International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Ingénierie des Systèmes d'Information
Vol. 30, No. 3, March, 2025, pp. 703-711
Journal homepage: http://iieta.org/journals/isi

Associative Memory for Recognition and Translating American Sign Language

Hasanain Ali Al Essa[1*], Wial Hanon[1], Tahseen A. Wotaifi[1], Ali K. Abdul Raheem[2]

[1] College of Information Technology, University of Babylon, Hillah 51001, Iraq
[2] Department of Information Technology, College of Science, University of Warith Al-Anbiyaa, Kerbala 56001, Iraq

Corresponding Author Email: Hasanain@uobabylon.edu.iq

https://doi.org/10.18280/isi.300314

Received: 3 November 2024
Revised: 23 February 2025
Accepted: 14 March 2025
Available online: 31 March 2025

Keywords:
American sign language, multi-connect associative memory (MCAM), neural network, real-time hand gesture recognition

ABSTRACT

It is essential to recognize extremely complicated hand movements with comparable shapes. Because the human community depends on gestures to convey their goals, it is imperative that a system be able to effectively recognize these gestures. Otherwise, it could do harm to the gesture recognition community. The purpose of the suggested approach is to highlight the most important phases in the hand gesture identification procedure which is the process of identifying and recognizing hand motions utilizing a multi-connect associative memory (MCAM) neural network and hand landmark points for hand detection. The problem with the similarity between the signs is because of the strong correlation between the movements of the fingers. In addition to the non-high accuracy between complex and very similar signs (e.g., A, S, and E) and the problem of response time in hand gesture recognition in real-time, thus, using the MCAM neural network improved efficiency in dealing with the correlation between similar patterns by taking similar vectors for each hand gesture pattern only once. The proposed system demonstrated promising outcomes in real-time with an accuracy for ASL is 96.28 and 99.77 for numbers. As well as the system's work in an uncontrolled environment, in addition to being an applicable system by converting the sign into its meaning as words and sentences, not just letters.

1. INTRODUCTION

Hand signs, face emotions, and other body language are the basis of sign languages [1, 2]. Learning sign language is beneficial for people of all ages and backgrounds, not just those who are deaf or hard of hearing, who use it as their primary way of communication. Others who may benefit from learning sign language include those with developmental disorders such as Autism, Apraxia of Speech, Cerebral Palsy, and Down Syndrome.

There is no universally accepted sign language. There are as many diverse sign languages as there are spoken languages because each has evolved independently as different groups of people have interacted with one another. The number of sign languages in use now is estimated to be between 138 and 300 [3]. Image processing has been used in many vital applications such as non-destructive detection of defects [4] and medical treatments, in addition to taking up a large space in computer vision [5].

It's interesting to note that even when a country and another share a spoken language, they don't always use the same sign language. American Sign Language (ASL), British Sign Language (BSL), and Australian Sign Language are all variations of the English language [6].

Most beginning sign language students begin their studies by learning the alphabet. 'Fingerspelling' refers to the practice of using one's hands to stand in for each letter of the alphabet. Used by signers to manually spell out names of persons, places, and things for which there is no standard sign [7], it is an essential tool. Visual cues such as hand gestures, face and body alignment, lip movement, and facial expressions and emotions form the basis of sign language. Arabic Sign Language (ArSL), American Sign Language (ASL), and Indian Sign Language are only a few examples of the various sign languages that exist, each with its own unique features because they are based on different spoken languages and their regional dialects. Isolated sign language refers to communication that consists of individual hand gestures for individual words, while continuous sign language refers to communication that consists of sequences of gestures that create sentences. Most sign language categorization systems rely heavily on the identification of hand gestures [8].

Hand gesture detection has recently increased by research on automated [4] due to several factors, including the growing population of the deaf, the popularity of gesture-controlled smart gadgets, games, and assistive technology, and the need for communication aids for these populations. Accurate hand gesture recognition can aid in the development of a powerful sign language recognition and classification system, which can enable both hearing-and speech-impaired people to communicate more effectively. Conventional algorithms find it difficult to uncover robust features when temporal misalignment is present, machine learning methods that don't use manually created features can't tell the difference between crucial and irrelevant parts of each frame [9].

The research is arranged according to the following structure: the first section deals with the introduction, the second section presents the related works with the methods used, the third section deals with the proposed method and finally the results section is found in the fourth section followed by the conclusion and reference.

2. RELATED WORK

In latest years, significant advancements have been made in the field of associative memory, particularly in its application to recognition and translation tasks. This paper explores the implementation of associative memory models to improve the recognition and translation of ASL. Existing research in this domain has predominantly focused on machine learning algorithms for gesture recognition and natural language processing for translation. However, integrating associative memory offers a novel approach that enhances the system's ability to learn and recall complex patterns inherent in ASL. Previous studies have demonstrated the potential of associative memory in various cognitive tasks, yet its application to sign language remains underexplored. This research aims to bridge that gap by leveraging associative memory to achieve more accurate and efficient ASL recognition and translation, thereby contributing to the broader field of accessible communication technologies.

This research presents the design of the ArSL Recognition Model. There were four phases of the recognition process involved in turning the alphabet into letters: There are three stages: an initial image loading phase in which ArSL letter images are loaded for use in subsequent training and testing phases a Pre-processing phase, wherein essential features for completing the recognition are extracted via image processing techniques such as Normalization, Resizing, Image Augmentation, and Filter; CNN and other methods of deep learning are used during the training phase, whereas the testing phase shows how well the model performs in practice [10].

