influence the more successful in boosting.

Previous research extensively discussed ensembled learning techniques (Dong et al., 2008; Polikar, 2012; Sagi & Rokach, 2018; Talia et al., 2016; Zhou, 2009).

Aloysius and Geetha (2020a, 2020b) presented a weighted average ensemble of CNNs, which includes a low-resolution Network (LRN), an intermediate resolution network (IRN) and a high-resolution Network (HRN). Their system demonstrated improved performance with high accuracy of 91.76% for static hand gestures than the CNN trained model. Zhang et al. (2005a) proposed an ensemble classifier based on continuous HMM (CHMM) and AdaBoost to boost the performance of Chinese sign language (CSL) subwords. Zhang et al. (2005b) developed hierarchical voting classification (HVC) based on a combination of continuous hidden Markov models (CHMM) with an accuracy of 95.4%. Simon et al. (2007) proposed an ensemble system to recognize sign language and human behavior recognition. Also, Huang et al. (2012) developed an ensemble system based on the hybridization of HMM and DWT for hand gesture recognition. Their method outperforms state of the art approach using HMM or DWT. In research (Savur & Sahin, 2017; Mustafa & Lahsasna, 2016), the American sign language recognition system was developed using ensemble learning and achieve good recognition accuracy. Yang et al. (2016) presented a system based on a combination of dynamic time warping based Level Building (LB-DTW) and Fast HMM to improve the recognition accuracy of the continuous sign. Fast HMM was used to handle computation complexity. The system achieved recognition accuracy of 87.80% on 100 Chinese sign language (CSL) sentences with only 21 signs vocabularies. Ji et al. (2017) proposed sign language interactive robot commandos using a hybrid classifier of CNN-SVM. Their system attained recognition accuracy of 97.72 % with NUS hand posture dataset II. Kim et al. (2018) presented an ensemble artificial neural network. The system used an electromyography armband sensor with 8-channel and eight ANN classifiers. The optimum recognition accuracy of 97.40% was achieved with their proposed method. Nguen et al. (2019) ensemble a system using ResNet-based CNN

Table 13

Summary of sign language recognition based on fuzzy logic systems.

Author	Extraction Techniques	Classification	Accuracy	Remarks
Al-Jarrah and Halawani (2001)	–	Adaptive Neuro-Fuzzy Inference System (ANFIS).	93.55%	Recognition of Arabic signs images with bare hand.
Al-Jarrah and Al-Omari (2007)	Boundary and region features.	Adaptive Neuro-Fuzzy Inference System.	100% with 19 rules and 97.5% with ten rules	The system performed better than (Al-Jarrah and Al-Omari, 2007) with gestures of similar boundaries.
Prasad et al. (2016)	Active contour	Sugeno fuzzy inference system	92.5%	The video dataset of Indian signs contains 80 words and sentence testing.
Kishore et al. (2016)	Active contours	Fuzzy Inference Engine	96%	The system achieved better results compared to other models in the same categories.
Lech and Kostek (2012)		fuzzy rule-based inference system with Kalman filters		Little effect defect was observed using Kalman filters.It was implemented on few simple gestures.
Elatawy et al. (2020)	Gray level co-occurrence matrix (GLCM)	fuzzy c means	91%	It recognized 28 Arabic sign language sign alphabets.

Table 14

Summary of reviewed ensemble learning methods in sign language recognition.

Author	Classification	Accuracy	Remarks
Aloysius and Geetha (2020a, 2020b)	weighted average ensemble of CNNs (LRN, IRN and HRN)	91.76%	The ensemble model performed better than various individual models selected for comparisons.
Zhang et al. (2005)	Continuous HMM (CHMM) and AdaBoost	92.70%	It achieved improved recognition accuracy of 3% than using only the CHMM classifier.
Zhang and Gao (2005)	Combination of CHMM	95.4%	It is effective and improves performance with a limited training dataset.
Huang et al. (2012)	HMM and DWT	-	The model demonstrated better than DWT or HMM alone with a larger dataset but with high computation cost due to hybridization techniques.
Savur and Sahin (2016)	SVM, Ensemble learning (Bagged Tree) classifier	60.85% 80%	The ensemble learning (Bagged Tree) classifier outperformed the SVM classifier for the single and multi-subject datasets.
Yang et al. (2016)	LB-DTW and Fast HMM	87.8%	The proposed system solves the problem of sequence segmentation and recognition. Also, grammar and sign length constraints were used to improve the recognition accuracy.
Ji et al. (2017)	CNN-SVM	97.72 %	An egocentric vision-based hand posture recognition is suitable for the control of reconnaissance robots on the battlefield for commandos.
Gupta Jha (2020)	Ensemble learning using Multiple SVM	98.7%	It performed well in real-time and better than a single classifier and other multiple classifiers.
Nguen et al. (2019)	ResNet-based CNN and ResNet quaternion CNN	94.1%	It achieved a better result using the ResNet quaternion convolutional neural network than the result obtained using ResNet CNN.
Sharma et al. (2020)	RF, TrAdaboost, TrBagg and TrResampling	97.04%	Trbaggboost has improved performance when the labelled dataset is given compared to RF, TrAdaboost, TrBagg and TrResampling.
Kim et al. (2018)	E-ANN	97.4%	Had a higher accuracy and more stable performance than a general ANN by comparing the average accuracies of the classifiers and their standard deviations
Yuan et al. 2020)	Random Forest (RF) ANN SVM	95.48%	RF achieved better recognition accuracy compared to ANN and SVM

and ResNet quaternion CNN to recognise Japanese sign language. Gupta and Jha (2020) proposed a real-time recognition system for continuously signed sentences of Indian sign language. The proposed system used an ensemble technique based on multiple SVM classifiers on features extracted from fixed windows durations and achieved an accuracy of 98.7% for 11 sentences. Raghuveera et al. (2020) used an ensemble of three features (histogram of oriented gradient (HOG), speeded up robust features (SURF) and local binary pattern (LBP)) to be trained SVM. Their system achieved a recognition accuracy of 71.85% with a response time of 35 s. Sharma et al. (2020a, 2020b) introduced a novel ensemble–based transfer learning called the Trbaggboost algorithm. The performance of the proposed ensemble method is compared with transfer learning algorithms which include TrAdaboost (Yao & Doretto, 2010), TrResampling (Liu et al., 2017), TrBagg (Kamishima et al., 2009), and Random forest (Yuan et al., 2020). An improved Trbaggboost algorithm outperforms the state-of-the-art Trbaggboost with a recognition accuracy of 97.04 %. Yuan et al. (2020) proposed an ensemble technique based on random forest (RF) for offline Chinese sign language alphabets recognition. Four features were extracted from the subjects' forearm and achieved recognition accuracy of 95.48%, which is better than SVM and ANN. The summary of the reviewed ensemble learning methods in sign language recognition is presented in Table 14.

Table 15
Advantages and disadvantages of classifiers.

Sign language classifier methods	Advantages	Disadvantages
K-Nearest Neighbors	It is effortless to implement. It is a simple algorithm to interpret.	Very sensitive to irrelevant features. It does not work well with a large dataset. It does not work well with high dimensions.
Artificial Neural Networks (ANN)	It is useful where the fast evaluation of the learned target function is required. It is quite robust to noise in the training dataset. It has fault tolerance.	It is computationally expensive and has difficulties in finding a proper network structure.
Support Vector Machine (SVM)	It performs better when dealing with multidimensions and continuous features. It is applicable in numerous domains. Tolerance to irrelevant attributes.	It requires a large sample of the dataset to achieve its maximum prediction accuracy. Hyperparameters are often challenging while interpreting their impact.
Hidden Markov Model (HMM)	It performs relatively well in recognition. It is easier to implement and analyse. It eliminates the label bias problem.	HMMs often have a large number of unstructured parameters. It requires a huge amount of training to obtain better results. It requires a large dataset for training.
Convolution Neural Network (CNN)	It automatically detects important features without any human supervision. It handling image classification successfully with high accuracy.	High computational cost. It requires a lot of training data to achieve good accuracy. Lack of ability to be spatially invariant to the input data. It does not encode the position and orientation of the object.
Fuzzy logic	It is a robust system where no precise inputs are required. It is flexible and can also allow modifications. It is an expert base technique that provides solutions to complex solutions. It deals with complicated problems in a simple way.	It is completely dependent on human intelligence and expertise. It has low accuracy, and its predictors are not always correct.
Ensemble Learning	It improves the average prediction performance. It provides high accuracy and a more stable model. It reduces the variance of predictive errors.	It can be more difficult to interpret. Sometimes the model can be overfitted or underfit using the ensemble learning method.

Table 16
Summary of reviewed less popular machine learning algorithms employed in sign language recognition.

Author	Extraction Techniques	Classification	Accuracy	Remarks
Wong and Cipolla (2005)	Motion Gradient Orientation (MGO)	Sparse Bayesian Classifier RVM		The proposed system works consistently in real-time and gives a probabilistic output useful in complex motion analysis.
Karami et al. (2011)	DWT	MLP-NN	94.06%.	The model was tested with a data set of 640 Persian sign images samples containing 20 images for each sign.
Verma and Dev (2009)	Sobel edge detector, harris corner and fuzzy c means	FSM	-	The proposed technique was successfully applied on gestures such as waving left hand, waving right hand, signalling to stop, forward and rewind.
Maraqa and Abu-Zaiter (2008)	fuzzy clustering means (FCM)	RNN	95.11%	The system was tested with 900 images of the 30 gestures each, signed by two people. RNN outperform Feedforward, Elman's and Jordan's neural networks.
Bhat et al. (2013)	Canny edge	SOM	92%	Real-time recognition of 18 gestures of American Sign Language.
Lim et al. (2016)	-	Serial particle filter with feature covariance matrix	87.33%	It used a serial particle filter to accurately track the hand's position and work well under the uncluttered background.
Camgoz et al. (2017)	-	CNN and BLSTM with CTC		It outperforms the existing model selected for comparison.
Vincent et al. (2019)	-	DeepConvLSTM with data augmentation	91.1%	Hybridization of the three models (DeepConvLSTM) proved to be better than other models compared with ConvNet, LSTM, RF, SVM, and KNN.
Meng and Li (2021)		Graph Convolutional Network (GCN)	98.08%	The system achieved better performance and reduce motion blurring, sign variation and finger occlusion.

Various sign language classifiers have been reviewed in this paper. The techniques have merits and demerits over others. Table 15 shows various advantages and disadvantages of the classifiers reviewed in this study.

