symmetry

MDPI

Article

Hand Gesture Recognition with Symmetric Pattern under Diverse Illuminated Conditions Using Artificial Neural Network

Muhammad Haroon [1], Saud Altaf [1,*], Shafiq Ahmad [2], Mazen Zaindin [3], Shamsul Huda [4] and Sofia Iqbal [5]

[1] University Institute of Information Technology, Pir Mehr Ali Shah Arid Agriculture University, Rawalpindi 46300, Pakistan
[2] Industrial Engineering Department, College of Engineering, King Saud University, Riyadh 11421, Saudi Arabia
[3] Department of Statistics and Operations Research, College of Science, King Saud University, Riyadh 11451, Saudi Arabia
[4] School of Information Technology, Deakin University, Burwood, VIC 3128, Australia
[5] Space and Upper Atmosphere Research Commission, Islamabad 44000, Pakistan
* Correspondence: saud@uaar.edu.pk

Abstract: This paper investigated the effects of variant lighting conditions on the recognition process. A framework is proposed to improve the performance of gesture recognition under variant illumination using the luminosity method. To prove the concept, a workable testbed has been developed in the laboratory by using a Microsoft Kinect sensor to capture the depth images for the purpose of acquiring diverse resolution data. For this, a case study was formulated to achieve an improved accuracy rate in gesture recognition under diverse illuminated conditions. For data preparation, American Sign Language (ASL) was used to create a dataset of all twenty-six signs, evaluated in real-time under diverse lighting conditions. The proposed method uses a set of symmetric patterns as a feature set in order to identify human hands and recognize gestures extracted through hand perimeter feature-extraction methods. A Scale-Invariant Feature Transform (SIFT) is used in the identification of significant key points of ASL-based images with their relevant features. Finally, an Artificial Neural Network (ANN) trained on symmetric patterns under different lighting environments was used to classify hand gestures utilizing selected features for validation. The experimental results showed that the proposed system performed well in diverse lighting effects with multiple pixel sizes. A total aggregate 97.3% recognition accuracy rate is achieved across 26 alphabet datasets with only a 2.7% error rate, which shows the overall efficiency of the ANN architecture in terms of processing time.

Keywords: American Sign Language; gesture recognition; variant lighting conditions; symmetric pattern; accuracy

Citation: Haroon, M.; Altaf, S.; Ahmad, S.; Zaindin, M.; Huda, S.; Iqbal, S. Hand Gesture Recognition with Symmetric Pattern under Diverse Illuminated Conditions Using Artificial Neural Network. *Symmetry* **2022**, *14*, 2045. https://doi.org/10.3390/sym14102045

Academic Editor: José Carlos R. Alcantud

Received: 25 August 2022
Accepted: 26 September 2022
Published: 30 September 2022

1. Introduction

A hand gesture is simply defined as "the known movement of a single or both hands". A Gesture usually conveys a precise single message. For example, waving a hand is to say 'Good Bye', and waving both hands shows excitement [1]. For humans, the eyes capture the hand movement, and the brain processes the image to recognize a gestural movement. In this process, the combination of sharp vision and right visibility produces a perfect picture in the brain. This results in accurate human gesture recognition. Similarly, machines require complete and clear input information for accurate vision-based gesture recognition. In addition, an efficient embedded system is required to extract meaningful information from the environment.

In a vision-based gestural system, expressive input is required to recognize gestures with human-like accuracy. The input types used for vision-based gesture recognition

technology are static and dynamic gestures. A static gesture image contains the hand in a static state. This means that the posture of the hand, its position, and shape remain the same. On the other hand, a dynamic gesture comprises a sequence of still photos that have been acquired frame by frame. Dynamic gestures can match the characteristics of actual hand movement by having different images express the gestural movement from start to finish. The accuracy of the dynamic gestural recognition system relies on an accurate static gesture recognition model [2]. This is because factors that affect the static gesture recognition process ultimately affect dynamic gesture recognition.

Image noise is one such factor that reduces the accuracy of the recognition system. Noise is defined as the undesired information that comes along with the input image. An example of this is the variation of brightness level within an image. This illumination variation occurs either because of the image sensor or the image surroundings. The change in the number of lumens (lm) emitted by the lighting device is referred to as brightness variation. More lumens mean brighter light, fewer lumens mean dim light, and in between is ambient or general lighting. Usually, rooms and kitchens are set to general lighting, whereas bathrooms have brighter lighting and bedrooms have dim lighting. Real-time recognition becomes complex when the user is moving around the house in different lighting conditions where the brightness level varies unexpectedly. Hence, image enhancement is necessary for accurate recognition.

Many researchers have proposed different image enhancement techniques to improve the feature extraction and overall gestural recognition process. Some of the applications, such as sign language recognition, need a high level of precision and reliability in detecting the hand and recognizing gestures. Therefore, distinguishing characteristics need to be identified for this purpose. However, the vast majority of efforts have only looked at a single lighting scenario to recognize the gesture pattern, so there is still a lot of room for improvement in the area of gesture recognition under varying lighting scenarios. In this study, symmetric patterns and a related luminosity-based filter are considered for use in gesture recognition. The main contributions of this paper are as follows:

- Firstly, we proposed a symmetry-pattern-based gesture recognition framework that works well in diverse illumination lighting effects.
- Secondly, the dataset is created based on 26 American Sign Language (ASL) hand gesture images under diverse illumination conditions. Then, an efficient method for gesture feature extraction is used that is based on luminosity-based grey-scale image conversion and perimeter feature extraction.
- Thirdly, segmentation and identifying the significant points to enhance the number of Scale-Invariant Feature Transform (SIFT) key points and minimized the time taken for key point localization within features.
- Then, the gesture recognition process is validated by different Artificial Neural Network (ANN) architectures to enhance the recognition accuracy rate and avoid any uncertainty management in decision-making.
- Finally, a comparison has been performed between our work and other available researchers' published work in a similar domain to show the efficiency of our proposed framework process.

The organization of the paper is as follows in different sections: Section 2 literature review. Section 3 briefly explains the proposed hand gesture recognition framework with mathematical modelling. Section 4 demonstrated the testbed environment and results. Finally, Section 5 discussed the conclusion and possible future work.

2. Literature Review

This section reviews the most relevant state-of-the-art literature on gesture detection in a variant illuminated background environment, considering its application in ASL. Researchers [3–6] have proposed different techniques to address the illumination variation problem in vision-based hand gesture recognition systems. For sign language recognition covering languages of their origin, that may affect image recognition. In a study [7], the

approach of pattern recognition for surface electromyography (sEMG) signals of nine different finger movements is described. The authors in [8] proposed log-spiral codes of symmetric patterns in the unique method that was developed to identify human hands and understand motions from video streams using long spiral codes. In a recent study [9], the authors present a symmetric CNN called HDANet. This CNN is built on the self-attention mechanism of the Transformer and makes use of symmetric convolution in order to capture the relationships of image information in two dimensions, specifically spatial and channel. In research [10], using a generative adversarial network to capture the implicit relationship between glyphs from Oracle Bone Characters and modern Chinese characters was the basis of a research project that proposed a method for image translation from Oracle Bone Characters to modern Chinese characters. This method could translate images from Oracle Bone Characters to modern Chinese characters. Another research [11] presents a collaborative surgical robot system for percutaneous treatment directed by hand gestures and supplemented by an AR-based surgical field. The use of hand gestures to instruct the surgical robot improved needle insertion accuracy in experiments. Whereas [12] proposed a depth-based palm biometrics system. The technology splits the user's palm and retrieves finger dimensions from the depth picture. In addition, studies [13–15] present a thorough review of hand gesture techniques to eliminate the effect of illumination variations. Some of the most common classifiers proposed by the researchers include k-NN, presented in [12], the SVM classifier discussed in [16], and the tree-based random forest classifier elaborated in [17].

The authors in [18] presented a novel recognition algorithm based on a double-channel convolutional neural network, which separates the varying illumination from the gesture. The study [19], recognized sign language gestures in seven categories using visibility, shape, and orientation of the hand features. At the preprocessing step, they applied skin detection using colour properties in the HSV domain to form a uniform linear binary pattern. The multiclass Support Vector Machine classifier classified the images from the dataset of 3414 signs corresponding to 37 Pakistan Sign Language alphabets with good categorization results. Chen et al. [20] proposed an event-based system that uses a biologically inspired neuromorphic vision sensor, an encoding process to identify objects, and a flexible system to classify hand movements. According to [21], the fitness function for gamma correction preserves the brightness and details of the image of both brighter and low-contrast images. Particle Swarm Optimization can be applied to make the gamma correction adaptive by calculating the optimal gamma values.

As shown by [22], a method for compensating for the poor ambient illumination in the scene is by balancing it against incident illumination. Demonstrated by [23], details about imaging hardware, the collection procedure, the organization of the database, several potential uses of the database, and how to obtain the database. The study collected a database of over 40,000 images of 68 people. Each person is captured in 13 different poses, under 43 different illumination conditions, and with 4 different expressions. In another study [24], a large-scale dataset was collected with various illumination variations to evaluate the performance of the Remote Photo Plethysmography (RPG) algorithm. The study also proposed a low-light enhancement solution for remote heart rate estimation under low-light conditions. In a recent study [25], fine perceptive generative adversarial networks (FP-GANs) are proposed to construct super-resolution (SR) MR images from low-resolution equivalents. FP-GANs use a divide-and-conquer strategy to process low-and high-frequency MR image components individually and in tandem. In another study [26] mild cognitive impairment and Alzheimer's disease are assessed using a tensorizing GAN with high-order pooling. The proposed model can benefit from brain structure by tensorizing a three-player cooperative gaming framework. By introducing high-order pooling into the classifier, the suggested model can employ second-order MRI statistics (MRI). The study [27] proposed state-of-the-art XAI algorithms for EMG hand gesture classification to understand the outcome of machine learning models with respect to physiological processes to recognise hand movements by mapping and merging synergic muscle activity.