An efficient sign language identification tool for e-learning with a novel angle and line feature set is proposed in this research. This feature set boosts machine learning algorithm performance efficiently. These traits are used for real-time hand motion recognition. Each frame of the incoming video was processed using MediaPipe and openCV to generate hand landmarks for the feature set. The system was evaluated on two popular sign language datasets, ASL-alphabet and ISL-HS. Various machine learning classifiers, such as random forest, decision tree, and naïve Bayesian, were tested for hand gesture classification utilizing this unique feature set. The suggested system's basis classifier is the random forest classifier because it performed better. It had 96.7% accuracy with ISL-HS and 93.7% with ASL-alphabet datasets utilizing extracted features [11]. In this work [12], combining recognition and tracking with image processing solves motion recognition. Many target detection and recognition image processing algorithms exist. CNN-based approaches for user-free hand gesture identification are presented in this research. Use synthetic methods to increase recognition accuracy. The CNN automatically retrieves high-level features from the raw image, and the SVM classifies them. This work used a CNN to automatically extract attributes from raw EMG images, unlike conventional feature extractors. The SVM classifier then identifies hand motions. CNN alone is less accurate than the proposed technique.

The modified CNN technique was used to construct a real-time Kurdish sign recognition model in this study. This study focuses on Kurdish alphabet recognition. Training the model with various activation functions over numerous iterations allowed it to predict on the KuSL2023 dataset. The dataset contains 71,400 images from two sources representing Kurdish sign languages and alphabets. This study introduces a Kurdish Sign Language (KuSL) classification model. In a photograph with a complicated backdrop, including lighting, ambient, and image color variations of varied intensities. The suggested KuSL detection method improves on earlier research by using an actual public dataset, real-time classification, and personal independence while maintaining high classification accuracy. With an average classification and prediction model training accuracy of 99.05 [13].

This research introduces a new deep learning neural network structure to identify sign language static hand gestures. The suggested structure uses the CNN and classical non-intelligent feature extraction. The proposed structure extracts effective features and classifies hand gestures from the hand gesture image after preprocessing and removing the background using three feature extraction streams. These three streams, which independently extract features, include CNN, Gabor filter, and ORB feature descriptor, three popular hand gesture categorization algorithms. Combining these efficient methods yields excellent hand gesture classification accuracy and makes the suggested structure more resistant to rotation and ambiguity. The transfer learning technique shows that the proposed structure can be pre-trained for any image database. It is applied to Massey, ASL Alphabet, and ASL, which have 2520, 87,000, and 23,400 hand gesture images, respectively. The proposed structure had a mean accuracy of 99.92%, 99.8%, and 99.80% for the Massey test set of 758 photos, ASL with 7020 images, and ASL Alphabet with 26,100 images [14].

3. THE PROPOSED SYSTEM

The suggested system's primary goal is to create a RTHGR model for real-time hand gesture recognition. To this end, the following objectives are listed:

1). Making use of the Real-Time Hand Gesture Recognition Translate (RTHGRT) model and associative memory approaches. Enhancing the dependability of the suggested system by utilizing Multi-Connect Architecture (MCAM) associative memory approaches to classify complicated and related movements.

2). Making good use of landmark points to identify hand gestures based on hand shapes in a variety of conditions, such as illumination fluctuation, clutter, and scale.

3). Enhancing the process of translating signs into words that accurately convey meaning in order to address the communication gap experienced by the deaf and dumb.

Using landmark points is important since it provides the hand's shape or skeletal features, which opens up the potential of acquiring the gesture structure graph. Its effectiveness in detecting the hand has been demonstrated. In order to separate the hand from the picture as a whole, the suggested solution used Google's landmark point methodology for hand detection. Convolution on the entire image using (mp.solutions.hands) to determine the palm based on skin color.

Illustrate the triangle on the palm of the hand, then recording the indices of its three vertices. The landmark points methodology was often used to extract features and take advantage of the coordinates of the landmark points in that. As for the proposed system, used only the landmark points and took only the shape of the skeleton of the hand resulting in the interdependence of the 21 points with each other, and then it is displayed thickly. Can determine the possibility of changing the size of the circles and lines that connect one joint to another. The thickness of the line and circle for each identifier is set in white, as well as the thickness is set to 12 pixels, and the radius of the circle is set to 3 pixels. These standards were taken because they clearly gave shape to the hand structure. After displaying the skeleton of the fingers of the hand within a specified width and radius of a circle, a drawing of the template is determined on the borders of the skeleton, and the hand is then cropped out of the overall image.

It is important to note that the middle, or zero point, of the hand was used as the basis for drawing the window since the zero point in the palm is thought to be the location from which the window's or templet's borders begin and it needs to include every point of the landmark. Making a hand template in order to facilitate cropping and provide for freedom in how the structure is taken. Figure 1 shows the proposed method.

The concept of associative memory was used as a new direction that researchers had not addressed before within the recognition of hand gestures or sign language, where MCAM was used in the stage of recognizing sign language. The use of MCAM gives good results with an efficient proposed system.

Making the system applicable and technologically important because it is considered an important tool to make communication between deaf people and ordinary people more flexible is an important part of the proposed system. In the stage of displaying the gestures flowing from the video in the form of text, the processing includes taking into account the number of classes in the look-up table and the possibility of giving priority to displaying the results of sign language recognition for the frame that has the highest confidence among several sequential frames.