4.9. Other classification methods

This section briefly presents the review of the less popular machine learning algorithms employed in sign language recognition. These algorithms include modified Long Short-Term Memory, Bayesian classifier, Finite state machine and fuzzy logic-based, MultiLayered Perceptron, and Self Organizing Maps. Wong and Cipolla (2005) used a relevance vector machine (RVM) with the probabilistic nature of the Bayesian classifier to improve the recognition accuracy of ten gestures in complex motion analysis. The system achieved an accuracy of 91/8%. Verma and Dev (2009) proposed a Finite state machine and fuzzy logic-based method for hand gesture recognition. Their system extracted features from the images comprising 2D hand positions using the harris corner detector and clustered them together with fuzzy c-means (FCM) algorithm. The

clusters of hand posture determine the states of FSM and give the appropriate gesture recognition.

Karami et al. (2011) developed a system based on MultiLayered Perceptron (MLPNN) for the recognition of 32 classes of Persian sign language (PSL). The features for classification were extracted using discrete wavelet transform (DWT) and achieved recognition accuracy of 94.06%. Maraqa and Abu-Zaiter (2008) introduced recurrent neural networks (RNN) to recognise static Arabic signs. The proposed model attained an accuracy of 95.11%. Bhat et al. (2013) proposed Self Organizing Maps (SOM) for hand gesture recognition. Their system transformed the image into radial enclosed edges used to train SOM and achieved recognition accuracy of 92%. Lim et al. (2016) proposed a feature covariance matrix with a serial particle filter for isolated sign language recognition. The proposed American sign language recognition system achieved recognition accuracy of 87.33%. Camgoz et al. (2017) introduced a novel deep learning architecture called SubUNets. This algorithm is based on convolutional neural networks (CNNs) and Bidirectional Long Short-Term Memory (BLSTM) with connectionist temporal classification (CTC) to recognition continuous sign language. The model was trained on a one million hand gestures dataset with improved accuracy of about 30% compared with the previous state of the art system.

Abraham et al. (2019) proposed a real-time Indian sign language recognition system using LSTM based neural networks. The proposed model attained a recognition accuracy of 98% for 26 gestures. Mittal et al. (2019) designed a modified Long Short-Term Memory (LSTM) Network to recognize isolated and continuous gestures of Indian sign language. The proposed system achieved average recognition accuracy of 72.3% and 89.5% for both continuous signs and isolated sign words. Vincent et al. (2019) developed an American sign language recognition system for human activities. The system incorporates a Convolutional Neural Network (ConvNet) and Recurrent Neural Network with Long-Short Term Memory (LSTM) (Bantupalli & Xie, 2019) called DeepConvLSTM. It achieved a recognition accuracy of 91.1% with data augmentation on testing datasets. Tornay et al. (2020) introduced Kullback Leibler divergence HMM (KL-HMM) for a multilingual sign language recognition system. The performance of the system was validated using three different sign languages. Meng & Li (2021) proposed a new multiscale and dual sign language recognition Network (SLR-Net) based on a graph convolutional network (GCN) to overcome the challenge of redundant information, finger occlusion, motion blurring and variation in the ways people sign. Their model consists of three sub-modules: multi-scale attention network (MSA), multi-scale spatiotemporal attention network (MSSTA), and attention enhanced temporal convolution network (ATCN). Table 16 presents a summary of the reviewed less popular machine learning algorithms employed in sign language recognition.

5. Conclusion and future Work

With the recent advancement in machine learning and computational intelligence methods, intelligent systems in sign language recognition continue to attract academic researchers and industrial practitioners' attention. This study presents a systematic analysis of intelligent systems employed in sign language recognition related studies between 2001 and 2021. An overview of intelligent-based sign language recognition research trends is provided based on 649 full-length research articles retrieved from the Scopus database. Using the publication trends of the article retrieved from the Scopus database, this study shows that machine learning and intelligent technologies in sign language recognition are proliferating for the last 12 years. The countries and academic institutions with a large number of published articles and solid international collaborations have been identified and presented in this paper. It is ex-

pected that this study will be an opportunity for the researcher in the countries with fewer collaborations to broaden their research collaborations.

As part of this work, this review gives an insightful analysis of previous techniques used by various researchers over two decades on different stages involved in vision-based SLR, including image acquisition, image segmentation, feature extraction, and classification algorithms employed by various researchers to achieved recognition accuracy. The study also established numerous shortcomings and challenges facing a vision-based approach to sign language recognition, namely, cost of implementation, techniques, the system's accuracy, nature of the sign, complex background of the image, variation in the illumination of the image and computational time. Several devices have been used to acquire sign data, such as Dataglove, Kinect, Leap motion controller and Camera in acquiring data. Despite the fact that these devices have contributed to the performance and accuracy of the ASLR system. These devices have some shortcomings, such as high cost and inconvenience to use associated with dataglove. The image acquired from a low-resolution camera also affects the recognition accuracy of the system. Therefore, there is a need for more research that fuses images from multiple devices such as a camera, dataglove, and Kinect to acquired images to produce better results without feature extraction. Skin colour segmentation and edge detection techniques have demonstrated robust improved segmentation performance. Hybridization of two or more feature extraction techniques has also been shown to produce more robust recognition features.

Numerous approaches have been proposed on the manual acquired sign images, which involves handshapes, and remarkable success has been attained. To attained benchmark performance in this context, the following points are worthy of more attention for future research:

1. Further study on a non-manual sign involves the face region, including the movement of the head, eye blinking, eyebrow movement, and mouth shape.
2. Need to address the recognition of signs with facial expression, hand gestures and body movement simultaneously with the better recognition accuracy in real-time with improved performance. The researchers envisage that these challenges can be achieved using a deep learning approach with a high configuration system to process the input data with low computational time.
3. Different researches have been done on words, alphabets and numbers. However, there is a need for more research in future for sentences recognition in sign language.
4. Most of the work on intelligent-based sign recognition systems are at the research and prototype stage. Implementation of the proposed model will find practical application to automatic sign language.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Pal, N.R, and Pal, S.K. A review on image segmentation techiniques, 26 pattern recognition 1277 (1993). https://doi.org/10.1016/0031-3203(93)90135-J.

Abiyev, R. H., Arslan, M., & Idoko, J. B. (2020). Sign language translation using deep convolutional neural networks. *KSII Transactions on Internet and Information Systems, 14*(2), 631–653. https://doi.org/10.3837/tiis.2020.02.009.

Abraham, E., Nayak, A., & Iqbal, A. (2019). Real-time translation of indian sign language using LSTM. In *Proceedings of the Global Conference for Advancement in Technology, GCAT 2019*. https://doi.org/10.1109/GCAT47503.2019.8978343.

Adegboye, M. A., Aibinu, A. M., Kolo, J. G., Aliyu, I., Folorunso, T. A., & Lee, S. H. (2020). Incorporating intelligence in fish feeding system for dispens-

ing feed based on fish feeding intensity. *IEEE Access, 8,* 91948–91960. https: //doi.org/10.1109/ACCESS.2020.2994442.

Adeyanju, I. A., Omidiora, E. O., & Oyedokun, O. F. (2015). Performance evaluation of different support vector machine kernels for face emotion recognition. In *Proceedings of the SAI intelligent systems conference, intellisys 2015* (pp. 804–806). https://doi.org/10.1109/IntelliSys.2015.7361233.

Adithya, V., & Rajesh, R. (2020). A deep convolutional neural network approach for static hand gesture recognition. *Procedia Computer Science, 171*(2019), 2353–2361. https://doi.org/10.1016/j.procs.2020.04.255.

Adithya, V., Vinod, P. R., & Gopalakrishnan, U. (2013). Artificial neural network based method for Indian sign language recognition. In *Proceedings of the IEEE conference on information and communication technologies, ICT 2013, Ict* (pp. 1080–1085). https://doi.org/10.1109/CICT.2013.6558259.

Admasu, Y. F., & Raimond, K. (2010). Ethiopian sign language recognition using artificial neural network. In *Proceedings of the 10th international conference on intelligent systems design and applications, ISDA'10* (pp. 995–1000). https://doi.org/10.1109/ISDA.2010.5687057.

Agrawal, S. C., Jalal, A. S., & Bhatnagar, C. (2012). Recognition of Indian sign language using feature fusion. In *Proceedings of the 4th international conference on intelligent human computer interaction: advancing technology for humanity, IHCI 2012*. https://doi.org/10.1109/IHCI.2012.6481841.

Agrawal, S. C., Jalal, A. S., & Tripathi, R. K. (2016). A survey on manual and non-manual sign language recognition for isolated and continuous sign. *International Journal of Applied Pattern Recognition, 3*(2), 99. https://doi.org/10.1504/ijapr.2016.079048.

Agris, U. V., & Kraiss, K. F (2007). Towards a video corpus for signer-independent continuous sign language recognition. In *Proceedings of the 7th international workshop on gesture in human-computer interaction and simulation: 7* (pp. 10–11).

Ahmad, K., Khan, J., & Iqbal, M. S. U. D. (2019). A comparative study of different denoising techniques in digital image processing. In *Proceedings of the 8th International conference on modeling simulation and applied optimization, ICMSAO 2019* (pp. 1–6). https://doi.org/10.1109/ICMSAO.2019.8880389.

Ahmed, A. A. M., & Shah, S. M. A. (2017). Application of adaptive neuro-fuzzy inference system (ANFIS) to estimate the biochemical oxygen demand (BOD) of Surma River. *Journal of King Saud University - Engineering Sciences*. https: //doi.org/10.1016/j.jksues.2015.02.001.

Aksoy, B., & Salman, O. K. M (2020). A new hybrid filter approach for image processing. *Sakarya University Journal of Computer and Information Sciences, 3*(3), 334–342. https://doi.org/10.35377/saucis.03.03.785749.

Alamelu, J., Satej, W., & Santhosh, V. (2013). An improved k nearest neighbor classifier using interestingness measures for medical image mining. *World Academy of Science, Engineering and Technology, 7*(9), 550–554. http://waset.org/publications/16638/an-improved-k-nearest-neighbor-classifier-using-interestingness-measures-for-medical-image-mining.

Al-Hmouz, A., Shen, J., Al-Hmouz, R., & Yan, J (2012). Modeling and simulation of an adaptive neuro-fuzzy inference system (ANFIS) for mobile learning. *IEEE Transactions on Learning Technologies*. https://doi.org/10.1109/TLT.2011.36.

Aliyu, I., Gana, K. J., Musa, A. A., Adegboye, M. A., & Lim, C. G. (2020). Incorporating recognition in catfish counting algorithm using artificial neural network and geometry. *KSII Transactions on Internet and Information Systems, 14*(12), 4866–4888. https://doi.org/10.3837/tiis.2020.12.014.

Al-Jarrah, O., & Al-Omari, F. A (2007). Improving gesture recognition in the Arabic sign language using texture analysis. *Applied Artificial Intelligence, 21*(1), 11–33. https://doi.org/10.1080/08839510600938524.