The systematic review shown in Table 1 that highlights the potential related research focused on diverse illumination variation factors is given by.

Table 1. Comparison of various existing techniques under different illuminated conditions.

Paper	# of Gestures	Technique	Lighting Changes	Background
[18]	10	DC-CNN	2-Variant	Redundant
[28]	8	CNN	N/A	Cluttered
[29]	10	ANN	2-Variant	Colourful
[30]	8	3D-CNN	variant	Occlusion
[31]	24	Darknet	2-Variant	N/A
[32]	40	D Learn	2-Variant	Cluttered
[33]	24	DC-CNN	Dissimilar	Noise
[34]	24	ANN	Artificial	Cluttered

3. Materials and Methods

This section proposed a framework of sensor-based sign language gesture recognition which consists of the following main phases, i.e., acquiring the image, image preprocessing, feature extraction, and classification, as shown in Figure 1.

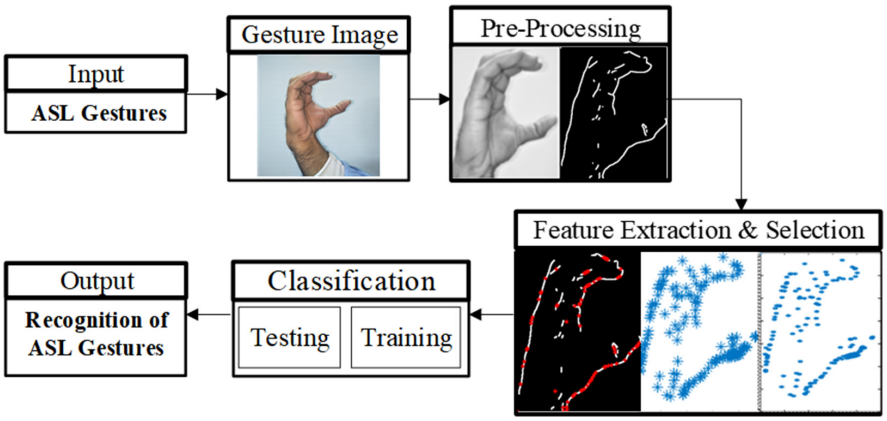


Figure 1. Proposed framework of sensor-based sign language gesture recognition.

The recognition process starts with the acquisition of depth images using the Kinect sensor. Each pixel in a depth image reflects the distance between the image plane and an RGB image object. Following that, hand shapes are precisely segmented in order to locate and track the hands' shapes, similar to the human's communication approach. To develop the datasets, ASL-based gesture images are stored in a database and converted into PNG format. The next step is the pre-processing of the acquired images to transform the captured image into a uniform level of brightness. For that, the system performs the luminosity method based on the grey-scale conversion of the input image. Grayscale conversion reduces complexity and is much easier to work with a variety of tasks such as image segmentation problems. Greyscale conversion is carried out through the weighted method, also called the luminosity method. The main reason for proposing the luminosity method is to equalize the weights of red, green, and blue according to their wavelengths. The luminosity method is a better version of the average method. As discussed in [35], luminosity-based greyscale conversion can be calculated as follows:

$$\text{Grayscale} = 0.299R + 0.587G + 0.114B \quad (1)$$

where R, G, and B represent red, green, and blue colours, respectively.

The next step is to extract the appropriate features and their selection. The selection of the number of features is a critical step because more features consume additional space and computational time. Fewer features affect the accuracy. For that, the SIFT method is proposed to select and extract the appropriate features from acquired data. The proposed

method extracts four significant features (perimeter, hand size, centre of hand and finger distance) from a given input image.

To define the shape of the hand and calculate the perimeter value, the perimeter feature extraction (PFE) technique is used to detect the edges and boundaries of the human hand by counting the pixels having values of 1 and 0 for neighboring pixels while skipping the grey shades. The shape is calculated by finding the projection of the hand that provides the size of the hand. For that, vertical and horizontal values are calculated by adding up all the values of rows and columns as follows:

$$v_i\ (c) = \sum_{c=0}^{n-1} P_i(r, c) \tag{2}$$

$$h_i\ (r) = \sum_{r=0}^{n-1} P_i(r, c) \tag{3}$$

where v and h represent the vertical and horizontal positions, and r and c represent the row and column, respectively, of pixel P in an image. In Equation (2), the letter "n" stands for the maximum "height" value, which is the vertical side of a hand. In Equation (3), the letter "n" stands for the maximum "width" value, which is the horizontal side of a hand.

The hand size feature is useful to recognize the change in the size of a hand because hand size differs from person to person. The size component of a hand represents the hand size at a specific time. To calculate the hand size, we define the function, *Mi(r, c)* as in Equations (4) and (5).

$$M_i\ (r, c) = \left\{ \begin{array}{cc} 1\ if\ P(r, c) = & i^{th} \text{ object number} \\ 0 & \text{otherwise} \end{array} \right\} \tag{4}$$

$$A_i = \sum_{r=0}^{n-1} \sum_{c=1}^{n-1} M_i(r, c) \tag{5}$$

where r and c represent the row and column, respectively, of pixel P in an image. Then, the area *Ai* is measured in pixels and indicates the relative size of the hand.

The centre of the hand feature recognizes the orientation and position of the hand. Finding the centre of the object is important for detecting any change in hand shape, palm position, and movement of the hand or fingers. We can define the centre of hand by the pair (r_i, c_i) for rows and columns, respectively, measured as follows:

$$r_i = \frac{1}{A_i} \sum_{r=0}^{n-1} \sum_{c=1}^{n-1} r\ M_i(r, c) \tag{6}$$

$$c_i = \frac{1}{A_i} \sum_{r=0}^{n-1} \sum_{c=1}^{n-1} c\ M_i(r, c) \tag{7}$$

where both r and c represent the row and column, respectively, of pixel P in an image, and *Mi(r, c)* is a size function.

Finally, the fingers position is calculated by finding the distance between two open or closed fingers. The finger's distance feature helps define the gesture. It can be carried out by counting the continuous pixel having a value of 0 until a neighboring pixel with a value of 1. The significant points can be calculated as follows:

$$Sp = (\text{avg } x, \text{avg } y) \tag{8}$$

$$Sp = \left(\frac{1}{5} \sum_{p=sx}^{p=x+5}, \frac{1}{5} \sum_{p=sy}^{p=y+5} + b \right) \tag{9}$$

where Sp is a significant point plotted on the plane averaging 5-pixel values along the x-axis and y-axis, respectively, sx is the starting point along the x-axis averaging the next five values, sy is the starting point along the y-axis averaging the next five-pixel values.

Additionally, b is a bias that gives the neural network an extra parameter to tune by initializing non-zero random values.

After that, extracted features are combined into the form of a feature vector set F_s, the data is displayed in the form of these symmetric patterns. To measure the boundary, size, and orientation of the hand for a particular gesture defined as

$$F_s = [Perimeter, Hand\ Size,\ Center\ of\ Hand,\ Finger\ Position] \tag{10}$$

where

$$Perimeter\ F_1 = [a_1, a_2, a_3, \ldots, a_n] \tag{11}$$

$$Hand\ Size\ F_2 = [b_1, b_2, b_3, \ldots, b_n] \tag{12}$$

$$Center\ of\ Hand\ F_3 = [x_1, x_2, x_3, \ldots, x_n] \tag{13}$$

$$Finger\ Position\ F_4 = [y_1, y_2, y_3, \ldots, y_n] \tag{14}$$

where the letter "n" denotes the total number of extracted 'feature points' for each defined feature (F_1, F_2, F_3, and F_4) for a hand gesture. The symmetric feature set enables the description of symmetric patterns such as the perimeter, center, finger position, and the size of the hand. Where 'perimeter features' outline the physical shape of the hand, 'size feature' extracts features for comparison of scale variation of the hand, 'center of hand' handles the orientation of the hand, and 'finger position' feature measures the distance between two fingers. The feature set created in matrix form shows all the features combined in Equation (15).

$$X = \begin{bmatrix} F_1 \\ F_2 \\ F_3 \\ F_4 \end{bmatrix} \Rightarrow \begin{bmatrix} a_1\ a_2\ a_3 \ldots a_n \\ b_1\ b_2\ b_3 \ldots b_n \\ x_1\ x_2\ x_3 \ldots x_n \\ y_1\ y_2\ y_3\ \ldots y_n \end{bmatrix} \Rightarrow \begin{bmatrix} 0\,0\,0\,1\,1\,0 \ldots 0 \\ 0\,0\,1\,1\,0\,0 \ldots 1 \\ 0\,0\,1\,1\,1\,0 \ldots 0 \\ 0\,1\,1\,1\,0\,0 \ldots 1 \end{bmatrix} \tag{15}$$

where each pixel corresponds to a matrix element value. "0" and "1" represent pixel values, with 1 indicating the presence of a feature point and 0 indicating its absence. Finally, the SIFT technique is employed, which was proposed by Lowe, D.G. [36] and later the anatomical structure was discussed by the authors in their work [37]. The SIFT technique extracts features and identifies the significant key points of ASL-based images with their relevant features in diverse illumination conditions. The SIFT algorithm is formulated as follows in Algorithm 1.