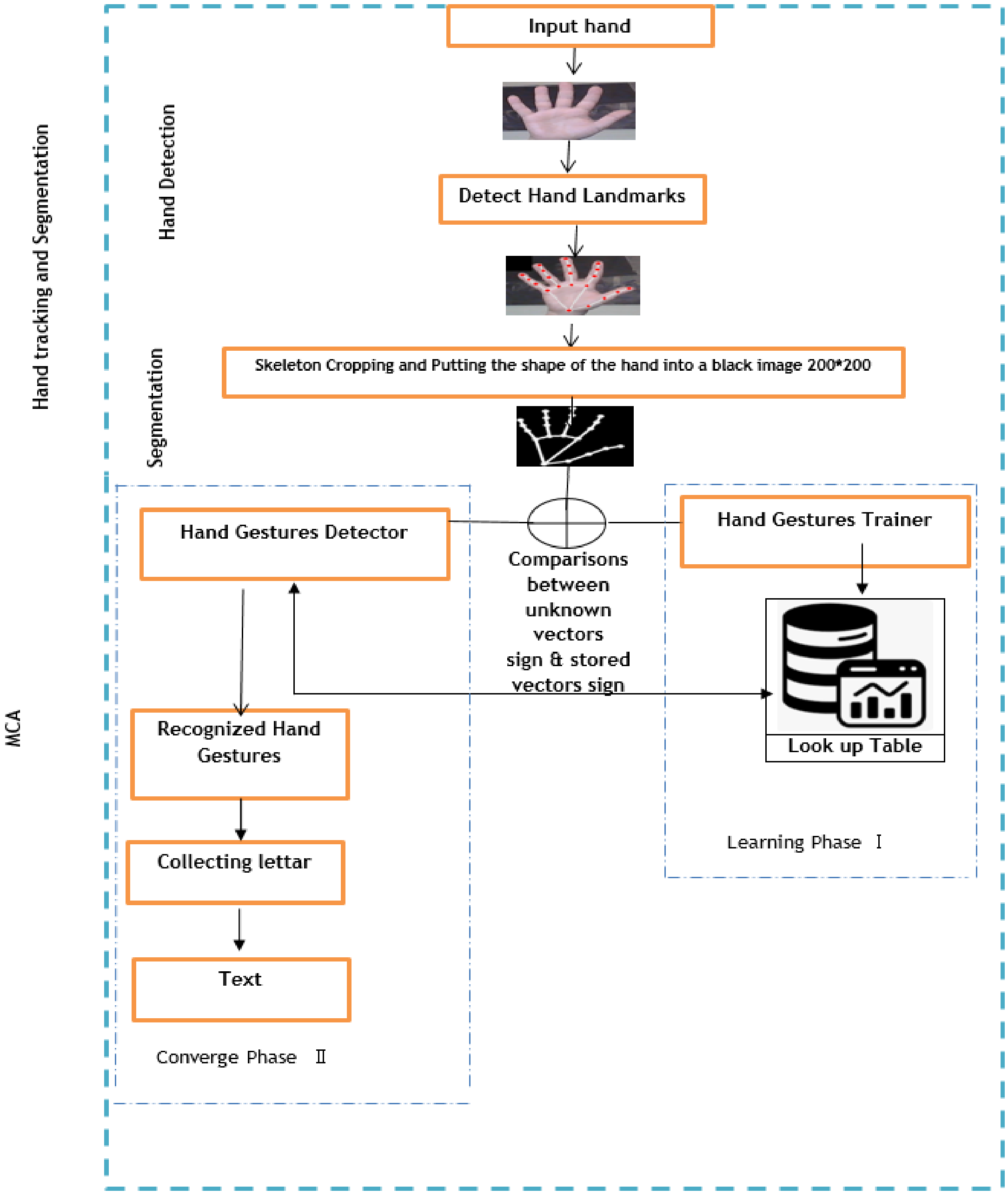


Figure 1. The RTHGRT system

This approach has demonstrated significant efficacy in identifying the hand and delineating a square or rectangle in various manners, contingent upon whether the hand is open or closed. Place the white skeletal hand against a black backgrounds image measuring 200 by 200 pixels. Algorithm 1 elucidates hand detection and segmentation.

Landmark point selection during segmentation (focusing on hand structure) yielded strong results, as binary images simplify and accelerate feature extraction. A quality assessment method will be added to evaluate accuracy (e.g., error metrics against ground truth) and consistency under varying conditions, ensuring robust input to the MCAM network.

Algorithm 1: Hand Detection and Segmentation

```
Input: Stream of Frames
Output: Binary Image (Shape of Hand)
   Step 1: Read the frame - jpg image
   image=cv2.imread('path_to_your_image.jpg')
   Step 2: Create a 200x200 pixel black image
   black_image=np.zeros((200, 200, 3), dtype=np.uint8)
   Step 3: Convert BGR to RGB
   Step 4: Detect hands in the image
           mp_hands=mp.solutions.hands
           mp_drawing=mp.solutions.drawing_utils
   Step 5: Depict the triangle on the palm and record the indices of the 3 vertices
   Step 6: Draw the landmark points
   mp_drawing.draw_landmarks(image, hand_landmarks, mp_hands.HAND_CONNECTIONS)
   Step 7: Set drawing parameters
          color=(255, 255, 255)
          thickness=12
          circle_radius=3
   Step 8: Drawing the landmark points with specified parameters
           for idx, lm in enumerate(hand_landmarks.landmark):
           h, w, c=image.shape
           cx, cy=int(lm.x * w), int(lm.y * h)
            cv2.circle(image, (cx, cy), circle_radius, color, cv2.FILLED)
   Step 9: Draw the template of hand
           start_point=(int(x_min)-15, int(y_min)-15)
           end_point=(int(x_max)+15, int(y_max)+15)
           cv2.rectangle(image, start_point, end_point, color, thickness)
   Step 10: Crop the image and set it to the black image
   Step 11: End
```

The application of the above algorithm gave promising results in obtaining the binary image that will be fed to the network. Figure 2 illustrates the results of hand segmentation.

The MCAM network uses 3 neurons (the smallest odd number for faster processing) and 4 weights due to orthogonality principles. Weights are derived by multiplying the vector by itself (diagonal zeroed), resulting in the first four weights mirroring the last four. Orthogonal weights (e.g., w0 with smd=1 and w7 with smd=-1) ensure efficient pattern recognition, as detailed in Figure 3.

The proposed system utilizes MCAM (Multiple Correspondence Analysis Method) to address the issue of capturing both similar and dissimilar patterns. By constructing a compact fully connected network structure with only four weights, the system ensures that each pattern is stored only once. This process is elaborated in the second chapter and is applied to every video frame. Upon obtaining a binary image sign pattern (SP) of the hand in the preliminary phase, this image will be transformed into a vector. The image elements, represented by pixel values of either 0 or 255, will be encapsulated in a compressed vector format of three pixels. The resulting vector pairs comprise just three distinct items, and since the vector's length is 3, the probable values of the vector elements range from 0 to 7. As depicted in Figure 4.