Al-Jarrah, O., & Halawani, A. (2001). Recognition of gestures in Arabic sign language using neuro-fuzzy systems. *Artificial Intelligence, 133*(1–2), 117–138. https://doi.org/10.1016/S0004-3702(01)00141-2.

Alnahhas, A., Alkhatib, B., Al-Boukaee, N., Alhakim, N., Alzabibi, O., & Ajalya-keen, N (2020). Enhancing the recognition of Arabic sign language by using deep learning and leap motion controller. *International Journal of Scientific and Technology Research, 9*(4), 1865–1870.

Aloysius, N., & Geetha, M. (2020a). A scale space model of weighted average CNN ensemble for ASL fingerspelling recognition. *International Journal of Computational Science and Engineering, 22*(1), 154–161. https://doi.org/10.1504/IJCSE.2020.107268.

Aloysius, N., & Geetha, M. (2020b). Understanding vision-based continuous sign language recognition. *Multimedia Tools and Applications, 79*(31–32), 22177–22209. https://doi.org/10.1007/s11042-020-08961-z.

Al-Shamayleh, A. S., Ahmad, R., Jomhari, N., & Abushariah, M. A. M. (2020). Automatic Arabic sign language recognition: A review, taxonomy, open challenges, research roadmap and future directions. *Malaysian Journal of Computer Science, 33*(4), 306–343. https://doi.org/10.22452/mjcs.vol33no4.5.

Ameen, S., & Vadera, S. (2017). A convolutional neural network to classify American Sign Language fingerspelling from depth and colour images. *Expert Systems, 34*(3). https://doi.org/10.1111/exsy.12197.

Amza, C. (2012). *A review on neural network-based image segmentation technique.* De Montfort University, Mechanical and Manufacturing Engineering..

An, F. P., & Liu, Z. W. (2019). Medical image segmentation algorithm based on feedback mechanism CNN. *Contrast Media and Molecular Imaging.* https://doi.org/10.1155/2019/6134942.

Anand, M. S., Kumar, N. M., & Kumaresan, A. (2016). An efficient framework for Indian sign language recognition using wavelet transform. *Circuits and Systems, 07*(08), 1874–1883. https://doi.org/10.4236/cs.2016.78162.

Assaleh, K., Shanableh, T., & Zourob, M. (2012). Low complexity classification system for glove-based arabic sign language recognition. *Lecture notes in computer science* (pp. 262–268) (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 7665 LNCS(PART 3). https://doi.org/10.1007/978-3-642-34487-9_32.

Athira, P. K., Sruthi, C. J., & Lijiya, A. (2019). A signer independent sign language recognition with co-articulation elimination from live videos: An Indian scenario. *Journal of King Saud University - Computer and Information Sciences,* 0–10 xxxx. https://doi.org/10.1016/j.jksuci.2019.05.002.

Athitsos, V., Neidle, C., Sclaroff, S., Nash, J., Stefan, A., Yuan, Q., & Thangali, A. (2008). The American sign language lexicon video dataset. In *Proceedings of the IEEE computer society conference on computer vision and pattern recognition workshops, CVPR workshops.* https://doi.org/10.1109/CVPRW.2008.4563181.

Aurangzeb, K., Aslam, S., Alhussein, M., Naqvi, R. A., Arsalan, M., & Haider, S. I. (2021). Contrast enhancement of fundus images by employing modified PSO for improving the performance of deep learning models. *IEEE Access.* https://doi.org/10.1109/ACCESS.2021.3068477.

Bantupalli, K., & Xie, Y. (2019). American sign language recognition using deep learning and computer vision. In *Proceedings of the IEEE international conference on big data, big data 2018* (pp. 4896–4899). Institute of Electrical and Electronics Engineers Inc. https://doi.org/10.1109/BigData.2018.8622141.

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. https://doi.org/10.1007/s11042-020-09829-y.

Barczak, A. L. C., Reyes, N. H., Abastillas, M., Piccio, A., & Susnjak, T. (2011). A new 2D static hand gesture colour image dataset for ASL gestures. *Res. Lett. Inf. Math. Sci., 15,* 12–20. http://iims.massey.ac.nz/research/letters/.

Basu, M. (2002). Gaussian-based edge-detection methods - a survey. *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews.* https://doi.org/10.1109/TSMCC.2002.804448.

Bessa Carneiro, S., De Santos, E. D. F. D. M., De Barbosa, T. M. D. A., Ferreira, J. O., Alcalá, S. G. S. S., & Da Rocha, A. F. (2017). Static gestures recognition for Brazilian sign language with kinect sensor. In *Proceedings of IEEE sensors.* https://doi.org/10.1109/ICSENS.2016.7808522.

Bhardwaj, S., & Mittal, A. (2012). A survey on various edge detector techniques. *Procedia Technology.* https://doi.org/10.1016/j.protcy.2012.05.033.

Bhat, N. N., Venkatesh, Y. V, Karn, U., & Vig, D (2013). Hand gesture recognition using self organizing map for human computer interaction. In *Proceedings of the 2013 international conference on advances in computing, communications and informatics, ICACCI 2013* (pp. 734–738). https://doi.org/10.1109/ICACCI.2013.6637265.

Bin, L. Y., Huann, G. Y., & Yun, L. K (2019). Study of convolutional neural network in recognizing static American sign language. In *Proceedings of the IEEE international conference on signal and image processing applications, ICSIPA 2019* (pp. 41–45). https://doi.org/10.1109/ICSIPA45851.2019.8977767.

Biswas, D., Nag, A. N., Ghosh, S., Pal, A., Biswas, S. B., & Banerjee, S. (2011). Novel gray scale conversion techniques based on pixel depth. *Journal of Global Research in Computer Science, 2*(6), 118–121.

Bobic, V., Tadic, P., & Kvascev, G. (2016). Hand gesture recognition using neural network based techniques. In *Proceedings of the 13th symposium on neural networks and applications, NEUREL 2016* (pp. 14–17). https://doi.org/10.1109/NEUREL.2016.7800104.

Brand, M., Oliver, N., & Pentland, A. (1997). Coupled hidden Markov models for complex action recognition. In *Proceedings of the IEEE computer society conference on computer vision and pattern recognition.* https://doi.org/10.1109/cvpr.1997.609450.

Butt, U. M., Husnain, B., Ahmed, U., Tariq, A., Tariq, I., Butt, M. A., & Zia, M. S (2019). Feature based algorithmic analysis on American sign language dataset. *International Journal of Advanced Computer Science and Applications, 10*(5), 583–589. https://doi.org/10.14569/ijacsa.2019.0100575.

Buyuksahin, U. (2014). Vision-based sensor technologies - webcam: A multifunction sensor. *Comprehensive materials processing.* https://doi.org/10.1016/B978-0-08-096532-1.01314-5.

Camgoz, N. C., Hadfield, S., Koller, O., & Bowden, R. (2017). SubUNets: End-to-end hand shape and continuous sign language recognition. In *Proceedings of the IEEE international conference on computer vision* (pp. 3075–3084). 2017-Octob. https://doi.org/10.1109/ICCV.2017.332.

Canny, J. (1986). A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence.* https://doi.org/10.1109/TPAMI.1986.4767851.

Cebeci, Z., & Yildiz, F (2015). Comparison of K-means and fuzzy C-means algorithms on different cluster structures. *Journal of Agricultural Informatics.* https://doi.org/10.17700/jai.2015.6.3.196.

Cheng, H. D., Jiang, X. H., & Wang, J (2002). Color image segmentation based on homogram thresholding and region merging. *Pattern Recognition, 35*(2), 373–393. https://doi.org/10.1016/S0031-3203(01)00054-1.

Cheok, M. J., Omar, Z., & Jaward, M. H. (2019). A review of hand gesture and sign language recognition techniques. *International Journal of Machine Learning and Cybernetics, 10*(1), 131–153. https://doi.org/10.1007/s13042-017-0705-5.

Chhabra, AChinu. (2014). Overview and comparative analysis of edge detection techniques in digital image processing. *International Journal of Information and Computation Technology.*

Chong, T. W., & Lee, B. G. (2018). American sign language recognition using leap motion controller with machine learning approach. *Sensors (Switzerland), (10),* 18. https://doi.org/10.3390/s18103554.

Chourasiya, A., Khare, N., Maini, R., Aggarwal, H., Chourasiya, A., & Khare, N. (2019). A comprehensive review of image enhancement techniques. *International Jour-*

17th IEEE international conference on computational science and engineering, CSE 2014, Jointly with 13th IEEE international conference on ubiquitous computing and communications, IUCC 2014, 13th international symposium on pervasive systems (pp. 21–26). https://doi.org/10.1109/CSE.2014.38.

Jielai, X., Hongwei, J., & Qiyi, T. (2015). Introduction to artificial neural networks. *Advanced Medical Statistics*, 1431–1449. https://doi.org/10.1142/9789814583312_0037.

Jin, C. M., Omar, Z., & Jaward, M. H. (2016). A mobile application of American sign language translation via image processing algorithms. In *Proceedings of the 2016 IEEE region 10 symposium, TENSYMP 2016* (pp. 104–109). https://doi.org/10.1109/TENCONSpring.2016.7519386.

Joshi, G., Singh, S., & Vig, R. (2020). Taguchi-TOPSIS based HOG parameter selection for complex background sign language recognition. *Journal of Visual Communication and Image Representation, 71*, Article 102834. https://doi.org/10.1016/j.jvcir.2020.102834.

Kadhim, R. A., & Khamees, M. (2020). A real-time american sign language recognition system using convolutional neural network for real datasets. *TEM Journal*. https://doi.org/10.18421/TEM93-14.

Kalhor, M., Kajouei, A., Hamidi, F., & Asem, M. M. (2019). Assessment of histogram-based medical image contrast enhancement techniques: An implementation. In *Proceedings of the IEEE 9th annual computing and communication workshop and conference, CCWC 2019*. https://doi.org/10.1109/CCWC.2019.8666468.

Kaluri, R., & Reddy, P. C. (2016a). A framework for sign gesture recognition using improved genetic algorithm and adaptive filter. *Cogent Engineering, 3*(1), Article 1251730. https://doi.org/10.1080/23311916.2016.1251730.

Kaluri, R., & Reddy, P. C. (2016b). Sign gesture recognition using modified region growing algorithm and adaptive genetic fuzzy classifier. *International Journal of Intelligent Engineering and Systems, 9*(4), 225–233. https://doi.org/10.22266/ijies2016.1231.24.

Kaluri, R., & Reddy, P. C. (2017). An enhanced framework for sign gesture recognition using hidden markov model and adaptive histogram technique. *International Journal of Intelligent Engineering and Systems, 10*(3), 11–19. https://doi.org/10.22266/ijies2017.0630.02.