Algorithm 1: SIFT Keypoints Generation

1: Gaussian scale-space computation
2: **Input:** i image
3: **Output**: s scale-space
4: Difference of Gaussians (DoG)
5: **Input:** s scale-space
6: **Output:** d DoG
7: Finding keypoints (extrema of DoG)
8: **Input:** d DoG
9: **Output**: {(rd, cd, αd)} list of discrete extrema (position and scale)
10: Keypoints localization to sub-pixel precision
11: **Input:** d DoG and {(rd, cd, αd)} discrete extrema
12: **Output**: {(r, c, α)} extreme points
13: Filter unstable extrema
14: **Input:** d DoG and {(r, c, α)}
15: **Output**: {(r, c, α)} filtered keypoints
16: Filter poorly localized keypoints on edges
17: **Input:** d DoG and {(r, c, α)}
18: **Output**: {(r, c, α)} filtered keypoints
19: Assign a reference orientation to each keypoint
20: **Input:** (∂mv, ∂nv) scale-space gradient and {(r, c, α)} list of keypoints
21: **Output**: {(x, y, α, θ)} list of oriented keypoints
22: SIFT Feature descriptor generator
23: **Input:** (∂mv, ∂nv) scale-space gradient and {(x, y, α, θ)} list of keypoints
24: **Output**: {(r, c, 1: α, θ, f)} list of described keypoints

4. Experimentation and Results

To prove the concept, a workable testbed has been developed in the laboratory by using the Microsoft Kinect sensor to capture the images and convert them into depth images for acquiring the diverse resolution data, as shown in Figure 2.

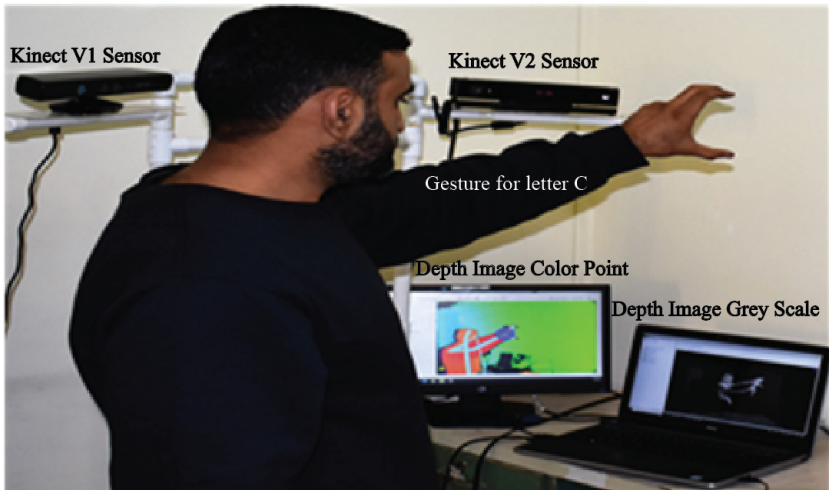


Figure 2. Data acquisition setup for development of case study.

For the development of a case study, a total of three subjects were considered to acquire the data in diverse illuminated conditions and save it into PNG format according to Equation (1), as shown in Figure 3.

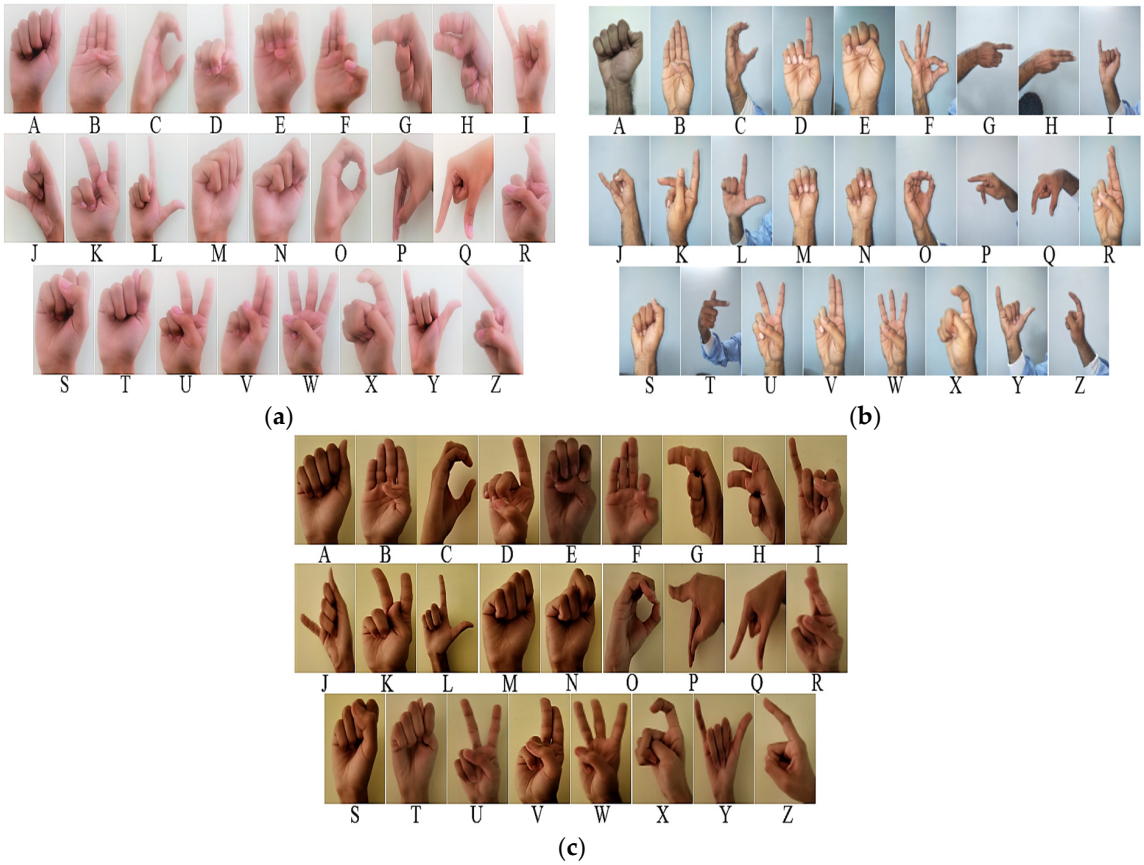


Figure 3. Data acquisition of ASL Images under (**a**) bright light, (**b**) ambient light, (**c**) dark light.

The subsequent step is to convert the ASL images into grayscale for the segmentation to separate the hand object from its background, as shown in Figure 4. The next stage is to calculate the SIFT points from the segmented hand objects to identify the feature points as mentioned in Equation (15). A feature descriptor method is used to process the significant image points from identified feature points to convert them into significant vector points. For that, the Matlab tool is used to extract the required feature point values and convert them into significant points using the SIFT algorithm from 26 letters of the alphabets.

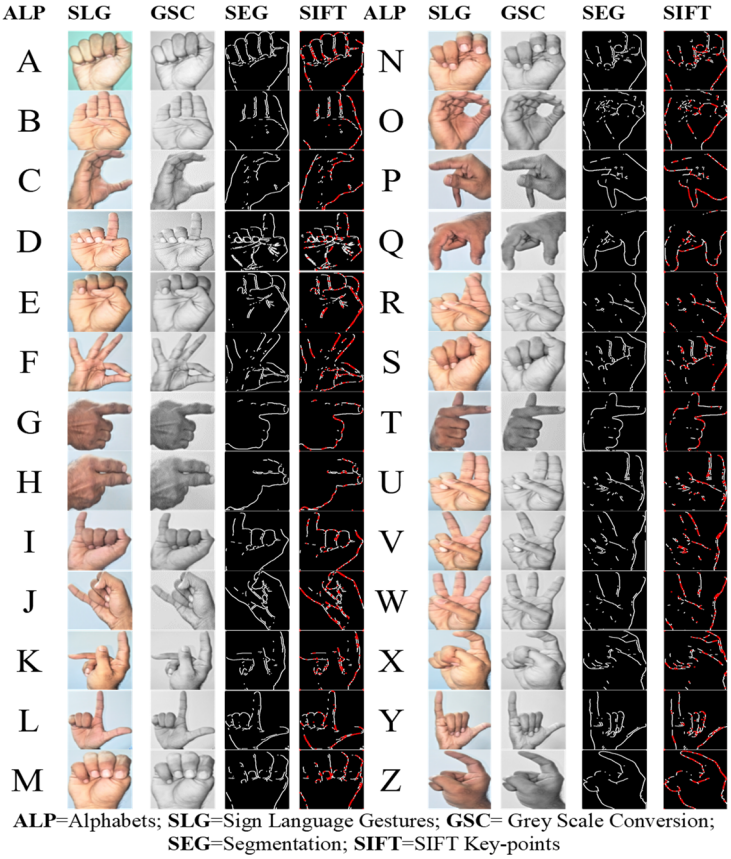


Figure 4. ASL conversion into grey scale and segmentation.

After calculating the SP using SIFT, the average processing time of every step is calculated under different lighting conditions and resolution rates as shown in the following Table 2. As we can observe from Table 2, the reasonable processing time is at a 1024 × 768-resolution rate in ambient light conditions, which is a good resolution rate for analysis. It is noticed that higher resolution rates consume more processing time. Therefore, we will consider the 1024 × 768 resolution rate at the next stage.