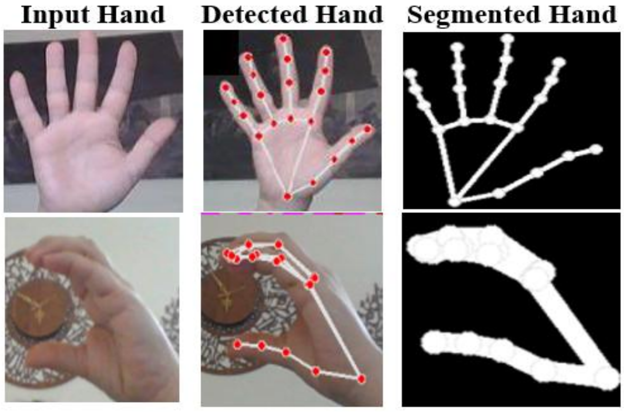
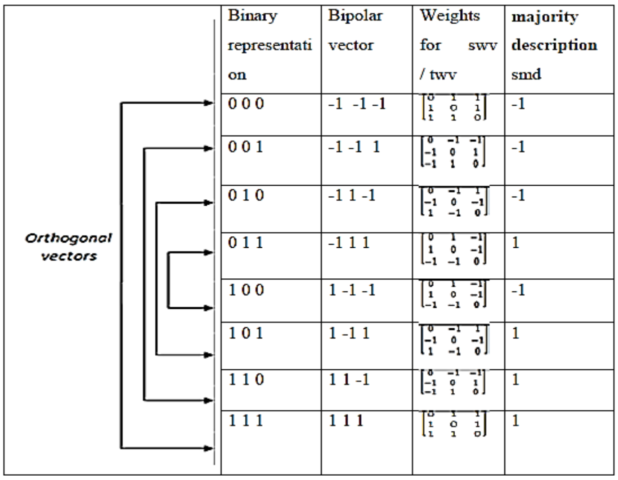


Figure 2. Hand detection and segmentation


Orthogonal vectors

Binary representation	Bipolar vector	Weights for swv / twv	majority description smd
0 0 0	-1 -1 -1	$\begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}$	-1
0 0 1	-1 -1 1	$\begin{bmatrix} 0 & -1 & -1 \\ -1 & 0 & 1 \\ -1 & 1 & 0 \end{bmatrix}$	-1
0 1 0	-1 1 -1	$\begin{bmatrix} 0 & -1 & 1 \\ -1 & 0 & -1 \\ 1 & -1 & 0 \end{bmatrix}$	-1
0 1 1	-1 1 1	$\begin{bmatrix} 0 & 1 & -1 \\ 1 & 0 & -1 \\ -1 & -1 & 0 \end{bmatrix}$	1
1 0 0	1 -1 -1	$\begin{bmatrix} 0 & 1 & -1 \\ 1 & 0 & -1 \\ -1 & -1 & 0 \end{bmatrix}$	-1
1 0 1	1 -1 1	$\begin{bmatrix} 0 & -1 & 1 \\ -1 & 0 & -1 \\ 1 & -1 & 0 \end{bmatrix}$	1
1 1 0	1 1 -1	$\begin{bmatrix} 0 & -1 & -1 \\ -1 & 0 & 1 \\ -1 & 1 & 0 \end{bmatrix}$	1
1 1 1	1 1 1	$\begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}$	1

Figure 3. All the possible vectors with their weight matrices (swv) and majority description (smd)

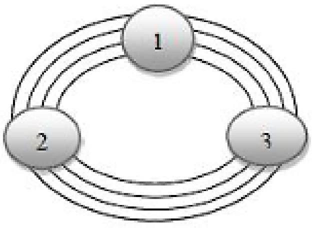
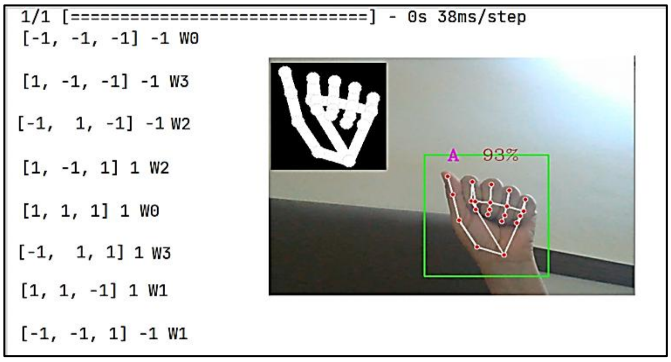


Figure 4. Network of 3 vectors and 4 weights

The RTHGRT is implemented on a limited neurons to provide a compact architecture, which enhances the speed of HGR computations, recognition, and real-time recognition. The network is constrained to only three neurons, however there are a multitude of connections between these three neurons. As a result, limiting the number of active connections to four can effectively handle the smallest network size and accelerate RTHGRT.

The weights in the MCAM neural network are initialized by multiplying the bipolar vector by itself while zeroing out the diagonal. This process results in 8 weight matrices derived from 8 vectors, where the first four matrices are symmetrical with the last four. This initialization leverages the concept of orthogonality, which ensures minimal interference between weight matrices and enhances the network's ability to generalize. By promoting orthogonal weight structures, this method significantly improves network convergence and recognition accuracy, as it minimizes redundancy and supports efficient learning during training.