Kamal, S. M., Chen, Y., Li, S., Shi, X., & Zheng, J. (2019). Technical approaches to Chinese sign language processing: A review. *IEEE Access, 7*, 96926–96935. https://doi.org/10.1109/ACCESS.2019.2929174.

Kamishima, T., Hamasaki, M., & Akaho, S (2009). TrBagg: A simple transfer learning method and its application to personalization in collaborative tagging. In *Proceedings of the IEEE international conference on data mining, ICDM* (pp. 219–228). https://doi.org/10.1109/ICDM.2009.9.

Kanezaki, A. (2018). Unsupervised image segmentation by backpropagation. In *Proceedings of the ICASSP, IEEE international conference on acoustics, speech and signal processing*. https://doi.org/10.1109/ICASSP.2018.8462533.

Karami, A., Zanj, B., & Sarkaleh, A. K. (2011). Persian sign language (PSL) recognition using wavelet transform and neural networks. *Expert Systems with Applications, 38*(3), 2661–2667. https://doi.org/10.1016/j.eswa.2010.08.056.

Karamizadeh, S., Abdullah, S. M., Manaf, A. A., Zamani, M., & Hooman, A. (2013). An overview of principal component analysis. *Journal of Signal and Information Processing*. https://doi.org/10.4236/jsip.2013.43b031.

Kasmin, F., Rahim, N. S. A., Othman, Z., Ahmad, S. S. S., & Sahri, Z. (2020). A comparative analysis of filters towards sign language recognition. *International Journal of Advanced Trends in Computer Science and Engineering*. https://doi.org/10.30534/ijatcse/2020/84942020.

Kassem, Y., Çamur, H., & Esenel, E. (2017). Adaptive neuro-fuzzy inference system (ANFIS) and response surface methodology (RSM) prediction of biodiesel dynamic viscosity at 313K. *Procedia Computer Science*. https://doi.org/10.1016/j.procs.2017.11.274.

Kaur, A., & Kaur, A. (2012). Comparison of mamdani-type and sugeno-type fuzzy inference systems for air conditioning system. *International Journal of Soft Computing and Engineering*.

Kaur, M., & Chand, L. (2018). Review of image segmentation and its techniques. *Journal of Emerging Technologies and Innovative Research (JETIR), 5*(7), 974–981.

Kaur, S. (2015). Noise types and various removal techniques. *Nternational Journal of Advanced Research in Electroni Cs and Communication Engineering (IJARECE), 4*(2), 226–230.

Kaur', D., & Kaur, Y (2014). Various image segmentation techniques: A review. *International Journal of Computer Science and Mobile Computing*.

Kausar, S., Javed, M., & Sohail, S. (2008). Recognition of gestures in Pakistani sign language using fuzzy classifier. In *Proceedings of the 8th international conference on signal processing, computational geometry and artificial vision (Iscgav'08)* (pp. 101–105). http://www.wseas.us/e-library/conferences/2008/rhodes/iscgav/iscgav16.pdf.

Khalid, S., Khalil, T., & Nasreen, S. (2014). A survey of feature selection and feature extraction techniques in machine learning. In *Proceedings of the 2014 science and information conference, SAI 2014* (pp. 372–378). https://doi.org/10.1109/SAI.2014.6918213.

Khan, A. M (2013). Image segmentation methods: A comparative study. *International Journal of Soft Computing and Engineering (IJSCE)*.

Khan, W. (2014). Image segmentation techniques: A survey. *Journal of Image and Graphics*. https://doi.org/10.12720/joig.1.4.166-170.

Khari, M., Garg, A. K., Gonzalez-Crespo, R., & Verdú, E. (2019). Gesture recognition of RGB and RGB-D static images using convolutional neural networks. *International Journal of Interactive Multimedia and Artificial Intelligence, 5*(7), 22. https://doi.org/10.9781/ijimai.2019.09.002.

Kim, S., Kim, J., Ahn, S., & Kim, Y. (2018). Finger language recognition based on ensemble artificial neural network learning using armband EMG sensors. *Technology and Health Care, 26*(S1), S249–S258. https://doi.org/10.3233/THC-174602.

Kiselev, V., Khlamov, M., & Chuvilin, K. (2019). Hand gesture recognition with multiple leap motion devices. In *Proceedings of the conference of open innovation association, FRUCT* (pp. 163–169). 2019-April. https://doi.org/10.23919/FRUCT.2019.8711887.

Kishore, P. V., Kumar, D. A., Goutham, E. N. D., & Manikanta, M (2016). Continuous sign language recognition from tracking and shape features using Fuzzy Inference Engine. In *Proceedings of the 2016 IEEE international conference on wireless communications, signal processing and networking, WiSPNET 2016* (pp. 2165–2170). https://doi.org/10.1109/WiSPNET.2016.7566526.

Kishore, P. V. V., Prasad, M. V. D., Prasad, C. R., & Rahul, R (2015). 4-Camera model for sign language recognition using elliptical Fourier descriptors and ANN. In *Proceedings of the international conference on signal processing and communication engineering systems, SPACES 2015, in Association with IEEE,* (pp. 34–38). https://doi.org/10.1109/SPACES.2015.7058288.

Kishore, P. V. V., & Kumar, Rajesh (2012). A video based Indian sign language recognition system (INSLR) using wavelet transform and fuzzy logic. *International Journal of Engineering and Technology, 4*(5), 537–542. https://doi.org/10.7763/ijet.2012.v4.427.

Koller, O., Zargaran, S., Ney, H., & Bowden, R. (2018). Deep sign: Enabling robust statistical continuous sign language recognition via hybrid CNN-HMMs. *International Journal of Computer Vision, 126*(12), 1311–1325. https://doi.org/10.1007/s11263-018-1121-3.

Kong, W. W., & Ranganath, S. (2014). Towards subject independent continuous sign language recognition: A segment and merge approach. *Pattern Recognition, 47*(3), 1294–1308. https://doi.org/10.1016/j.patcog.2013.09.014.

Korzynska, A., Roszkowiak, L., Lopez, C., Bosch, R., Witkowski, L., & Lejeune, M. (2013). Validation of various adaptive threshold methods of segmentation applied to follicular lymphoma digital images stained with 3,3'-DiaminobenzidineandHaematoxylin. *Diagnostic Pathology*. https://doi.org/10.1186/1746-1596-8-48.

Kotsiantis, S. B., Zaharakis, I. D., & Pintelas, P. E. (2006). Machine learning: A review of classification and combining techniques. *Artificial Intelligence Review, 26*(3), 159–190. https://doi.org/10.1007/s10462-007-9052-3.

Krishnaveni, M., Subashini, P., & Dhivyaprabha, T. T. (2019). An assertive framework for automatic tamil sign language recognition system using computational intelligence. *Intelligent systems reference library*: 150. Springer International Publishing. https://doi.org/10.1007/978-3-319-96002-9_3.

Kudrinko, K., Flavin, E., Zhu, X., & Li, Q. (2021). Wearable sensor-based sign language recognition: A comprehensive review. *IEEE Reviews in Biomedical Engineering, 14*, 82–97. https://doi.org/10.1109/RBME.2020.3019769.

Kumar, G., & Bhatia, P. K. (2014). A detailed review of feature extraction in image processing systems. In *Proceedings of the international conference on advanced computing and communication technologies, ACCT, February* (pp. 5–12). https://doi.org/10.1109/ACCT.2014.74.

Kumar, P., Gauba, H., Roy, P. P., & Dogra, D. P. (2017). Coupled HMM-based multi-sensor data fusion for sign language recognition. *Pattern Recognition Letters, 86*, 1–8. https://doi.org/10.1016/j.patrec.2016.12.004.

Kumarage, D., Fernando, S., Fernando, P., Madushanka, D., & Samarasinghe, R. (2011). Real-time sign language gesture recognition using still-image comparison and motion recognition. In *Proceedings of the 6th international conference on industrial and information systems, ICIIS 2011* (pp. 169–174). https://doi.org/10.1109/ICIINFS.2011.6038061.

Lahiani, H., Elleuch, M., & Kherallah, M. (2016). Real time hand gesture recognition system for android devices. In *Proceedings of the international conference on intelligent systems design and applications, ISDA*. https://doi.org/10.1109/ISDA.2015.7489184.

Lan, Y., Zhou, D., Zhang, H., & Lai, S. (2017). Development of early warning models. *Early warning for infectious disease outbreak: Theory and practice*. https://doi.org/10.1016/B978-0-12-812343-0.00003-5.

Lech, M., & Kostek, B. (2012). Hand gesture recognition supported by fuzzy rules and Kalman filters. *International Journal of Intelligent Information and Database Systems, 6*(5), 407–420. https://doi.org/10.1504/IJIIDS.2012.049304.

Lee, C. K. M., Ng, K. H., Chen, C. H., Lau, H. C. W., Chung, S. Y., & Tsoi, T. (2021). American sign language recognition and training method with recurrent neural network. *Expert Systems with Applications, 167*(August 2020), Article 114403. https://doi.org/10.1016/j.eswa.2020.114403.

Lee, G. C., Yeh, F. H., & Hsiao, Y. H. (2016). Kinect-based Taiwanese sign-language recognition system. *Multimedia Tools and Applications, 75*(1), 261–279. https://doi.org/10.1007/s11042-014-2290-x.

Lee, K., Jin, K., Kim, Y., Lee, J. H., & Lee, E. C. (2021). A comparative analysis on the impact of face tracker and skin segmentation onto improving the performance of real-time remote photoplethysmography. *Lecture notes in computer science* (pp. 27–37) (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 12616 LNCS. https://doi.org/10.1007/978-3-030-68452-5_3.

Lee, S. U., Yoon Chung, S., & Park, R. H. (1990). A comparative performance study of several global thresholding techniques for segmentation. *Computer Vision, Graphics, and Image Processing, 52*(2), 171–190. https://doi.org/10.1016/0734-189X(90)90053-X.

Lemaster, B., & Monaghan, L. (2007). Variation in Sign Languages. *A companion to linguistic anthropology*. https://doi.org/10.1002/9780470996522.ch7.

Lenard, J. G., Pietrzyk, M., & Cser, L. (1999). Chapter 9 - Knowledge Based Modeling. *Mathematical and physical simulation of the properties of hot rolled products*

(pp. 279–318) In J. G. Lenard, M. Pietrzyk, and L. Cser (Eds.). Elsevier Science Ltd. https://doi.org/10.1016/B978-008042701-0/50009-7.

Li, S. Z., Yu, B., Wu, W., Su, S. Z., & Ji, R. R (2015). Feature learning based on SAE-PCA network for human gesture recognition in RGBD images. *Neurocomputing, 151*(P2), 565–573. https://doi.org/10.1016/j.neucom.2014.06.086.