Table 2. Average processing time under variant lightening conditions at different resolution.

Resolution	Average Processing Time (Sec)		
	Bright Light	Ambient Light	Dark Light
260 × 175	2.38	2.24	2.34
320 × 240	2.50	2.20	2.46
640 × 480	2.58	2.24	2.25
800 × 600	2.21	2.18	2.41
1024 × 768	2.28	2.21	2.35
2048 × 1540	2.28	2.36	2.32
4160 × 3120	2.39	2.50	2.84

After converting the 26 alphabets into grayscale, segmentation and significant point calculation of A–Z, four letters are considered to show the efficiency of the proposed framework. Letters "P", "A", "I", and "R" are used for segmentation, SIFT, and measure the significant points of the hand gesture as shown in the following Figure 5.

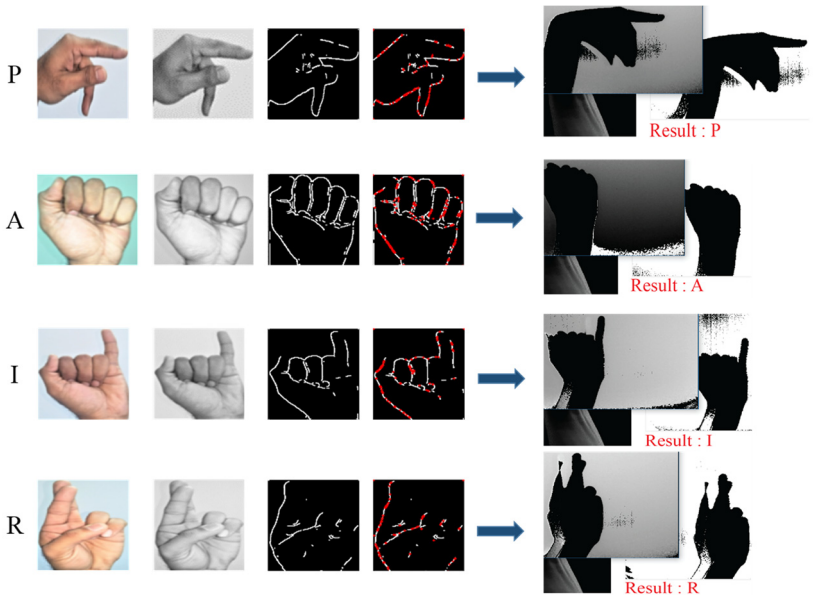


Figure 5. Case study feature and significant point's calculation using SIFT.

Referring to Equation (15), and Figure 6, we considered the mean ($\bar{x}$), standard deviation (σ), variance (μ) and average deviation (AD) values against each feature of selected SP and converted them into the compact form of featured SP, respectively. The details are shown in Table 3.

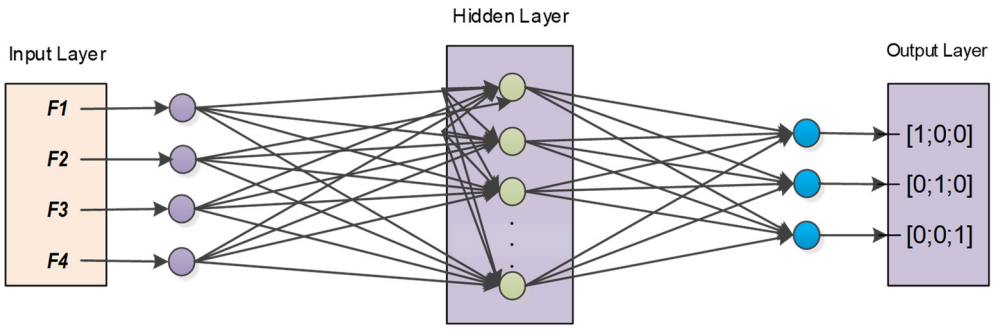


Figure 6. Proposed ANN classification architecture for the gesture.

Selection of applicable gesture features is a key task as inputs of ANN for training to measure the accuracy from acquired data and it may indistinct the network structure. However, the right feature selection improves the efficiency of the ANN network, and training time may also be reduced by adopting the right ANN hidden layer architecture according to the input and output. In classification and data processing, ANN learning accommodates a variety of conditions better than any other classification technique [38]. In this paper, we have chosen four different features as inputs (F_1, F_2, F_3 *and* F_4) for each layer of neurons. Each network consists of one hidden layer that contains multiple neurons according to the inputs. The number of hidden layer neurons has a reliable impact on the performance of the ANN model. Therefore, the selection of a number of the hidden layer neurons depends on ANN accuracy in primary trials. For the target output, a vector of classes to recognize the hand gesture is written as follows:

1. [1; 0; 0]: Hand Natural Position;
2. [0; 1; 0]: Hand Gestural Position;
3. [0; 0; 1]: Hand Unknown Position.

Table 3. Feature values calculation.

Features		P	A	I	R
F1	$\bar{x}$	348.57	335.12	278.37	433.92
	μ	18744.8	11651.6	13358.49	28356.31
	σ	136.91	107.94	115.57	168.39
	AD	108.64	76.01	91.82	125.29
F2	$\bar{x}$	362.96	317.5	432.61	402.57
	μ	17503.4	5888.75	10446.19	5307.95
	σ	132.30	76.73	102.20	72.85
	AD	103.22	70.75	76.43	43.01
F3	$\bar{x}$	456.12	392.47	457.19	413.45
	μ	3845.16	9542.87	1481.61	5662.65
	σ	62.00	97.68	38.49	75.25
	AD	47.56	79.88	30.47	63.90
F4	$\bar{x}$	494.23	527.55	519.97	540.97
	μ	5157.58	4899.46	3547.31	6015.84
	σ	71.81	69.99	59.55	77.56
	AD	61.92	57.36	48.78	67.11

A multi-layer Feedforward Neural Network (FFNN) method is used in this paper for the recognition of hand gestures from datasets. The proposed architecture of ANN for a single hand gesture is presented in Figure 7.

Based on the same ANN training process, we used a similar classification architecture for the recognition of multiple hand gestures as objects. Every hand gesture feature's vector class was used as input data and classified through a similar network architecture. The output layer in Figure 6 presents the current state of the gesture. It contains a total of four NN input nodes, and the hidden layer activation function $(logsig)$ is employed for every proposed output. All object feature values in Table 3 were stored in the *Mat* extension file and assigned these values with each hand gesture and divided all features into sub-features. The training goal is set at 0.01 target value and the Back Propagation (BP) learning method is adopted for training. Figure 7 presents the inside architecture of each neural network for each hand object.

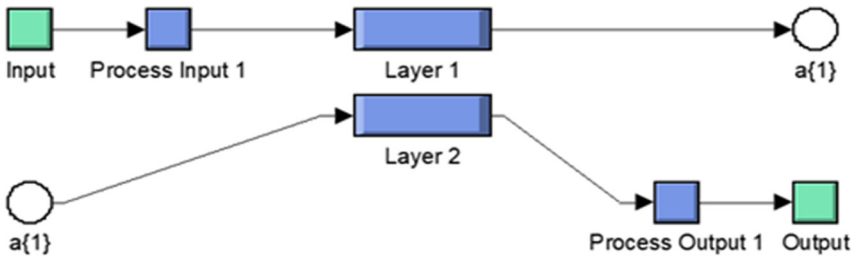


Figure 7. The internal design of NN architecture.

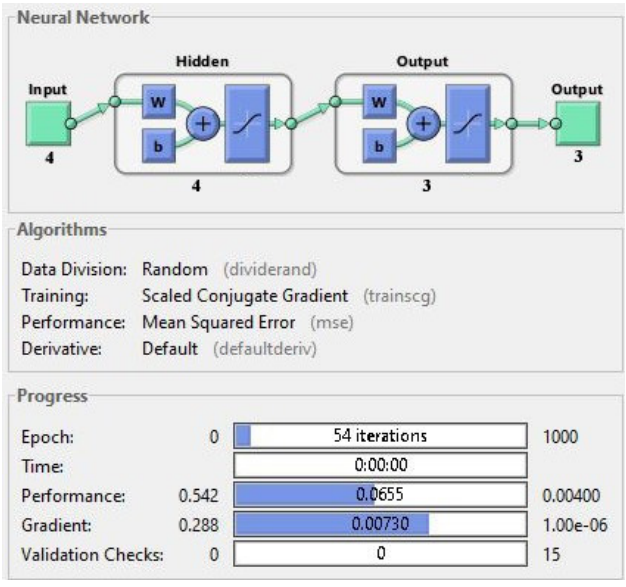
After initializing the ANN model for non-linear system modelling, specific ANN data must be assessed targeting nodes precedents. The hidden layer neurons and transfer function are set up to compute the training objective. Then, the layer weight is set for output. Table 4 shows the brief explanation and ANN layer setup information.