The images from the previous phase are stored in a one-dimensional array only if there are no duplicates of these three elements in the new one-dimensional array. In this instance, a 200x200 image will be compressed into a one-dimensional array including 24 elements or fewer, but not exceeding that number. The possible vectors that may arise are: [1 1 1], [-1 1 1], [1 -1 1], [1 1- 1], [-1 -1 1], [-1 1 -1], [1 -1 -1], [-1 -1 -1]. All occurrences of the value 0 in the segmented picture were replaced with -1, whereas all occurrences of the value 255 were replaced with 1. The result of the training phase will be a table that contains the computed information. The learning phase is a crucial aspect that greatly influences the overall efficacy of the network. The outcomes of the learning phase will influence the output. Figure 5 depicts the convergence phase. Although the alteration was made during the learning phase, it is crucial to fully adjust for the convergence phase to ensure the overall effectiveness of the RTHGRT. The RTHGRT proposed a solution to the problem of hand gesture recognition that is characterized by its simplicity. Unlike deep learning applications, it only requires one layer and a small number of training data sets. Additionally, by expanding its application to include phrases and sentences instead of just numbers and letters, the suggested system has the potential to improve sign language recognition.



```
1/1 [==============================] - 0s 38ms/step
[-1, -1, -1] -1 W0

[1, -1, -1] -1 W3

[-1,  1, -1] -1 W2

[1, -1, 1] 1 W2

[1, 1, 1] 1 W0

[-1,  1, 1] 1 W3

[1, 1, -1] 1 W1

[-1, -1, 1] -1 W1
```

Figure 5. Compute the 8 possible vectors, smd and weight for the pattern

In order for the proposed RTHGRT by using MCAM associative memory to work properly, the images are translated to binary format, with pixel values represented as having a smaller dataset to process during the training phase, either 1 or -1. All four weight matrices (w_0,w_1,w_2,w_3) are initialized, and the unknown gesture is split into k vectors of three elements each. the modification process incorporates the growing importance of the energy function in the convergence process. Figure 3 illustrates the process of determining both based on the principle of orthogonality between the weights and vectors extracted from the binary image.

In the convergence phase, the same steps followed in the training phase will be repeated, in addition, the energy function matrix (efm) is used to know the extent of similarity between the entered pattern and what is stored [14]. After knowing the closest pattern in the table to the currently tested pattern, calculate the convergence percentage using the following equation.

$$\text{Convergence Rate}= \frac{efm}{The\ Length\ of\ the\ Sign\ pattern} * 100 \tag{1}$$

When (efm) is sum up all k-vectors for test hand gesture pattern. The energy function is designed to ensure system convergence by minimizing energy states, which stabilizes pattern recognition. It handles noise by penalizing large deviations while allowing minor variations. The threshold is selected through empirical analysis and optimization, using training data to balance sensitivity and specificity. Cross-validation ensures the threshold generalizes well, directly impacting recognition accuracy.

If the result of the sum of the (efm) is -24, then the entered pattern of the gesture or sign is 100% identical with stored sign. If the result of the summation is 8, then the entered pattern of the gesture or signal is not completely identical with stored sign. Figure 3 shows an example of calculating the value that will be stored in the lookup table that output of the learning process.

Making the system adaptable and technologically significant since it is regarded as a key instrument for improving the flexibility of communication between deaf individuals and ordinary people is an important component of the proposed system. The processing includes taking into account the number of classes in the look-up table and the possibility of giving priority to displaying the results of sign language recognition for the frame with the highest confidence among several sequential frames at the stage of displaying the gestures flowing from the video in the form of text.

The translation was applied to ASL in Algorithm 2, which was recognized using the associative memory network as a new direction. In addition, the resulting sentence is more adaptable via the addition of two gestures (delete and space). ASL was learned with the right hand, whereas the digits 1 through 5 and a few other motions (delete and space) were learned with the left. Determining the result of the text and showing the letters as a word depends on the speed and number of frames. Set a time period for the frame rate and choose the frame with the highest confidence to be the result shown on the text box.

Both the right and left hands were used in this system, but the number of classes for the right hand is 26 specified to recognize 26 letters A to Z , in contrast, the number of classes for the left hand is 7, so the speed of the processing of the frames is given to the left hand much less than what is the case for the right hand, Because the number of classes for right hand is more and you need time to search more in the lookup table, and therefore the period granted to the number of frames is greater than the period granted to process the gestures of the left hand.

The system adapts to variations in gesture speed by dynamically selecting the frame with the highest confidence for recognition, rather than relying on a fixed number of frames. This ensures that the recognition process is flexible and responsive to differences in gesture execution speed. For instance, the recognition decision for one letter may occur at the 5th frame, while for another letter, it might occur at the 25th frame, depending on the gesture's complexity and speed. This adaptive mechanism allows the system to handle both fast and slow gestures effectively, thereby maintaining high recognition accuracy across practical scenarios with varying gesture speeds.

```
Algorithm 2: Hand Gestures translator
Input: Alphabet corresponding the input sign
Output: Text
    Step 1: Coose No. of frames (period). For both hand.
    Step 2: Take the alphabet of frames with the highest convergence rate
    Step 3: Pint the result (letter or alphabet) of sum the frame with the highest convergence rate in the text box
    Step 4: Collecting the letters resulting from the recognition stage as a word or a sentence from several words
    Step 5: Return text
    Step 6: End
```

4. RESULTS AND DISCUSSIONS

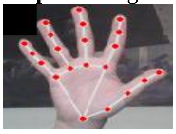
The research utilizes the ASL data, consisting of an average of 150 photos per sign during the training phase and 50 images per sign during the testing phase. The confusion matrix is extensively employed in the evaluation of systems due to its influence on accuracy, percentage, and other evaluation criteria. The accuracy criterion was employed to assess the sign language recognition of all the datasets utilized in the RTHGRT system. Based on the confusion matrix, the proposed method achieved a high level of accuracy, specifically 96.28%, in real-time when utilizing ASL.