Li, T. H. S, Kao, M. C., & Kuo, P. H (2016). Recognition system for home-service-related sign language using entropy-based K-means algorithm and ABC-Based HMM. *IEEE Transactions on Systems, Man, and Cybernetics: Systems, 46*(1), 150–162. https://doi.org/10.1109/TSMC.2015.2435702.

Lim, K. M., Tan, A. W. C. C., & Tan, S. C. (2016). A feature covariance matrix with serial particle filter for isolated sign language recognition. *Expert Systems with Applications, 54*, 208–218. https://doi.org/10.1016/j.eswa.2016.01.047.

Lionnie, R., Timotius, I. K., & Setyawan, I. (2012). Performance comparison of several pre-processing methods in a hand gesture recognition system based on nearest neighbor for different background conditions. *ITB Journal of Information and Communication Technology, 6*(3), 183–194. https://doi.org/10.5614/itbj.ict.2012.6.3.1.

Liu, X., Liu, Z., Wang, G., Cai, Z., & Zhang, H. (2017). Ensemble transfer learning algorithm. *IEEE Access, 6*, 2389–2396. https://doi.org/10.1109/ACCESS.2017.2782884.

Liwicki, S., & Everingham, M. (2009). Automatic recognition of fingerspelled words in british sign language. In *Proceedings of the 2009 IEEE conference on computer vision and pattern recognition, CVPR 2009* (pp. 50–57). https://doi.org/10.1109/CVPR.2009.5204291.

Longkumer, N., Kumar, M., & Saxena, S. (2014). Contrast enhancement techniques using histogram equalization : A survey. *International Journal of Current Engineering and Technology, 4*(3), 1561–1565.

Lowe, G. (2004). SIFT - The scale invariant feature transform. *International Journal.*

Mahmud, I., Tabassum, T., Uddin, M. P., Ali, E., Nitu, A. M., & Afjal, M. I. (2019). Efficient noise reduction and HOG feature extraction for sign language recognition. In *Proceedings of the international conference on advancement in electrical and electronic engineering, ICAEEE 2018* (pp. 1–4). https://doi.org/10.1109/ICAEEE.2018.8642983.

Maini, R., & Aggarwal, H. (2009). Study and comparison of various image edge detection techniques. *International Journal of Image Processing.*

Maini, R., & Aggarwal, H. (2010). A comprehensive review of image enhancement techniques. *Journal of Computing, 2*(3), 8–13.

Majeed, S. H., & Isa, N. A. M. (2021). Adaptive entropy index histogram equalization for poor contrast images. *IEEE Access, 9*, 6402–6437. https://doi.org/10.1109/ACCESS.2020.3048148.

Mamdani, E. H., & Assilian, S (1975). An experiment in linguistic synthesis with a fuzzy logic controller. *International Journal of Man-Machine Studies.* https://doi.org/10.1016/S0020-7373(75)80002-2.

Maraqa, M., & Abu-Zaiter, R. (2008). Recognition of Arabic sign language (ArSL) using recurrent neural networks. In *Proceedings of the 1st international conference on the applications of digital information and web technologies, ICADIWT 2008* (pp. 478–481). https://doi.org/10.1109/ICADIWT.2008.4664396.

Martínez, A. M., Wilbur, R. B., Shay, R., & Kak, A. C. (2002). Purdue RVL-SLLL ASL database for automatic recognition of American sign language. In *Proceedings of the 4th IEEE international conference on multimodal interfaces, ICMI 2002* (pp. 167–172). https://doi.org/10.1109/ICMI.2002.1166987.

Martiskainen, P., Järvinen, M., Skön, J.-P., Tiirikainen, J., Kolehmainen, M., & Mononen, J. (2009). Cow behaviour pattern recognition using a three-dimensional accelerometer and support vector machines. *Applied Animal Behaviour Science, 119*(1), 32–38. https://doi.org/10.1016/j.applanim.2009.03.005.

Maru, M., & Parikh, C. M. (2017). Image restoration techniques: A survey. *International Journal of Computer Applications, 160*(6), 15–19. https://doi.org/10.5120/ijca2017913060.

Mehdi, S. A., & Khan, Y. N. (2002). Sign language recognition using sensor gloves. In *Proceedings of the ICONIP, 9th international conference on neural information processing: Computational intelligence for the E-Age: 5* (pp. 2204–2206). https://doi.org/10.1109/ICONIP.2002.1201884.

Mehnert, A., & Jackway, P. (1997). An improved seeded region growing algorithm. *Pattern Recognition Letters.* https://doi.org/10.1016/S0167-8655(97)00131-1.

Mekala, P., Gao, Y., Fan, J., & Davari, A. (2011). Real-time sign language recognition based on neural network architecture. In *Proceedings of the annual Southeastern symposium on system theory* (pp. 195–199). https://doi.org/10.1109/SSST.2011.5753805.

Meng, L., & Li, R. (2021). An attention-enhanced multi-scale and dual sign language recognition network based on a graph convolution network. *Sensors (Switzerland), 21*(4), 1–22. https://doi.org/10.3390/s21041120.

Merenda, M., Porcaro, C., & Iero, D (2020). Edge machine learning for ai-enabled iot devices: A review. *Sensors (Switzerland)*, (9), 20. https://doi.org/10.3390/s20092533.

Merin Joy, D., & Rajesh, D. M. (2017). Hand gesture recognition from surveillance video using surf feature. *International Journal of Engineering Research and Science, 3*(5), 82–89. https://doi.org/10.25125/engineering-journal-ijoer-may-2017-31.

Mistry, D., & Banerjee, A. (2017). Comparison of feature detection and matching approaches: SIFT and SURF. *GRD Journals- Global Research and Development Journal for Engineering.*

Mittal, A., Kumar, P., Roy, P. P., Balasubramanian, R., & Chaudhuri, B. B. (2019). A modified LSTM model for continuous sign language recognition using leap motion. *IEEE Sensors Journal, 19*(16), 7056–7063. https://doi.org/10.1109/JSEN.2019.2909837.

Moghaddam, M. J., & Soltanian-Zadeh, H. (2011). Medical image segmentation using artificial neural networks. *Artificial neural networks - Methodological advances and biomedical applications.* https://doi.org/10.5772/16103.

Mohandes, M. A. (2013). Recognition of two-handed arabic signs using the cyberglove. *Arabian Journal for Science and Engineering, 38*(3), 669–677. https://doi.org/10.1007/s13369-012-0378-z.

Moreira Almeida, S. G., Guimarães, F. G., & Arturo Ramírez, J. (2014). Feature extraction in Brazilian sign language recognition based on phonological structure and using RGB-D sensors. *Expert Systems with Applications, 41*(16), 7259–7271. https://doi.org/10.1016/j.eswa.2014.05.024.

Mufarrohi, F. A., & Utaminingrum, F. (2017). Hand gesture recognition using adaptive network based fuzzy inference system and K-nearest neighbor. *International Journal of Technology, 8*(3), 559–567. https://doi.org/10.14716/ijtech.v8i3.3146.

Mustafa, M. B., & Lahsasna, A. (2016). An accurate sign language recognizer using random sphere oracle. In *Proceedings of the 2016 1st international conference on multimedia and image processing, ICMIP 2016* (pp. 36–40). https://doi.org/10.1109/ICMIP.2016.15.

Muthukrishnan, R., & Radha, M. (2011). Edge detection techniques for image segmentation. *International Journal of Computer Science and Information Technology.* https://doi.org/10.5121/ijcsit.2011.3620.

Narayanan, A., Sharmila, T. S., & Gajalakshmi, P. (2018). Word sign recognition of invariant images based on SURF with Laplacian eigenmaps. In *Proceedings of the 2nd international conference on computer, communication, and signal processing: Special focus on technology and innovation for smart environment, ICCCSP 2018.* https://doi.org/10.1109/ICCCSP.2018.8452825.

Nedeljkovic, I., Vi, C., & Vi, W. G. (2004). Image classification based on fuzzy logic. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 34*(30), 1–6.

Nelson, A., Price, K. J., & Multari, R. (2019). ASL reverse dictionary - ASL translation using deep learning. *SMU Data Science Review, 2*(1). https://scholar.smu.edu/datasciencereviewhttp://digitalrepository.smu.edu.Availableat:https://scholar.smu.edu/datasciencereview/vol2/iss1/21.

Ngai, E. W. T., Peng, S., Alexander, P., & Moon, K. K. L. (2014). Decision support and intelligent systems in the textile and apparel supply chain: An academic review of research articles. *Expert Systems with Applications, 41*(1), 81–91. https://doi.org/10.1016/j.eswa.2013.07.013.

Nguen, N. T., Sako, S., & Kwolek, B. (2019). Deep CNN-based recognition of JSL finger spelling. In C. L. M. Q. P. H. C. R. E. Perez Garcia H. Sanchez Gonzalez L. (Ed.). *Lecture notes in computer science.* Springer International Publishing (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics): Vol. 11734 LNAI. https://doi.org/10.1007/978-3-030-29859-3_51.

Niblack, W. (1986). An introduction to digital image processing. *An introduction to digital image processing.* https://doi.org/10.1002/0470035528.ch15.

Nisha, S. S., & Meeral, N. M. (2021). Applications of deep learning in biomedical engineering. *Handbook of deep learning in biomedical engineering.* https://doi.org/10.1016/b978-0-12-823014-5.00008-9.

Nuhu, B. K., Aliyu, I., Adegboye, M. A., Ryu, J. K., Olaniyi, O. M., & Lim, C. G. (2021). Distributed network-based structural health monitoring expert system. *Building Research and Information, 49*(1), 144–159. https://doi.org/10.1080/09613218.2020.1854083.

Oliveira, J., Sutherland, A., & Farouk, M. (2017). Two-stage PCA with interpolated data for hand shape recognition in sign language. In *Proceedings of the applied imagery pattern recognition workshop.* https://doi.org/10.1109/AIPR.2016.8010587.

Oyewumi Jimoh, K., Morufat Adepoju, T., Sobowale, A. A., & Ayilara, O. A (2018). Offline gesture recognition system for yorùbá numeral counting. *Asian Journal of Research in Computer Science*, 1–11 September 2018. https://doi.org/10.9734/ajrcos/2018/v1i424753.

Pan, J., Luo, Y., Li, Y., Tham, C.-K., Heng, C.-H., & Thean, A. V. Y. (2020). A wireless multi-channel capacitive sensor system for efficient glove-based gesture recognition with AI at the edge. *IEEE Transactions on Circuits and Systems II: Express Briefs, 67*(9), 1624–1628. https://doi.org/10.1109/TCSII.2020.3010318.

Pan, T. Y, Lo, L. Y, Yeh, C. W, Li, J. W, Liu, H. T., & Hu, M. C (2016). Real-time sign language recognition in complex background scene based on a hierarchical clustering classification method. In *Proceedings of the IEEE 2nd international conference on multimedia big data, BigMM 2016* (pp. 64–67). https://doi.org/10.1109/BigMM.2016.44.