For the selection of suitable hidden layer neurons architecture, three types of ANN architecture (the [4 × 4 × 3], [4 × 14 × 3] and [4 × 24 × 3]) were tested in this paper for training purposes, as shown in Figure 8. To alter the weights of the hidden layer until the

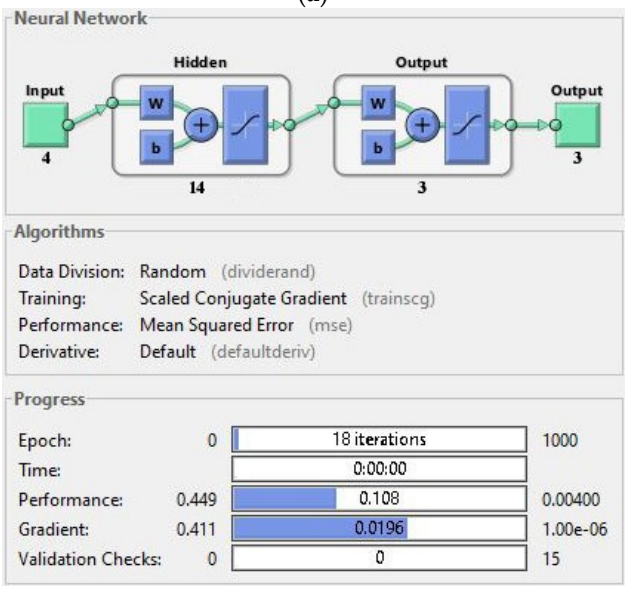
targeted output was achieved at reasonable epoch numbers with less error rate, as shown in Table 5.

Table 4. Description of the implemented ANN.

NN Steps	Artificial Neural Network Structure for Performance Matrices
Network Mode	FFNN
Learning Pattern	Back Propagation
Training Goal	0.001
Input data	Four inputs of 1D ANN matrix where all data were placed in each image's class for recognition process index
No. of neurons in hidden layer	Diverse N architectures are used with different values of neurons inside hidden layer. For example, [4 × 4 × 3], [4 × 14 × 3] and [4 × 24 × 3] (see Figure 9).
Vector of classes for the target outputs	Mathematical matrices refer to the classified vector classes with value 0 or 1.

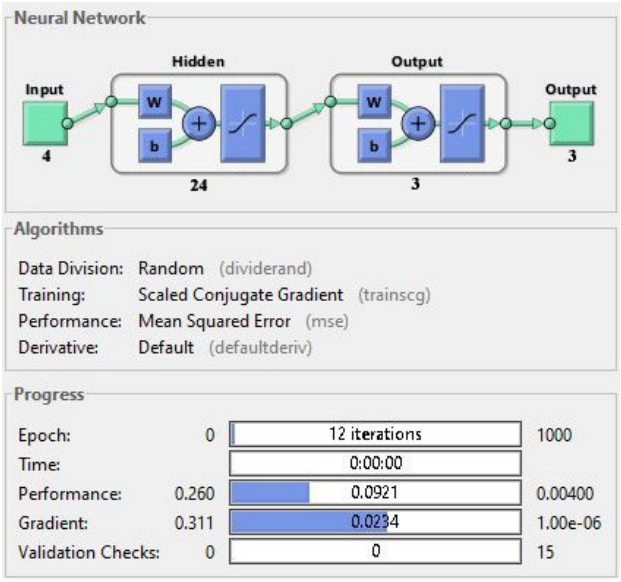


(**a**)



(**b**)

Figure 8. *Cont.*



(**c**)

Figure 8. Overview of the different ANN architectures chosen: (**a**) [4 × 4 × 3]; (**b**) [4 × 14 × 3]; (**c**) [4 × 24 × 3].

It can be observed from Table 5 that the selected architecture [4 × 14 × 3] has presented better mean squared error (MSE) performance with reasonable epoch numbers and error rate than other ANN architectures. The next process is to measure the validation of acquired features in Table 2. Figures 9–12 show the training performance graph of the ANN architecture [4 × 14 × 3], which attained a good and considerable performance result during ANN testing.

Table 5. Different ANN architecture for classification performance.

Arch	Sample	MSE	No. of Epoch	Accuracy	Classification Error
[4 × 4 × 3]	F1	7.56×10^{-2}	70	93.6	6.4
	F2	7.22×10^{-2}	62	92.7	7.3
	F3	6.56×10^{-2}	72	93.1	6.9
	F4	7.22×10^{-2}	98	93.9	6.1
[4 × 14 × 3]	F1	8.96×10^{-2}	114	96.7	3.3
	F2	8.75×10^{-2}	122	96.8	3.2
	F3	7.5×10^{-2}	130	97.4	2.6
	F4	9.28×10^{-2}	125	97.1	2.9
[4 × 24 × 3]	F1	7.65×10^{-2}	372	93.7	6.3
	F2	6.22×10^{-2}	304	92.8	7.2
	F3	7.90×10^{-2}	385	90.9	9.1
	F4	7.56×10^{-2}	374	90.4	9.6

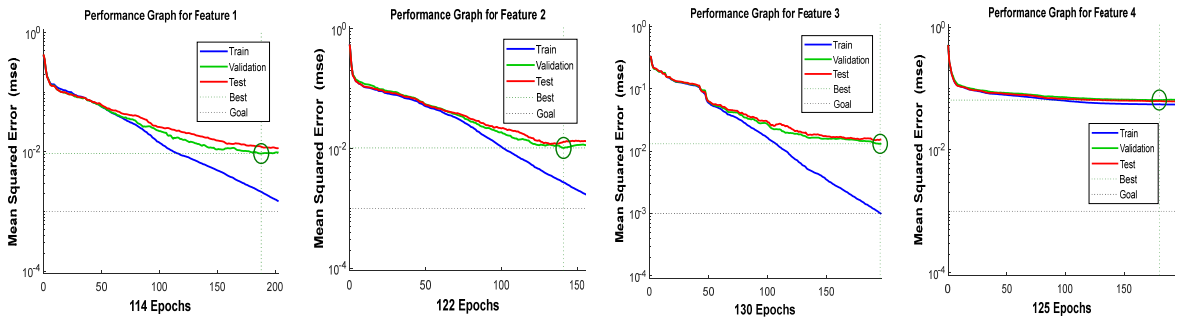


Figure 9. Performance graphs using [4 × 14 × 3] neural network architecture for gesture **P**.

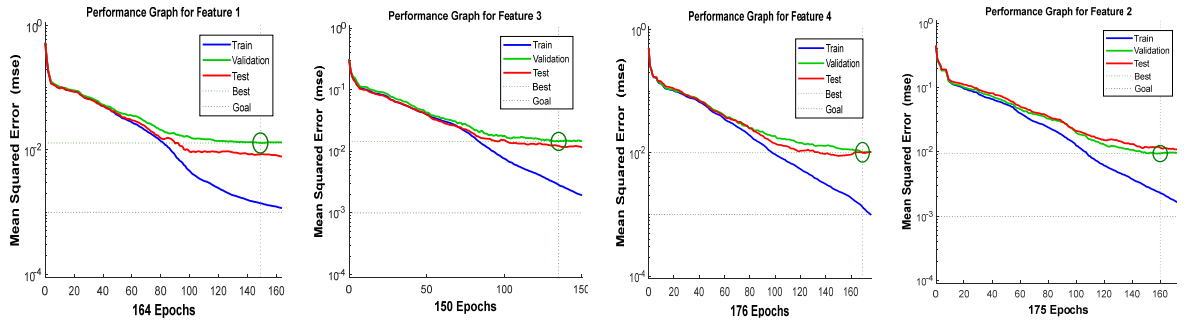


Figure 10. Performance graphs using [4 × 14 × 3] neural network architecture for gesture **A**.

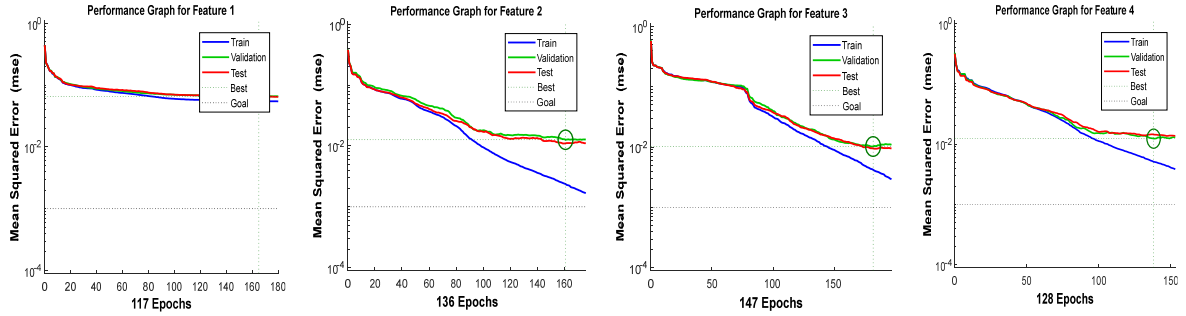


Figure 11. Performance graphs using [4 × 14 × 3] neural network architecture for gesture **I**.

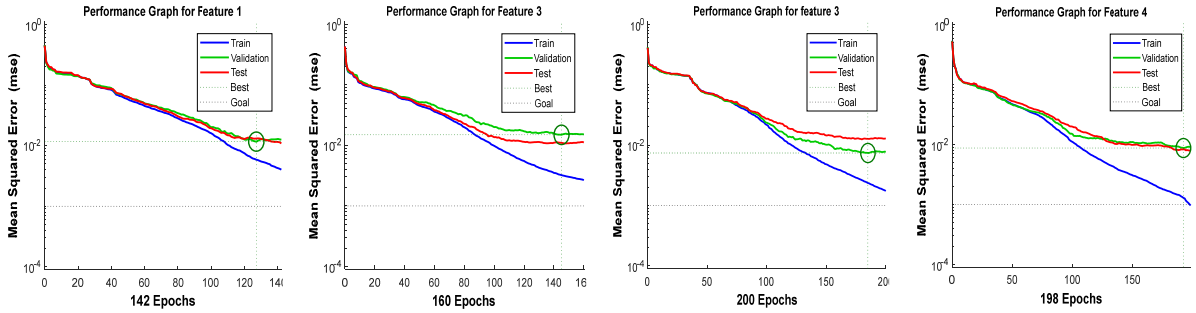


Figure 12. Performance graphs using [4 × 14 × 3] neural network architecture for gesture **R**.