In this section, an example is taken and the steps of the proposed method are applied to it in sequence from detecting the hand to recognizing the meaning of the input sign:

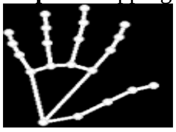
Case study steps

Step 1: Input hand / Unknown sign

Step 2: Using landmark point to detect the hand



Step 3: Cropping the shape of hand



Step 4: Convert the pattern to the sign pattern vector SP

SP = [-1 -1 -1 -1 1 1 1 -1 -1-1 1 -1 -1 -1 1 1-1 1 1 1 -1 1 1 1]

Step 5: Divided the sign pattern vector SP for 8 vector that length 3

[-1, -1, -1][-1, 1, 1] [1, -1, -1][-1, 1, -1]
[-1, -1, 1][1, -1, 1] [1, 1, -1][1, 1, 1]

Step 6: Assign smd and tvw for all 8 vector's

Vector	Decimal	smd	tvw
[-1, -1, -1]	0	-1	W0
[-1, 1, 1]	3	1	W3
[1, -1, -1]	4	-1	W3
[-1, 1, -1]	2	-1	W2
[-1, -1, 1]	1	1	W1
[1, -1, 1]	5	1	W2
[1, 1, -1]	6	1	W1
[1, 1, 1]	7	1	W0

Step 7: Comparisons between test and storage vector in memory tvw, SVW

Vector	tvw	SVW	Ep1
[-1, -1, -1]	W0	W0	-3
[-1, 1, 1]	W3	W3	-3
[1, 1, -1]	W3	W3	-3
[-1, 1, -1]	W2	W2	-3
[-1, -1, 1]	W1	W1	-3
[1, -1, 1]	W2	W2	-3
[1, 1, -1]	W1	W1	-3
[1, 1, 1]	W0	W0	-3

Vector	tvw	SVW	Ep2
[-1, -1, -1]	W0	W0	-3
[-1, 1, 1]	W3	W2	1
[1, -1, -1]	W3	W1	1
[-1, 1, -1]	W2	W1	1
[-1, -1, 1]	W1	W0	1
[1, -1, 1]	W2	W3	1
[1, 1, -1]	W1	W2	1
[1, 1, 1]	W0	W3	1

Look up table in memory

Image ID	Vectors	SVM	SVW
5	[-1, -1, -1]	-1	W0
5	[-1, 1, 1]	1	W3
5	[1, -1, -1]	-1	W3
5	[-1, 1, -1]	-1	W2
5	[-1, -1, 1]	1	W1
5	[1, -1, 1]	1	W2
5	[1, 1, -1]	1	W1
5	[1, 1, 1]	1	W0
6	[-1, -1, -1]	-1	W0
6	[-1, 1, -1]	-1	W2
6	[-1, -1, 1]	1	W1
6	[1, 1, -1]	1	W1
6	[1, 1, 1]	1	W0
6	[1, -1, -1]	-1	W3
6	[-1, 1, -1]	-1	W2
6	[-1, 1, 1]	1	W3

Step 8: Summing the energy function for all 8 vectors containing a test sign is done with each weight sign vector stored in look up table

$$\sum \text{Ep1} = -24$$
$$\sum \text{Ep2} = 4$$

According to the comparisons made between the test vectors of the sign with stored vectors in the look up table. The first id (5) stored in look up table have minim summing the energy function for all 8 vectors containing a test sign.

Step 9: Calculate Confidence (Convergence Rate CR) Depending on The Energy Function using Eq. (1)

Step 10: Returns alphabet corresponding to the input sign: 5

The system recognizes signs within a maximum of one second by selecting the most reliable frame (highest confidence) and comparing it with 26 character patterns. Response times vary per character, averaging between 12 and 38 milliseconds, as shown in the comparison table.

RTHGRT efficiency comes from the utilization of orthogonality in the MCAM neural network, resulting in the utilization of only four weight matrices, which is its primary advantage. Thus, the comparison procedure is facilitated during the retrieval process. MCAM streamlined the conventional pre-processing methods, features extraction techniques, and addressed the challenge of resolving correlations among similar patterns.

The convergence phase of the RTHGRT network yields remarkable real-time results solely by assignment, without including time-consuming complex calculations or processes. The suggested approach for hand gesture identification is distinguished by several factors. Firstly, the network consists of only one layer. Secondly, the weights used in the network are limited to just four. Lastly, the number of images utilized in the system is rather modest compared to what is typically processed by a Convolutional Neural Network (CNN). The energy function was used to calculate the degree of confidence. Because the recognized pattern is not 100% identical with the saved sign patterns in the look-up table, a convergence rate of 70% to 90% or less shows that the convergence process is successful this is done without any complication due to the assignment process for only four weights per sign pattern. The procedure of recognizing four signs with a confidence rating for both is depicted in (Figure 6).

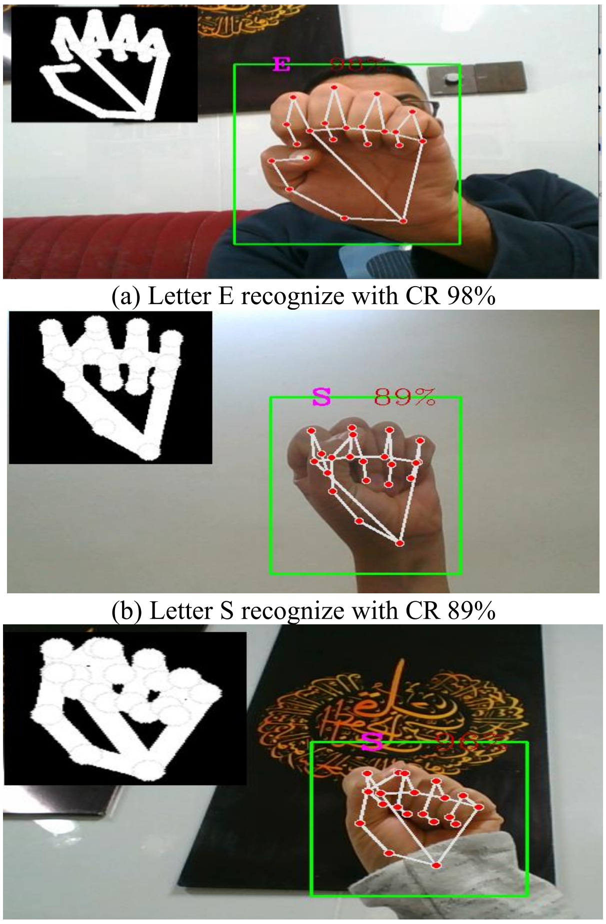
One of the challenges facing sign language recognition systems is that even an expert person cannot always repeat the same sign with the same structure exactly, especially for complex signs with finger movement. Therefore, the possibility of deleting the wrong letter (sign) displayed on the translation screen gave the desired flexibility. The accuracy results of the proposed system within a real-time perspective will be discussed in the next section.