Pansare, J R, Dhumal, H., Babar, S., and International, K.S. (2015). Real time static hand gesture recognition system in complex background that uses number system of indian sign language static hand gesture recognition for sign language alphabets using edge oriented histogram and multi class SVM real - Time hand gesture R. 2(3), 2014–2015.

Pansare, J. R., Gawande, S. H., & Ingle, M. (2012). Real-time static hand gesture recognition for American sign language (ASL) in complex background. *Journal of Signal and Information Processing, 03*(03), 364–367. https://doi.org/10.4236/jsip.2012.33047.

Pansare, Jayshree R., & Ingle, M. (2016). Vision-based approach for American sign language recognition using edge orientation histogram. In *Proceedings of the international conference on image, vision and computing, ICIVC 2016* (pp. 86–90). https://doi.org/10.1109/ICIVC.2016.7571278.

Panwar, P., Gopal, G., & Kumar, R. (2016). Image Segmentation using K-means clustering and thresholding image segmentation using K-means clustering and thresholding. *International Research Journal of Engineering and Technology.*

Patel, J. M., & Gamit, N. C. (2016). A review on feature extraction techniques in content based image retrieval. In *Proceedings of the 2016 IEEE international conference on wireless communications, signal processing and networking, WiSPNET 2016* (pp. 2259–2263). https://doi.org/10.1109/WiSPNET.2016.7566544.

Patil, S. B., & Sinha, G. R. (2017). Distinctive feature extraction for Indian sign language (ISL) gesture using scale invariant feature transform (SIFT). *Journal of The*

Institution of Engineers (India): Series B, 98(1), 19–26. https://doi.org/10.1007/s40031-016-0250-8.

Perumal, S., & Velmurugan, T. (2018). Preprocessing by contrast enhancement techniques for medical images. *International Journal of Pure and Applied Mathematics, 118*(18), 3681–3688.

Phi, L. T., Nguyen, H. D., Bui, T. T. Q., & Vu, T. T. (2015). A glove-based gesture recognition system for Vietnamese sign language. In *Proceedings of the ICCAS, 15th international conference on control, automation and systems* (pp. 1555–1559). https://doi.org/10.1109/ICCAS.2015.7364604.

Polikar, R. (2012). Ensemble machine learning. *Ensemble machine learning.* https://doi.org/10.1007/978-1-4419-9326-7.

Porfirio, A. J., Wiggers, K. L., Oliveira, L. E. S., & Weingaertner, D (2013). LIBRAS sign language hand configuration recognition based on 3D meshes. In *Proceedings of the IEEE international conference on systems, man, and cybernetics, SMC 2013* (pp. 1588–1593). https://doi.org/10.1109/SMC.2013.274.

Prasad, M. V. D., Kishore, P. V. V., Kiran Kumar, E., & Anil Kumar, D. (2016). Indian sign language recognition system using new fusion based edge operator. *Journal of Theoretical and Applied Information Technology, 88*(3), 574–584. https://www.scopus.com/inward/record.uri?eid=2-s2.0-84976576556andpartnerID=40andmd5=6057d7251cae34213b054e885cc97967.

Prasad, M. V. D., Kishore, P. V. V., Anil Kumar, D., & Prasad, C. R (2016). Fuzzy classifier for continuous sign language recognition from tracking and shape features. *Indian Journal of Science and Technology*, (30), 9. https://doi.org/10.17485/ijst/2016/v9i30/98726.

Raghuveera, T., Deepthi, R., Mangalashri, R., & Akshaya, R. (2020). A depth-based Indian sign language recognition using Microsoft Kinect. *Sadhana - Academy Proceedings in Engineering Sciences, 45*(1), 1–13. https://doi.org/10.1007/s12046-019-1250-6.

Raheja, J. L., Mishra, A., & Chaudhary, A. (2016). Indian sign language recognition using SVM. *Pattern Recognition and Image Analysis, 26*(2), 434–441. https://doi.org/10.1134/S1054661816020164.

Rahim, M A, Shin, J., & Yun, K. S (2020). Hand gesture-based sign alphabet recognition and sentence interpretation using a convolutional neural network. *Annals of Emerging Technologies in Computing, 4*(4), 20–27. https://doi.org/10.33166/AETiC.2020.04.003.

Rahim, M. A., Islam, M. R., & Shin, J (2019). Non-touch sign word recognition based on dynamic hand gesture using hybrid segmentation and CNN feature fusion. *Applied Sciences (Switzerland)*, (18), 9. https://doi.org/10.3390/app9183790.

Raj, R. D., & Jasuja, A. (2018). British sign language recognition using HOG. In *Proceedings of the IEEE international students' conference on electrical, electronics and computer science, SCEECS 2018* (pp. 1–4). https://doi.org/10.1109/SCEECS.2018.8546967.

Raj, R., & Joseph, N (2016). Keypoint Extraction Using SURF Algorithm for CMFD, *Procedia computer science* (93, pp. 375–381) In J. J. Krishna D.D. Mathew J. (Ed.). Elsevier B.V. https://doi.org/10.1016/j.procs.2016.07.223.

Ramos, A. L. A., Dalhag, G. D. M., Daygon, M. L. D., Omar, J., Cruz, K. D. L., Macaranas, A. A., & Almodovar, K. L. J. V (2019). Alphabet hand gesture recognition using histogram of oriented gradients, support vector machine and K – nearest neighbor algorithm. *International Research Journal of Computer Science, 6*(04), 200–205.

Rao, G. A., & Kishore, P. V. V. (2018). Selfie video based continuous Indian sign language recognition system. *Ain Shams Engineering Journal.* https://doi.org/10.1016/j.asej.2016.10.013.

Rashid, O., Al-Hamadi, A., & Michaelis, B. (2009). A framework for the integration of gesture and posture recognition using HMM and SVM. In *Proceedings of the IEEE international conference on intelligent computing and intelligent systems, ICIS 2009: 4* (pp. 572–577). https://doi.org/10.1109/ICICISYS.2009.5357615.

Rashmi, Kumar, M., & Saxena, R (2013). Algorithm and technique on various edge detection : A survey. *Signal and Image Processing : An International Journal.* https://doi.org/10.5121/sipij.2013.4306.

Rastgoo, R., Kiani, K., & Escalera, S. (2021a). Hand pose aware multimodal isolated sign language recognition. *Multimedia Tools and Applications, 80*(1), 127–163. https://doi.org/10.1007/s11042-020-09700-0.

Rastgoo, R., Kiani, K., & Escalera, S. (2021b). Sign language recognition: A deep survey. *Expert Systems with Applications, 164*(February 2020), Article 113794. https://doi.org/10.1016/j.eswa.2020.113794.

Ravi, S., Maloji, S., Polurie, V. V. K., & Eepuri, K. K. (2018). Sign language recognition with multi feature fusion and ANN classifier. *Turkish Journal of Electrical Engineering and Computer Sciences, 26*(6), 2871–2885. https://doi.org/10.3906/elk-1711-139.

Razmjooy, N., Razmjooy, S., Vahedi, Z., Estrela, V. V, & de Oliveira, G. G. (2021). Skin color segmentation based on artificial neural network improved by a modified grasshopper optimization algorithm. *Lecture Notes in Electrical Engineering, 696*, 169–185. https://doi.org/10.1007/978-3-030-56689-0_9.

Reeves, S.J. (2014). Image restoration: Fundamentals of image restoration. 10.1016/b978-0-12-396501-1.00006-6

Rekha, J., Bhattacharya, J., & Majumder, S. (2011a). Hand gesture recognition for sign language: A new hybrid approach. In *Proceedings of the international conference on image processing, computer vision, and pattern recognition, IPCV 2011: 1* (pp. 80–86).

Rekha, J., Bhattacharya, J., & Majumder, S. (2011b). Shape, texture and local movement hand gesture features for indian sign language recognition. In *Proceedings of the TISC, 3rd international conference on trendz in information sciences and computing* (pp. 30–35). https://doi.org/10.1109/TISC.2011.6169079.

Rogowska, J. (2009). Overview and fundamentals of medical image segmentation.

Handbook of medical image processing and analysis (2nd Edition). Elsevier Inc. https://doi.org/10.1016/B978-012373904-9.50013-1.

Rubini, C., & Pavithra, N. (2019). Contrast Enhancement of MRI images using AHE and CLAHE techniques. *International Journal of Innovative Technology and Exploring Engineering, 9*(2), 2442–2445. https://doi.org/10.35940/ijitee.b7017.129219.

Ryumin, D., Ivanko, D., Axyonov, A., Kagirov, I., Karpov, A., & Zelezny, M. (2019). Human-robot interaction with smart shopping trolley using sign language: Data collection. In *Proceedings of the IEEE international conference on pervasive computing and communications workshops, PerCom Workshops* (pp. 949–954). https://doi.org/10.1109/PERCOMW.2019.8730886.

Saha, H. N., Tapadar, S., Ray, S., Chatterjee, S. K., & Saha, S. (2018). A machine learning based approach for hand gesture recognition using distinctive feature extraction. In *Proceedings of the IEEE 8th annual computing and communication workshop and conference, CCWC* (pp. 91–98). 2018, 2018-Janua. https://doi.org/10.1109/CCWC.2018.8301631.

Saha, S. (2018). A comprehensive guide to convolutional neural networks — the ELI5 way | by Sumit Saha | Towards data science.

Sahoo, A. K., Mishra, G. S., & Ravulakollu, K. K. (2014). Sign language recognition: State of the art. *ARPN Journal of Engineering and Applied Sciences, 9*(2), 116–134.

Saleh, Y., & Issa, G. F. (2020). Arabic sign language recognition through deep neural networks fine-tuning. *International Journal of Online and Biomedical Engineering, 16*(5), 71–83. https://doi.org/10.3991/IJOE.V16I05.13087.

Sallam, A. A., Kabir, M. N., Shamhan, A. N. M., Nasser, H. K., & Wang, J (2021). A racial recognition method based on facial color and texture for improving demographic classification, *Lecture notes in electrical engineering*, (666, pp. 843–852). https://doi.org/10.1007/978-981-15-5281-6_61.

Savur, C., & Sahin, F. (2016). Real-time American sign language recognition system using surface EMG signal. In *Proceedings of the IEEE 14th international conference on machine learning and applications, ICMLA 2015* (pp. 497–502). https://doi.org/10.1109/ICMLA.2015.212.

Savur, C., & Sahin, F. (2017). American sign language recognition system by using surface EMG signal. In *Proceedings of the IEEE international conference on systems, man, and cybernetics, SMC 2016* (pp. 2872–2877). https://doi.org/10.1109/SMC.2016.7844675.