After measuring and testing the performance graph of the ANN architecture [4 × 14 × 3], the next stage of validation is to calculate the accuracy by calculating the Confusion Matrix (CM). To construct the CM, the four features input $(F_1,\ F_2, F_3\ and\ F_4)$ values are inserted into ANN architecture in Figure 7 by adjusting the height of the hidden layer. The combined confusion matrices of all features against each character of the *PAIR* word are shown in Figures 13–16.

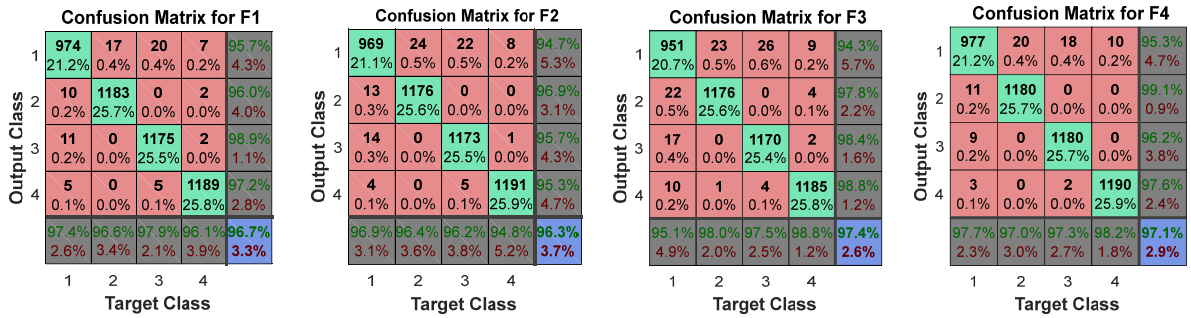


Figure 13. Confusion matrices for letter **P** on architecture [4 × 14 × 3].

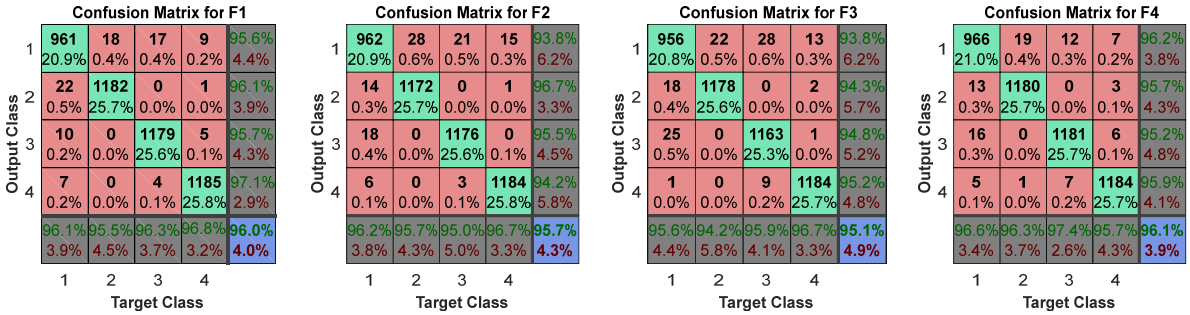


Figure 14. Confusion matrices for letter **A** on architecture [4 × 14 × 3].

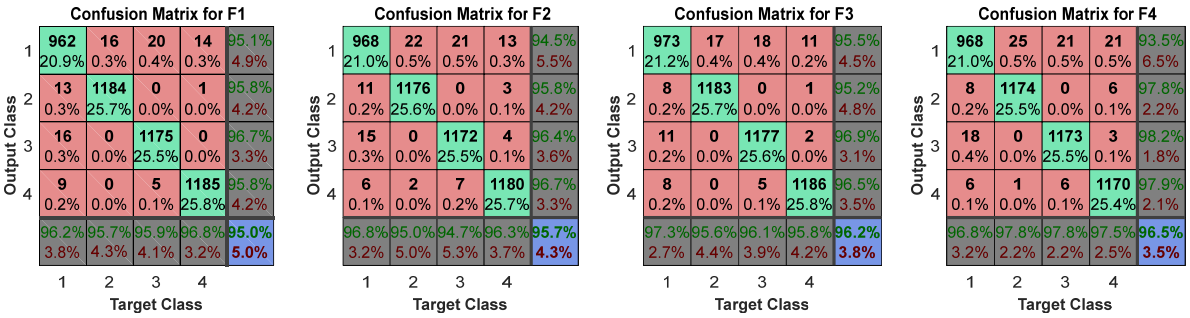


Figure 15. Confusion matrices for letter I on architecture [4 × 14 × 3].

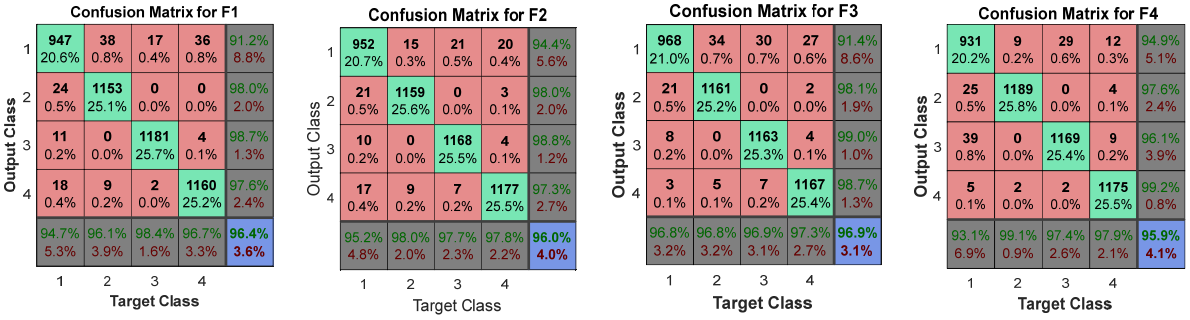


Figure 16. Confusion matrices for letter **R** on architecture [4 × 14 × 3].

In Figures 13–16 above, each corner cell shows the accurately tested pattern cases of gestures through the proposed ANN architecture [4 × 14 × 3] to decide the recognition of the right-hand gesture. In the confusion matrices graph, the confusion grid holds the features processed training data between the target and output classes, consisting of three procedural stages: preparing, testing, and training of the gesture recognition and individually measuring the performance of the ANN architecture.

To perform these procedural stages, four horizontal target and vertical output classes were defined to demonstrate the precise data validation testing process to reflect all possible targeted sample values of feature sets. The green cells in the CM grid graphs show those data groups that are accurately classified and have completed a successful training process. Each grey corner cell in horizontal position shows those data groups of targeted classes that are accurately classified and complete the testing phase in the training process. The red cell presents those data sets that are wrongly classified or might not be properly validated in the testing phase. Finally, the blue cell presents the overall percentage of correctly classified gesture test cases from datasets. From confusion matrix diagrams, we can easily observe that each class has been tested under 1200 test cases and show percentages in green cells to observe the targeted class output parentage with error rates which are accurately classified during the testing phase with less than 1% wrongly classified in all trained datasets. Overall, a maximum 97.4% accurate rate of the word "PAIR" was achieved in the blue cell with only

a 2.6% error rate, which shows the overall efficiency of the ANN architecture in terms of processing time.

After measuring, testing, and calculating the accuracy of four selected features, the next step is to measure the accuracy of the whole dataset (A–Z). For that, the same ANN architecture [4 × 14 × 3] is utilized with the previous configuration settings that were used in the case study. Figure 17 shows the training performance graph of ANN architecture [4 × 14 × 3], NN testing produced an excellent and significant performance result. The combined confusion matrix of all chosen features against each character (A–Z) word is shown in Figure 17. From Figure 18, we can certainly perceive that each class has been tested under 1400 test cases from trained datasets to observe the accurately classified target output class and only less than 1% are wrongly classified in all trained datasets. Overall, a maximum of 97.3% accuracy rate of all 26 alphabets was achieved in the blue cell with only a 2.7% error rate, which shows the overall efficiency of the adopted NN architecture [4 × 14 × 3] in terms of processing time.

Finally, a comparison has been performed between our work and other researcher's published works in a similar domain to show the efficiency of our proposed framework process. Table 6 compares the performance of our and other works based on gesture image datasets and recognition methods in terms of the number of gestures, frame resolution, response time, recognition approaches, and accuracy rate with an error rate under various illuminated conditions. The combination of preprocessing process, PFE, segmentations, significant point extraction, and utilising multiple ANN architectures for classification to reduce the error rate and achieve the high accuracy rate in gesture recognition is the reason for achieving the high accuracy rate compared with others.