Table 1 contains comparisons with studies [15-17], which were conducted on the same dataset. The primary difference lies in the number of images used in each study.

The system distinguishes similar gestures using the convergence rate from the first equation, which differentiates exact and similar outputs based on patterns, as illustrated in Figure 6. This strategy ensures accurate recognition of letter gestures with subtle differences.

The normal rate for video frames is 30 frames per second, so if the classes of signs are few, the number (period) of frames can be low. On the contrary, if the number of classes of signs is large, it needs time for bigger processors, so give a number (period) of frames that can be large. The possibility of choosing a frame at different speeds and within the actual time of the process of selecting the letter corresponding to a specific sign gives the translation process the necessary flexibility. It is worth noting that the translation process was applied to ASL, and the right and left hands were used.

The number of classes allocated for the right hand to recognize it is 26, while the number of classes that the left hand must recognize is only 10. The proposed system characterized is far from complicated, and the number of data sets used is much less than what researchers used in other ways, such as CNN. The proposed method of sign language recognition has been compared with some related research and the result of the comparison is presented in the Table 1.



(a) Letter E recognize with CR 98%

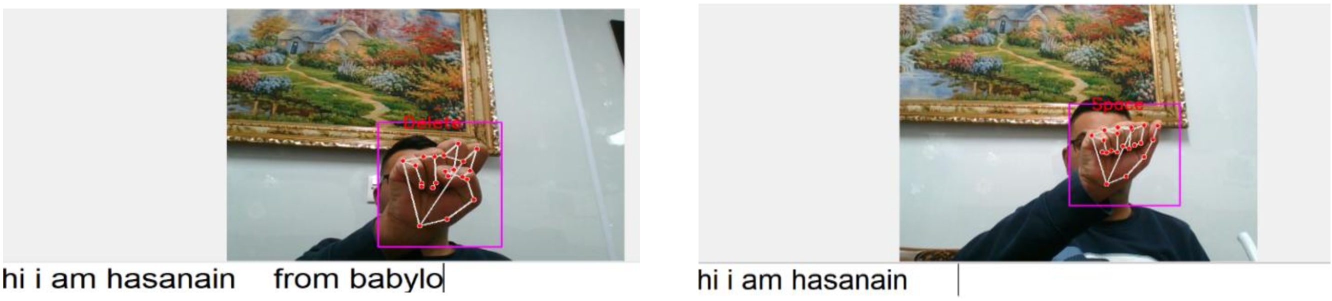
(b) Letter S recognize with CR 89%

(c) Letter S recognize with CR 96%

Figure 6. Recognize the signs with convergence rate

Table 1. Comparison of the proposed method with some previous works

Reference	Year	Approach	Accuracy	No of Sign	Data Set	Real Time	Exec-Ution Time
[16]	2018	Densely Connected Convolutional Neural Networks (DenseNet)	90.3	26	64790	√	0.12-0.14ms
[17]	2021	CNN and LSTM	85.6%	28	60000	√	N/L
[18]	2020	Deep CNN	94.34%	26	12500		N/L
[19]	2020	Neutrosophic and fuzzy c-means	91.0%	28	5400	√	
[20]	2024	Complex deep learning	89.7%	6	3700 video	N/L	N/L
The Proposed Method	**2024**	**MCAM**	**96.28%** **99.77%**	**26** **1-5**	**5200** **500**	√	**0.12-0.30s**



hi i am hasanain from babylo

hi i am hasanain

(a) Example of a special space command gesture

(b) Example of a delete command gesture

Figure 7. Create a complete sentence with the ability to delete the wrong letter

The proposed method enhances fault tolerance by allowing users to control the duration of letter printing and correct errors. It selects the frame with the highest confidence for sentence construction, uses a special space sign between words, and enables the removal of incorrectly printed letters, as illustrated in Figure 7. These features ensure robust error recovery and improve system reliability in real-world scenarios.

5. CONCLUSIONS

In recent years, Gestures are a form of non-verbal communication that helps deaf people to communicate with others. Therefore, there is an urgent need to develop applications capable of interpreting the meaning of the gesture by building an accurate and effective system. Gesture recognition within a computer vision approach is therefore an appropriate option.

The results of proposed method in real-time gave 96.28%, 99.77% accuracies for American and numbers sign language sequentially. In the segmentation stage, the Landmark Point gives accurate and effective results. It is very efficient against changing the illumination in addition to avoiding the clutter in the background. The environment in which the system operates is not specified, thus the proposed system will be able to adapt to any for any target environment in terms of lighting and background. Using multi-connected associative memory as a substitute for other methods in hand gesture recognition is a new direction, especially since associative memory does not work in real-time, so the use of MCAM was a good choice because it reduced the weights that are used in the training and convergence processes to half within the concept of orthogonality. The ability to convert the signs into words or sentences is an important contribution that gives the proposed system its efficiency and ability to be applied.