Schembri, A., Fenlon, J., Rentelis, R., Reynolds, S., & Cormier, K. (2013). Building the British sign language corpus. *Language Documentation and Conservation, 7*, 136–154.

Scott, S.M., O'Hare, W.T., and Ali, Z. (2003). Fuzzy logic and fuzzy classification techniques. 95–134. 10.1007/978-3-540-36213-5_5

Selim, S., Elhinamy, E., Othman, H., Abouelsaadat, W., & Salem, M. A. M. (2019). A review of machine learning approaches for epileptic seizure prediction. In *Proceedings of the ICCES, 14th international conference on computer engineering and systems* (pp. 239–244). Institute of Electrical and Electronics Engineers Inc. https://doi.org/10.1109/ICCES48960.2019.9068190.

Senthilkumaran, N., & Vaithegi, S. (2016). Image segmentation by using thresholding techniques for medical images. *Computer Science and Engineering: An International Journal.* https://doi.org/10.5121/cseij.2016.6101.

Sethi, A., Hemanth, S., Kumar, K., N, B. R., & Krishnan, R (2012). SignPro-an application suite for deaf and dumb. *International Journal of Computer Science Engineering and Technology, 2*(5), 1203–1206.

Shaik, K. B., Ganesan, P., Kalist, V., Sathish, B. S., & Jenitha, J. M. M. (2015). Comparative study of skin color detection and segmentation in HSV and YCbCr color space. In S. A. K. Lobiyal D.K. (Ed.). *Procedia Computer Science, 57*, 41–48 Elsevier. https://doi.org/10.1016/j.procs.2015.07.362.

Shanta, S. S., Anwar, S. T., & Kabir, M. R. (2018). Bangla sign language detection using SIFT and CNN. In *Proceedings of the 9th international conference on computing, communication and networking technologies, ICCCNT 2018.* https://doi.org/10.1109/ICCCNT.2018.8493915.

Sharma, A, Badal, T., Gupta, A., Gupta, A., & Anand, A (2021). An intelligent sign communication machine for people impaired with hearing and speaking abilities. *Communications in Computer and Information Science, 1367*, 75–86. https://doi.org/10.1007/978-981-16-0401-0_6.

Sharma, A., Mittal, A., Singh, S., & Awatramani, V (2020). Hand gesture recognition using image processing and feature extraction techniques. *Procedia Computer Science, 173*(2019), 181–190. https://doi.org/10.1016/j.procs.2020.06.022.

Sharma, S., Gupta, R., & Kumar, A. (2020). Trbaggboost: An ensemble-based transfer learning method applied to Indian sign language recognition. *Journal of Ambient Intelligence and Humanized Computing, 2018.* https://doi.org/10.1007/s12652-020-01979-z.

Shleeg, A. A., & Ellabib, I. M. (2013). Comparison of mamdani and sugeno fuzzy interference systems for the breast cancer risk. *International Journal of Computer, Electrical, Automation, Control and Information Engineering.*

Shohieb, S. M., Elminir, H. K., & Riad, A. M. (2015). Signs world atlas; a benchmark Arabic sign language database. *Journal of King Saud University - Computer and Information Sciences, 27*(1), 68–76. https://doi.org/10.1016/j.jksuci.2014.03.011.

Shrivakshan, G. T., & Chandrasekar, C. (2012). A comparison of various edge detection techniques used in image processing. *International Journal of Computer Science Issues.*

Shukla, P., Garg, A., Sharma, K., & Mittal, A. (2016). A DTW and Fourier descriptor based approach for Indian sign language recognition. In *Proceedings of the 3rd international conference on image information processing, ICIIP 2015* (pp. 113–118). https://doi.org/10.1109/ICIIP.2015.7414750.

Simon, C., Meessen, J., Tzovaras, D., & De Vleeschouwer, C (2007). Using decision trees for knowledge-assisted topologically structured data analysis. In *Proceedings of the 8th international workshop on image analysis for multimedia interactive services, WIAMIS 2007* (pp. 6–9). https://doi.org/10.1109/WIAMIS.2007.84.

Simske, S. (2019). Introduction, overview, and applications. *Meta-analytics*. https://doi.org/10.1016/b978-0-12-814623-1.00001-0.

Singh, P., & Shree, R. (2016). A comparative study to noise models and image restoration techniques. *International Journal of Computer Applications, 149*(1), 18–27. https://doi.org/10.5120/ijca2016911336.

Singh, S., Gupta, A. K., & Singh, T. (2019). Sign language recognition using hybrid neural networks. *International Journal of Innovative Technology and Exploring Engineering, 9*(2), 1092–1098. https://doi.org/10.35940/ijitee.I3349.129219.

Song, Q., & Chissom, B. S. (1993). Fuzzy time series and its models. *Fuzzy Sets and Systems, 54*(3), 269–277. https://doi.org/10.1016/0165-0114(93)90372-O.

Song, T., Zhao, H., Liu, Z., Liu, H., Hu, Y., & Sun, D (2021). Intelligent human hand gesture recognition by local–global fusing quality-aware features. *Future Generation Computer Systems, 115*, 298–303. https://doi.org/10.1016/j.future.2020.09.013.

Sonkusare, J. S., Chopade, N. B., Sor, R., & Tade, S. L. (2015). A review on hand gesture recognition system. In *Proceedings of the 1st international conference on computing, communication, control and automation, ICCUBEA: 2015* (pp. 790–794). https://doi.org/10.1109/ICCUBEA.2015.158.

Suharjito, Thiracitta, N., Gunawan, H., & Witjaksono, G (2019). The comparison of some hidden markov models for sign language recognition. In *Proceedings of the 1st Indonesian association for pattern recognition international conference, INAPR 2018* (pp. 6–10). https://doi.org/10.1109/INAPR.2018.8627031.

Suharjito, Anderson, R., Wiryana, F., Ariesta, M. C., & Kusuma, G. P (2017). Sign language recognition application systems for deaf-mute people: A review based on input-process-output. *Procedia Computer Science, 116*, 441–448. https://doi.org/10.1016/j.procs.2017.10.028.

Sujatha, P., & Sudha, K. K. (2015). Performance analysis of different edge detection techniques for image segmentation. *Indian Journal of Science and Technology*. https://doi.org/10.17485/ijst/2015/v8i14/72946.

Sun, C., Zhang, T., & Xu, C. (2015). Latent support vector machine modeling for sign language recognition with kinect. *ACM Transactions on Intelligent Systems and Technology, 6*(2). https://doi.org/10.1145/2629481.

Supančič, J. S., Rogez, G., Yang, Y., Shotton, J., & Ramanan, D (2018). Depth-based hand pose estimation: Methods, data, and challenges. *International Journal of Computer Vision, 126*(11), 1180–1198. https://doi.org/10.1007/s11263-018-1081-7.

Sykora, P., Kamencay, P., & Hudec, R. (2014). Comparison of SIFT and SURF methods for use on hand gesture recognition based on depth map. *AASRI Procedia, 9*(Csp), 19–24. https://doi.org/10.1016/j.aasri.2014.09.005.

Tabassum, M. R., Gias, A. U., Kamal, M. M., Islam, S., Muctadir, H. M., Ibrahim, M., Shakir, A. K., Imran, A., Islam, S., Rabbani, M. G., Khaled, S. M., Islam, M. S., & Begum, Z. (2010). Comparative study of statistical skin detection algorithms for sub-continental human images. *Information Technology Journal, 9*(4), 811–817. https://doi.org/10.3923/itj.2010.811.817.

Tan, Z. Y., Basah, S. N., Yazid, H., & Safar, M. J. A. (2021). Performance analysis of Otsu thresholding for sign language segmentation. *Multimedia Tools and Applications*. https://doi.org/10.1007/s11042-021-10688-4.

Tania, S., & Rowaida, R. (2016). A comparative study of various image filtering techniques for removing various noisy pixels in aerial image. *International Journal of Signal Processing, Image Processing and Pattern Recognition, 9*(3), 113–124. https://doi.org/10.14257/ijsip.2016.9.3.10.

Tariq, M., Iqbal, A., Zahid, A., Iqbal, Z., & Akhtar, J. (2012). Sign language localization: Learning to eliminate language dialects. In *Proceedings of the 15th international multitopic conference, INMIC 2012* (pp. 17–22). https://doi.org/10.1109/INMIC.2012.6511463.

Taskiran, M., Killioglu, M., & Kahraman, N. (2018). A real-time system for recognition of American sign language by using deep learning. In *Proceedings of the 41st international conference on telecommunications and signal processing, TSP 2018* (pp. 1–5). https://doi.org/10.1109/TSP.2018.8441304.

Tavari, N. V, & Deorankar, P. A. V. (2014). Indian sign language recognition based on histograms of oriented gradient. *International Journal of Computer Science and Information Technologies, 5*(3), 3657–3660.

Tharwat, A., Gaber, T., Hassanien, A. E., Shahin, M. K., & Refaat, B. (2015). Sift-based arabic sign language recognition system. *Advances in Intelligent Systems and Computing, 334*, 359–370. https://doi.org/10.1007/978-3-319-13572-430.

Thepade, S. D., Kulkarni, G., Narkhede, A., Kelvekar, P., & Tathe, S. (2013). Sign language recognition using color means of gradient slope magnitude edge images. In *Proceedings of the international conference on intelligent systems and signal processing, ISSP 2013* (pp. 216–220). https://doi.org/10.1109/ISSP.2013.6526905.

Tolentino, L. K. S., Serfa Juan, R. O., Thio-ac, A. C., Pamahoy, M. A. B., Forteza, J. R. R., & Garcia, X. J. O. (2019). Static sign language recognition using deep learning. *International Journal of Machine Learning and Computing, 9*(6), 821–827. https://doi.org/10.18178/ijmlc.2019.9.6.879.

Tornay, S., Razavi, M., & Magimai-Doss, M. (2020). Towards multilingual sign language recognition. In *Proceedings of the ICASSP, IEEE international conference on acoustics, speech and signal processing* (pp. 6309–6313). https://doi.org/10.1109/ICASSP40776.2020.9054631.

Torrione, P. A., Morton, K. D., Sakaguchi, R., & Collins, L. M. (2014). Histograms of oriented gradients for landmine detection in ground-penetrating radar data. *IEEE Transactions on Geoscience and Remote Sensing, 52*(3), 1539–1550. https://doi.org/10.1109/TGRS.2013.2252016.

Umamaheswari, D., & Karthikeyan, E. (2019). Comparative analysis of various filtering techniques in image processing. *International Journal of Scientific and Technology Research*.