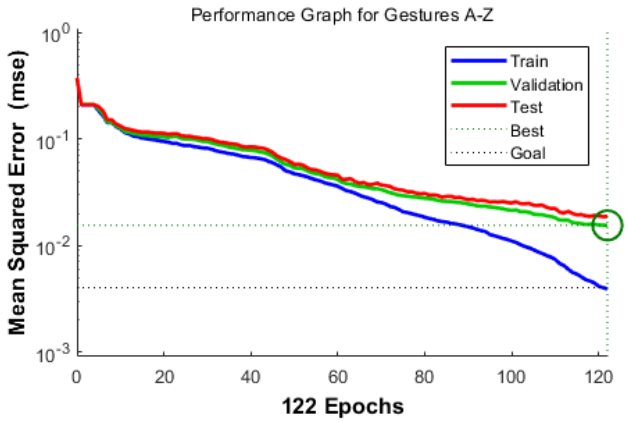


Figure 17. Performance graphs using [4 × 14 × 3] neural network architecture for the gestures A−Z.

Table 6. Performance comparison with other gesture recognition research.

Paper	# of Gestures	Test Image	Frame Resolution	Recognition Time (sec)	Technique	Accuracy (%)	Error Rate (%)
[18]	10	400	128 × 128	0.4	DC-CNN	94.8	5.2
[25]	8	195	320 × 240	0.09–0.11	CNN	93.9	6.1
[26]	10	600	512 × 424	0.133	ANN	95.6	4.4
[27]	8	220	112 × 112	0.03	3D-CNN	95.8	4.2
[28]	24	300	416 × 416	0.0666	Darknet	96.7	3.3
[29]	40	90	112 × 112	N/A	D Learn	96.2	3.8
[30]	24	66	320 × 240	N/A	DC-CNN	94.5	5.5
[31]	24	135	400 × 400	0.19	ANN	95.7	4.3
Our work	26	800	1024 × 768	0.013	ANN	97.4	2.6

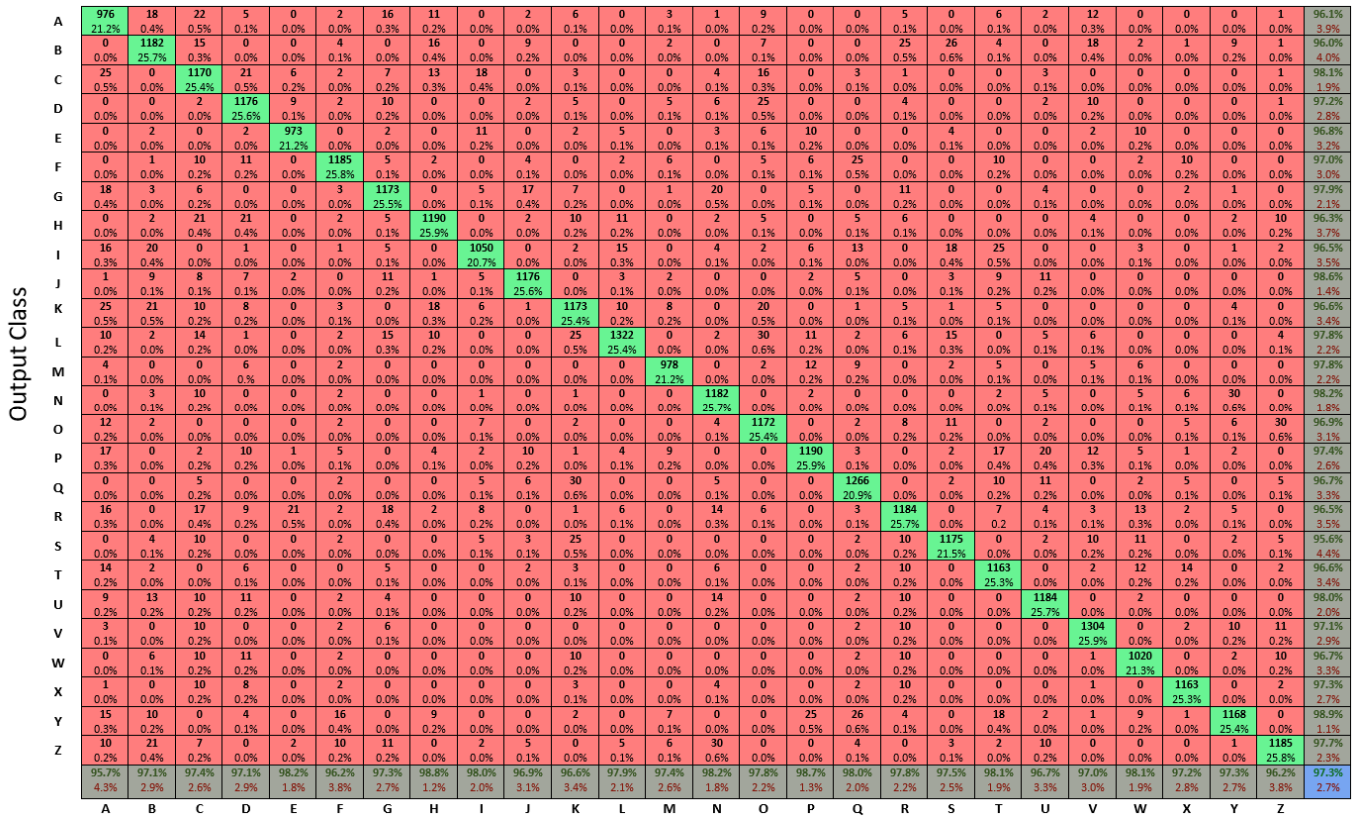


Figure 18. Confusion matrices for letters A–Z on the architecture [4 × 14 × 3].

5. Conclusions

This paper presented an efficient framework to improve the performance of gesture recognition under variant illumination using the luminosity method. Symmetric patterns and a related luminosity-based filter are considered for use in gesture recognition. The proposed framework consists of four main phases, i.e., acquiring the image, image pre-processing, feature extraction, and classification. To develop the datasets, a workable testbed has been developed in the laboratory by using two Microsoft Kinect sensors to capture the depth images for the purpose of acquiring diverse resolution data. ASL-based gesture images are stored in the database and converted into PNG format. The next step is the pre-processing of the acquired images to transform the captured image into a uniform level of brightness. For that, the system performs the luminosity method based on the grey-scale conversion of the input image. Grayscale conversion reduces complexity and is much easier to work with a variety of tasks such as image segmentation problems. Greyscale conversion was carried out through the weighted method. The next step is to extract the appropriate features and make their selection. It is a critical step because more appropriate features consume additional space and computational time. For that, the SIFT method is proposed to select and extract the appropriate features from acquired data. The proposed method extracts four significant features (perimeter, hand size, centre of hand, finger distance) from a given input image. After that, extracted features are combined into the form of a feature vector set to measure the boundary, size, and orientation of the hand for a particular gesture. Then, SIFT algorithm is to identify the significant key points of ASL-based images with their relevant features in diverse illuminated conditions. A feature descriptor method is used to calculate the significant image points from identified feature points and convert them into significant vector points. From the results, we can observe that the reasonable processing time is at a 1024 × 768 resolution rate, which is a good resolution rate for analysis. It is also noticed that higher resolution rates consume higher processing time and are shown in tabular form. Finally, we have chosen four different features as inputs (F_1, F_2, F_3 *and* F_4) for each layer of neurons. Each network consists of one hidden

layer that contains multiple neurons with NN architectures ([4 × 4 × 3], [4 × 14 × 3], and [4 × 24 × 3]), respectively. After training, it is identified that the architecture [4 × 14 × 3] presented a better mean squared error (MSE) performance with reasonable epoch numbers and error rate among other NN architectures. From confusion matrix diagrams, we can easily observe that each class has been tested under 1200 test cases and show percentages in green cells to observe the targeted class output parentage with error rates which are accurately classified during the testing phase with less than 1% wrongly classified in all trained datasets. Overall, a maximum 97.4% accurate rate of the word "PAIR" was achieved with only a 2.6% error rate, which shows the overall efficiency of the NN architecture.

Future development would be extended toward the incorporation of complex depth images with more gesture angles in other languages (regional sign language) by using multiple high-resolution camera modules. Further complexity in recognition can be created to select more features at multiple angles and lighting conditions in outdoor environments.