The future work can be tailored to meet the needs of the deaf and mute category, enabling them to benefit from programs that contribute to conveying their desires to ordinary people. This can be achieved by developing the proposed method to learn about local sign language, as the Iraqi Ministry of Labor and Social Affairs has a guide for signs that they can use and apply, thereby benefiting the deaf student. Create a visual representation of the sign's meaning, such as a picture of a house, from the system's output. Alternatively, the system could produce audio output instead of just text.

REFERENCES

[1] Vardhan, R.V., Girinath, B., Murali, L., Ferdin, D.J., Aswathaman, V.K. (2024). Automatic sign language recognition using convolutional neural networks. In 2024 International Conference on Science Technology Engineering and Management (ICSTEM), Coimbatore, India, pp. 1-6. https://doi.org/10.1109/ICSTEM61137.2024.10560930

[2] Goldin-Meadow, S., Brentari, D. (2017). Gesture, sign, and language: The coming of age of sign language and gesture studies. Behavioral and Brain Sciences, 40: e46. https://doi.org/10.1017/S0140525X15001247

[3] Rathi, D. (2018). Optimization of transfer learning for sign language recognition targeting mobile platform. arXiv Preprint arXiv: 1805.06618. https://doi.org/10.48550/arXiv.1805.06618

[4] Al-Hameed, W., Fadel, N. (2019). Fuzzy logic for defect detection of radiography images. Journal of Computational and Theoretical Nanoscience, 16(3): 1023-1028. https://doi.org/10.1166/jctn.2019.7992

[5] Fadel, N., Kareem, E.I.A. (2022). Computer vision techniques for hand gesture recognition: Survey. In International Conference on New Trends in Information and Communications Technology Applications, pp. 50-76. https://doi.org/10.1007/978-3-031-35442-7_4

[6] Sarma, D., Bhuyan, M.K. (2021). Methods, databases and recent advancement of vision-based hand gesture recognition for HCI systems: A review. SN Computer Science, 2(6): 436. https://doi.org/10.1007/s42979-021-00827-x

[7] Xi, C., Chen, J., Zhao, C., Pei, Q., Liu, L. (2018). Real-time hand tracking using kinect. In Proceedings of the 2nd International Conference on Digital Signal Processing, pp. 37-42. https://doi.org/10.1145/3193025.3193056

[8] Jiang, X., Zhang, Y.D. (2019). Chinese sign language fingerspelling via six-layer convolutional neural network with leaky rectified linear units for therapy and rehabilitation. Journal of Medical Imaging and Health Informatics, 9(9): 2031-2090. https://doi.org/10.1166/jmihi.2019.2804

[9] Camgoz, N.C., Koller, O., Hadfield, S., Bowden, R. (2020). Sign language transformers: Joint end-to-end sign language recognition and translation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 10023-10033.

[10] AlKhuraym, B.Y., Ismail, M.M.B., Bchir, O. (2022). Arabic sign language recognition using lightweight cnn-based architecture. International Journal of Advanced Computer Science and Applications, 13(4). https://doi.org/10.14569/IJACSA.2022.0130438

[11] Hussain, M.J., Shaoor, A., Alsuhibany, S.A., Ghadi, Y.Y., Al Shloul, T., Jalal, A., Park, J. (2022). Intelligent sign language recognition system for e-learning context. Computers, Materials & Continua, 72: 5327-5343. http://doi.org/10.32604/cmc.2022.025953

[12] Veronica, P.G., Mokkapati, R.K., Jagupilla, L.P., Santhosh, C. (2023). Static hand gesture recognition using novel convolutional neural network and support vector machine. International Journal of Online & Biomedical Engineering, 19(9). http://doi.org/10.3991/ijoe.v19i09.39927

[13] Hama Rawf, K.M., Abdulrahman, A.O., Mohammed, A.A. (2024). Improved recognition of Kurdish sign language using modified CNN. Computers, 13(2): 37. https://doi.org/10.3390/computers13020037

[14] Damaneh, M.M., Mohanna, F., Jafari, P. (2023). Static hand gesture recognition in sign language based on convolutional neural network with feature extraction method using ORB descriptor and Gabor filter. Expert Systems with Applications, 211: 118559. https://doi.org/10.1016/j.eswa.2022.118559

[15] Kareem, E.I.A., Alsalihy, W.A.A., Jantan, A. (2012). Multi-Connect Architecture (MCA) associative memory: A modified Hopfield neural network. Intelligent Automation & Soft Computing, 18(3): 279-296.

[16] Lodhi, B., Kang, J. (2019). Multipath-DenseNet: A supervised ensemble architecture of densely connected convolutional networks. Information Sciences, 482: 63-72. https://doi.org/10.1016/j.ins.2019.01.012

[17] Abdul, W., Alsulaiman, M., Amin, S.U., Faisal, M., Muhammad, G., Albogamy, F.R., Bencherif, M.A., Ghaleb, H. (2021). Intelligent real-time Arabic sign language classification using attention-based inception and BiLSTM. Computers and Electrical Engineering, 95: 107395. https://doi.org/10.1016/j.compeleceng.2021.107395

[18] Das, P., Ahmed, T., Ali, M.F. (2020). Static hand gesture recognition for American sign language using deep convolutional neural network. In 2020 IEEE Region 10 symposium (TENSYMP). pp. 1762-1765. https://doi.org/10.1109/TENSYMP50017.2020.9230772

[19] Elatawy, S.M., Hawa, D.M., Ewees, A.A., Saad, A.M. (2020). Recognition system for alphabet Arabic sign language using neutrosophic and fuzzy c-means. Education and Information Technologies, 25(6): 5601-5616. https://doi.org/10.1007/s10639-020-10184-6

[20] Yaseen, Kwon, O.J., Kim, J., Jamil, S., Lee, J., Ullah, F. (2024). Next-gen dynamic hand gesture recognition: MediaPipe, inception-v3 and LSTM-based enhanced deep learning model. Electronics, 13(16): 3233. https://doi.org/10.3390/electronics13163233