Vandeginste, B. G. M., Massart, D. L., Buydens, L. M. C., De Jong, S., Lewi, P. J., & Smeyers-Verbeke, J. (1998). Chapter 44 - Artificial Neural Networks, *Handbook of chemometrics and qualimetrics: Part B* (20, pp. 649–699) In B. G. M. Vandeginste, D. L. Massart, L. M. C. Buydens, S. De Jong, P. J. Lewi, and J. Smeyers-Verbeke (Eds.). Elsevier. https://doi.org/10.1016/S0922-3487(98)80054-3.

Verma, N., & Dutta, M. (2017). Contrast enhancement techniques : A brief and concise review. *International Research Journal of Engineering and Technology(IRJET)*.

Verma, O. P., Hanmandlu, M., Susan, S., Kulkarni, M., & Jain, P. K. (2011). A simple single seeded region growing algorithm for color image segmentation using adaptive thresholding. In *Proceedings of the international conference on communication systems and network technologies, CSNT 2011*. https://doi.org/10.1109/CSNT.2011.107.

Verma, R., & Dev, A. (2009). Vision based hand gesture recognition using finite state machines and fuzzy logic. In *Proceedings of the international conference on ultra modern telecommunications and workshops*. https://doi.org/10.1109/ICUMT.2009.5345425.

Vincent, H., Tomoya, S., and Gentiane, V. (2019). Convolutional and recurrent neural network for human action recognition: Application on American sign language. BioRxiv. 10.1101/535492

Vogler, C., & Metaxas, D (1999). Parallel hidden Markov models for American sign language recognition. In *Proceedings of the IEEE International Conference on Computer Vision*. https://doi.org/10.1109/iccv.1999.791206.

Voyant, C., Notton, G., Kalogirou, S., Nivet, M.-L., Paoli, C., Motte, F., & Fouilloy, A. (2017). Machine learning methods for solar radiation forecasting: A review. *Renewable Energy, 105*, 569–582. https://doi.org/10.1016/j.renene.2016.12.095.

Wadhawan, A., & Kumar, P. (2020). Deep learning-based sign language recognition system for static signs. *Neural Computing and Applications, 32*(12), 7957–7968. https://doi.org/10.1007/s00521-019-04691-y.

Wadhawan, A., & Kumar, P. (2021). Sign language recognition systems: A decade systematic literature review. *Archives of Computational Methods in Engineering*. https://doi.org/10.1007/s11831-019-09384-2.

Wang, H., Chai, X., Hong, X., Zhao, G., & Chen, X (2016). Isolated sign language recognition with Grassmann covariance matrices. *ACM Transactions on Accessible Computing, 8*(4). https://doi.org/10.1145/2897735.

Wang, H., Chai, X., Zhou, Y., & Chen, X (2015). Fast sign language recognition benefited from low rank approximation. In *Proceedings of the 11th IEEE international conference and workshops on automatic face and gesture recognition, FG 2015*. https://doi.org/10.1109/FG.2015.7163092.

Wang, K. (2001). Computational intelligence in agile manufacturing engineering. *Agile manufacturing: The 21st century competitive strategy*. https://doi.org/10.1016/b978-008043567-1/50016-4.

Wang, S. J., Zhang, L. B., Zhang, D. C., Jia, C. C., Zhang, N., & Zhou, C. G. (2010). A sign language recognition based on tensor. In *Proceedings of the international conference on multimedia and information technology, MMIT 2010: 2* (pp. 192–195). https://doi.org/10.1109/MMIT.2010.21.

Wang, Y., Yang, M. Z., & Luo, Y. (2015). Improved SURF algorithm based on proportional distance and angle for static sign language recognition. In *Proceedings of the network security and communication engineering, international conference on network security and communication engineering, NSCE 2014* (pp. 265–270). CRC Press/Balkema. https://doi.org/10.1201/b18660-59.

Wilcox, S., & Occhino, C. (2016). Historical change in Signed Languages. *Oxford handbooks* Online.

Wilson, A. D., & Bobick, A. F. (1999). Parametric hidden Markov models for gesture recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. https://doi.org/10.1109/34.790429.

Wong, S. F., & Cipolla, R. (2005). Real-time adaptive hand motion recognition using a sparse Bayesian classifier. *Lecture Notes in Computer Science* (pp. 170–179) (Including subseries lecture notes in artificial intelligence and lecture notes in bioinformatics), 3766 LNCS. https://doi.org/10.1007/11573425_17.

Xanthopoulos, P., & Razzaghi, T. (2014). A weighted support vector machine method for control chart pattern recognition. *Computers and Industrial Engineering, 70*, 134–149. https://doi.org/10.1016/j.cie.2014.01.014.

Xu, B., Huang, S., & Ye, Z (2021). Application of tensor train decomposition in S2VT model for sign language recognition. *IEEE Access, 9*. https://doi.org/10.1109/ACCESS.2021.3059660.

Xu, W., Li, Q., Feng, H., Xu, Z., & Chen, Y (2013). A novel star image thresholding method for effective segmentation and centroid statistics. *Optik, 124*(20), 4673–4677. https://doi.org/10.1016/j.ijleo.2013.01.067.

Yang, Q., & Peng, J. Y. (2014). Chinese sign language recognition method based on depth image information and SURF-BoW. *Moshi Shibie Yu Rengong Zhineng/Pattern Recognition and Artificial Intelligence, 27*(8), 741–749.

Yang, W., Tao, J., & Ye, Z (2016). Continuous sign language recognition using level building based on fast hidden Markov model. *Pattern Recognition Letters, 78*, 28–35. https://doi.org/10.1016/j.patrec.2016.03.030.

YanuTara, R., Insap Santosa, P., & Bharata Adji, T. (2012). Sign language recognition in robot teleoperation using centroid distance fourier descriptors. *International Journal of Computer Applications, 48*(2), 8–12. https://doi.org/10.5120/7318-0100.

Yao, Y., & Doretto, G. (2010). Boosting for transfer learning with multiple sources. In *Proceedings of the IEEE computer society conference on computer vision and pattern recognition* (pp. 1855–1862). https://doi.org/10.1109/CVPR.2010.5539857.

Yasir, F., Prasad, P. W. C. C., Alsadoon, A., & Elchouemi, A. (2016). SIFT based approach on Bangla sign language recognition. In *Proceedings of the IEEE 8th international workshop on computational intelligence and applications, IWCIA 2015* (pp. 35–39). https://doi.org/10.1109/IWCIA.2015.7449458.

Yasir, R., & Khan, R. A. (2014). Two-handed hand gesture recognition for Bangla sign language using LDA and ANN. In *Proceedings of the SKIMA, 8th interna-*

tional conference on software, knowledge, information management and applications. https://doi.org/10.1109/SKIMA.2014.7083527.

Yogamangalam, R., & Karthikeyan, B. (2013). Segmentation techniques comparison in image processing. *International Journal of Engineering and Technology*.

Yu, C., Wang, X., Huang, H., Shen, J., & Wu, K. (2010). Vision-based hand gesture recognition using combinational features. In *Proceedings of the 6th international conference on intelligent information hiding and multimedia signal processing, IIHMSP 2010* (pp. 543–546). https://doi.org/10.1109/IIHMSP.2010.138.

Yuan, S., Wang, Y., Wang, X., Deng, H., Sun, S., Wang, H., Huang, P., & Li, G. (2020). Chinese sign language alphabet recognition based on random forest algorithm. In *Proceedings of the IEEE international workshop on metrology for industry 4.0 and IoT, MetroInd 4.0 and IoT 2020* (pp. 340–343). https://doi.org/10.1109/MetroInd4.0IoT48571.2020.9138285.

Yusnita, L., Rosalina, R., Roestam, R., & Wahyu, R. B. (2017). Implementation of real-time static hand gesture recognition using artificial neural network. *CommIT (Communication and Information Technology) Journal*. https://doi.org/10.21512/commit.v11i2.2282.

Zahedi, M., Dreuw, P., Rybach, D., Deselaers, T., & Ney, H. (2006). Geometric features for improving continuous appearance-based sign language recognition. In *Proceedings of the BMVC, British machine vision conference* (pp. 1019–1028). https://doi.org/10.5244/c.20.104.

Zaki, M. M., & Shaheen, S. I. (2011). Sign language recognition using a combination of new vision based features. *Pattern Recognition Letters, 32*(4), 572–577. https://doi.org/10.1016/j.patrec.2010.11.013.

Zapata, M.M., Madrenas, J., Zapata, M.M., and Alvarez, J. (2019). Advances in artificial intelligence, software and systems engineering (Vol. 787). https://doi.org/10.1007/978-3-319-94229-2.

Zarit, B. D., Super, B. J., & Quek, F. K. H. (1999). Comparison of five color models in skin pixel classification. In *Proceedings of the international workshop on recognition, analysis, and tracking of faces and gestures in real-time systems, RATFG-RTS*. https://doi.org/10.1109/RATFG.1999.799224.

Zhang, H., Wang, Y., & Deng, C. (2011). Application of gesture recognition based on simulated annealing BP neural network. In *Proceedings of the international conference on electronic and mechanical engineering and information technology, EMEIT 2011*. https://doi.org/10.1109/EMEIT.2011.6022891.

Zhang, L. G., Chen, Y., Fang, G., Chen, X., & Gao, W. (2004). A vision-based sign language recognition system using tied-mixture density HMM. In *Proceedings of the ICMI'04, sixth international conference on multimodal interfaces*. https://doi.org/10.1145/1027933.1027967.

Zhang, L. G, Chen, X., Wang, C., Chen, Y., & Gao, W. (2005). Recognition of sign language subwords based on boosted hidden Markov models. In *Proceedings of the seventh international conference on multimodal interfaces, ICMI'05* (pp. 282–287). https://doi.org/10.1145/1088463.1088511.

Zhang, L. G., Chen, X., Wang, C., & Gao, W. (2005). Hierarchical voting classification scheme for improving visual sign language recognition. In *Proceedings of the 13th ACM international conference on multimedia, MM 2005* (pp. 339–342). https://doi.org/10.1145/1101149.1101220.

Zhao, S., Zhang, S., Liu, J., Wang, H., Zhu, J., Li, D., & Zhao, R. (2021). Application of machine learning in intelligent fish aquaculture: A review. *Aquaculture, 540*, Article 736724. https://doi.org/10.1016/j.aquaculture.2021.736724.

Zhao, W., Zhang, J., Li, P., & Li, Y. (2010). Study of image segmentation algorithm based on textural features and neural network. In *Proceedings of the international conference on intelligent computing and cognitive informatics, ICICCI*. https://doi.org/10.1109/ICICCI.2010.53.

Zheng, X., Lei, Q., Yao, R., Gong, Y., & Yin, Q (2018). Image segmentation based on adaptive K-means algorithm. *Eurasip Journal on Image and Video Processing*. https://doi.org/10.1186/s13640-018-0309-3.