Author Contributions: Conceptualization, M.H., M.Z., S.A. (Shafiq Ahmad) and S.A. (Saud Altaf); methodology, M.H., S.A. (Saud Altaf) and M.Z.; software, M.H.; validation, M.H., S.A. (Shafiq Ahmad) and S.I.; formal analysis, M.H., M.Z. and S.A. (Saud Altaf); investigation, M.H.; resources, S.A. (Saud Altaf); data curation, S.A. (Saud Altaf), M.Z. and S.A. (Shafiq Ahmad); writing—original draft preparation, M.H., S.H. and S.I.; writing—review and editing, S.A. (Saud Altaf); visualization, M.H. and S.A. (Saud Altaf), M.Z.; supervision, S.H.; project administration, M.Z.; funding acquisition, M.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by King Saud University, grant number RSP-2021/387, and the APC was funded by RSP-2021/387.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The authors extend their appreciation to King Saud University for funding this work through Researchers Supporting Project number (RSP-2021/387), King Saud University, Riyadh, Saudi Arabia.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Mahmud, H.; Hasan, M.K.; al Tariq, A.; Kabir, M.H.; Mottalib, M.A. Recognition of Symbolic Gestures Using Depth Information. *Adv. Hum.-Comput. Interact.* **2018**, *2018*, 1069823. [CrossRef]
2. Sagayam, K.M.; Andrushia, A.D.; Ghosh, A.; Deperlioglu, O.; Elngar, A.A. Recognition of Hand Gesture Image Using Deep Convolutional Neural Network. *Int. J. Image Graph.* **2022**, *22*, 2140008. [CrossRef]
3. Khan, N.S.; Abid, A.; Abid, K. A Novel Natural Language Processing (NLP)–Based Machine Translation Model for English to Pakistan Sign Language Translation. *Cogn. Comput.* **2020**, *12*, 748–765. [CrossRef]
4. Rezende, T.M.; Almeida, S.G.M.; Guimarães, F.G. Development and validation of a Brazilian sign language database for human gesture recognition. *Neural Comput. Appl.* **2021**, *33*, 10449–10467. [CrossRef]
5. Van, Q.P.; Binh, N.T. Vietnamese Sign Language Recognition using Dynamic Object Extraction and Deep Learning. In Proceedings of the 2020 IEEE Eighth International Conference on Communications and Electronics, Phu Quoc Island, Vietnam, 13–15 January 2021; pp. 402–407.
6. Mustafa, M. A study on Arabic sign language recognition for differently abled using advanced machine learning classifiers. *J. Ambient Intell. Humaniz. Comput.* **2021**, *12*, 4101–4115. [CrossRef]
7. Moh, A.; Wahyu, C.; Ariyanto, M.; Mol, M.; Setiawan, J.D.; Glowacz, D. Pattern Recognition of Single-Channel sEMG Signal Using PCA and ANN Method to Classify Nine. *Symmetry* **2020**, *12*, 541.
8. Nemati, H.; Fan, Y.; Alonso-fernandez, F. Hand Detection and Gesture Recognition Using Symmetric Patterns. *Stud. Comput. Intell.* **2016**, *642*, 365–375.
9. Zhang, Q.; Feng, L.; Liang, H.; Yang, Y. Hybrid Domain Attention Network for Efficient Super-Resolution. *Symmetry* **2022**, *14*, 697. [CrossRef]
10. Gao, F.; Zhang, J.; Liu, Y.; Han, Y. Image Translation for Oracle Bone Character Interpretation. *Symmetry* **2022**, *14*, 743. [CrossRef]
11. Karbasi, M.; Zabidi, A.; Yassin, I.M.; Waqas, A.; Bhatti, Z.; Alam, S. Malaysian Sign Language Dataset for Automatic Sign Language Recognition System. *J. Fundam. Appl. Sci.* **2017**, *9*, 459–474. [CrossRef]

12. Pariwat, T.; Seresangtakul, P. Multi-Stroke Thai Finger-Spelling Sign Language Recognition System with Deep Learning. *Symmetry* **2021**, *13*, 262. [CrossRef]
13. Wen, R.; Tay, W.L.; Nguyen, B.P.; Chng, C.B.; Chui, C.K. Hand gesture guided robot-assisted surgery based on a direct augmented reality interface. *Comput. Methods Programs Biomed.* **2014**, *116*, 68–80. [CrossRef] [PubMed]
14. Nguyen, B.P.; Tay, W.L.; Chui, C.K. Robust Biometric Recognition from Palm Depth Images for Gloved Hands. *IEEE Trans. Hum.-Mach. Syst.* **2015**, *45*, 799–804. [CrossRef]
15. Oudah, M.; Al-Naji, A.; Chahl, J. Hand Gesture Recognition Based on Computer Vision: A Review of Techniques. *J. Imaging* **2020**, *6*, 73. [CrossRef] [PubMed]
16. Sagayam, K.M.; Hemanth, D.J. A probabilistic model for state sequence analysis in hidden Markov model for hand gesture recognition. *Comput. Intell.* **2019**, *35*, 59–81. [CrossRef]
17. Chang, C.C.; Lin, C.J. LIBSVM: A Library for support vector machines. *ACM Trans. Intell. Syst. Technol.* **2011**, *2*, 1–27. [CrossRef]
18. Wu, X.Y. A hand gesture recognition algorithm based on DC-CNN. *Multimed. Tools Appl.* **2020**, *79*, 9193–9205. [CrossRef]
19. Saqlain Shah, S.M.; Abbas Naqvi, H.; Khan, J.I.; Ramzan, M.; Zulqarnain; Khan, H.U. Shape based Pakistan sign language categorization using statistical features and support vector machines. *IEEE Access* **2018**, *6*, 59242–59252. [CrossRef]
20. Chen, G.; Xu, Z.; Li, Z.; Tang, H.; Qu, S.; Ren, K.; Knoll, A. A Novel Illumination-Robust Hand Gesture Recognition System with Event-Based Neuromorphic Vision Sensor. *IEEE Trans. Autom. Sci. Eng.* **2021**, *18*, 508–520. [CrossRef]
21. Mahmood, A.; Khan, S.A.; Hussain, S.; Almaghayreh, E.M. An Adaptive Image Contrast Enhancement Technique for Low-Contrast Images. *IEEE Access* **2019**, *7*, 161584–161593. [CrossRef]
22. Al Delail, B.; Bhaskar, H.; Zemerly, M.J.; Werghi, N. Balancing Incident and Ambient Light for Illumination Compensation in Video Applications. In Proceedings of the 2018 25th IEEE International Conference on Image Processing (ICIP), Athens, Greece, 7–10 October 2018; pp. 1762–1766. [CrossRef]
23. Yu, W.; Lei, B.; Ng, M.K.; Cheung, A.C.; Shen, Y.; Wang, S. Tensorizing GAN With High-Order Pooling for Alzheimer's Disease Assessment. *IEEE Trans. Neural Netw. Learn. Syst.* **2021**, *33*, 4945–4959. [CrossRef] [PubMed]
24. You, S.; Lei, B.; Wang, S.; Chui, C.K.; Cheung, A.C.; Liu, Y.; Gan, M.; Wu, G.; Shen, Y. Fine Perceptive GANs for Brain MR Image Super-Resolution in Wavelet Domain. *IEEE Trans. Neural Netw. Learn. Syst.* **2022**, 1–13. [CrossRef] [PubMed]
25. Gozzi, N.; Malandri, L.; Mercorio, F.; Pedrocchi, A. XAI for myo-controlled prosthesis: Explaining EMG data for hand gesture classification. *Knowl.-Based Syst.* **2022**, *240*, 108053. [CrossRef]
26. Wang, Y.; Ren, A.; Zhou, M.; Wang, W.; Yang, X. A novel detection and recognition method for continuous hand gesture using fmcw radar. *IEEE Access* **2020**, *8*, 167264–167275. [CrossRef]
27. Xi, L.; Chen, W.; Zhao, C.; Wu, X.; Wang, J. Image Enhancement for Remote Photoplethysmography in a Low-Light Environment. In Proceedings of the 15th IEEE International Conference on Automatic Face and Gesture Recognition, Buenos Aires, Argentina, 16–20 November 2020; pp. 761–764. [CrossRef]
28. Li, G.; Tang, H.; Sun, Y.; Kong, J.; Jiang, G.; Jiang, D.; Tao, B.; Xu, S.; Liu, H. Hand gesture recognition based on convolution neural network. *Clust. Comput.* **2019**, *22*, 2719–2729. [CrossRef]
29. Wang, J.; Liu, T.; Wang, X. Human hand gesture recognition with convolutional neural networks for K-12 double-teachers instruction mode classroom. *Infrared Phys. Technol.* **2020**, *111*, 103464. [CrossRef]
30. Al-Hammadi, M.; Muhammad, G.; Abdul, W.; Alsulaiman, M.; Bencherif, M.A.; Mekhtiche, M.A. Hand Gesture Recognition for Sign Language Using 3DCNN. *IEEE Access* **2020**, *8*, 79491–79509. [CrossRef]
31. Mujahid, A.; Awan, M.J.; Yasin, A.; Mohammed, M.A.; Damaševičius, R.; Maskeliūnas, R.; Abdulkareem, K.H. Real-time hand gesture recognition based on deep learning YOLOv3 model. *Appl. Sci.* **2021**, *11*, 4164. [CrossRef]
32. Al-Hammadi, M.; Muhammad, G.; Abdul, W.; Alsulaiman, M.; Bencherif, M.A.; Alrayes, T.S.; Mathkour, H.; Mekhtiche, M.A. Deep learning-based approach for sign language gesture recognition with efficient hand gesture representation. *IEEE Access* **2020**, *8*, 192527–192542. [CrossRef]
33. Tan, Y.S.; Lim, K.M.; Lee, C.P. Hand gesture recognition via enhanced densely connected convolutional neural network. *Expert Syst. Appl.* **2021**, *175*, 114797. [CrossRef]
34. Pinto, R.F.; Borges, C.D.B.; Almeida, A.M.A.; Paula, I.C. Static Hand Gesture Recognition Based on Convolutional Neural Networks. *J. Electr. Comput. Eng.* **2019**, *2019*, 4167890. [CrossRef]
35. Pin, X.; Chunrong, Z.; Gang, H.; Yu, Z. Object Intelligent Detection and Implementation Based on Neural Network and Deep Learning. In Proceedings of the 2020 International Conference on Computer Engineering and Application ICCEA 2020, Guangzhou, China, 18–20 March 2020; pp. 333–338. [CrossRef]
36. Lowe, D.G. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vis.* **2004**, *60*, 91–110. [CrossRef]
37. Rey-Otero, I.; Delbracio, M. Anatomy of the SIFT method. *Image Process. Line* **2014**, *4*, 370–396. [CrossRef]
38. Gopal, P.; Gesta, A.; Mohebbi, A. A Systematic Study on Electromyography-Based Hand Gesture Recognition for Assistive Robots Using Deep Learning and Machine Learning Models. *Sensors* **2022**, *22*, 3650. [CrossRef] [PubMed]