Deep Learning-based Deaf & Mute Gesture Translation System

Azher Atallah Fahad[1], Hassan Jaleel Hassan[2], Salma Hameedi Abdullah[3]

[1, 2, 3]Computer Engineering Department, University of Technology, Baghdad, Iraq

Abstract: ***Translation System (TS) is a method used by Deaf individuals to interact with other ordinary persons in the world. Since computers are an important component of our community, the progress of human-computer interaction (HCI) has supported disabled people. And the main purpose of our proposed system is to progress an intelligent system which can turn as a translator between normal and deaf or dumb individuals and can be the communication path between people with speaking deficiency and normal people with both effective and efficient ways. The proposed system consists of a Convolutional Neural Network (CNN) based on the deep learning algorithm for effective extraction of handy features to understand the American Sign Language (ASL), for classifying the hand sign. This paper constructs to interpret ASL and also provides a complete overview of deep learning-based methodologies for sign language recognition. The proposed solution was tested on data samples from ASL data sets and achieved an overall accuracy of 96.68%. The proposed system was appropriate and reliable for Deaf individuals. Furthermore, an efficient and low-cost Hand Gesture Recognition (HGR) system for the real-time video stream from a mobile device camera. A separate individual hand gesture is utilized for validation in this article. The proposed system has to be designed with the front of the camera and the output is given in the form of text or audio.***

Keywords: Human-computer-interaction (HCI), Hand Gesture Recognition (HGR), American Sign Language (ASL), Image Segmentation, Convolutional Neural Network (CNN)

1. Introduction

Only a few people recognize the meaning of the sign. Normally, deaf individuals are denied of ordinary contact with other common persons in the community. HCI is an interesting mechanism among individuals and devices. This is an interesting area of research that focuses on the formation and usage of computer technology and, in specific, integrated interactions among humans and devices. HCI infrastructure has been amazingly extended and improved by the technical transition[1]. On the grounds of successful usability, emerging technologies implement modern user interfaces such as Non-Touch, Gesture, and Speech recognition. It's a complicated and costly technology to achieve. Such recently implemented technologies are then incorporated as applying to specific implementations on the grounds of demands and cost-effectiveness. To order to address the difficulties, several researchers are trying to develop these interfaces in terms of performance, accessibility and robustness [2]. The optimal design can have many standard features such as simplicity, precision, scalability, and flexibility. Today, the human gesture is becoming a widespread HCI application, and the utilization of human gestures, which satisfies all these standards, is growing rapidly[3]. The HGR has many uses in various areas, such as video gaming, sign language recognition and augmented reality (SLR). Between these, an SLR is the best widely utilized method where voice communication is difficult. On this perspective, this paper suggests an effective method for the extraction of features via CNN.

CNN comprises of multiple fully connected conventional layers as a regular multilayer neural network [4]. The architecture of CNN is designed to manage 2D images adequately [5].Already, CNN has many parameters to train the computer effectively [6]. Lastly, SoftMax function is used to identify the sign language gesture. The purpose of this paper is to offer the extraction of features and classification technique by CNN. The goal is to provide an intelligent application with a high degree of accuracy (less computing time usage). Where a successful gesture recognition time ranges between 0.25 and 0.50 seconds.

2. Literature Review

Many kinds of research have also been conducted on the interpretation of human signs using a variety of icons. Sign recognition of the letters, though, is more difficult. Almost all researchers have developed human body-related methods and hand gestures to improve technology utilization. Kilioz et al. (2015) have implemented an innovative method for the recognition of dynamic hand gestures on the grounds of real-time HCI. They use a six-degree, location tracker to gather trajectory data and depict motions as an orderly series of directional motions in 2D. Only the motion trajectory of the hand (except for finger bending and orientation information) is assumed to describe the gestures. The results of the proposed method In respect of gesture identification and recognition efficiency (73 per cent accuracy) in the flow of motion [7]. Modanwal et al. (2019) eliminated the distance between the computer and the blind by implementing gesture recognition. They are utilizing tactile or touch as just a replacement to vision. For creating gestures, they also created a tabletop system. Audio input is also provided to the user via the earphone or microphone to ensure reliable and efficient data input. The suggested dactylology was developed on the basis of a definition close to the Brailler method used to implement the Braille code. The suggested dactylology still requires both hands, but the person has to place the fingers rather than depressing the buttons. For this experimental work, just finger-based gestures are investigated. A maximum of 31 gestures can be produced for each hand with the aid of five fingers. The sign recognition

Licensed Under Creative Commons Attribution CC BY

score of the suggested system was 97.53%. [8]. Josiane Uwineza et al. (2019) suggested a model manage Human-robot interaction. The suggested hand gesture recognition utilizing hybrid feature extraction approaches such as Hu Moments, color histogram and Haralick texture, plus Extreme Learning Machine (ELM) for classification. The precision of 98.7% was reached, which is stronger precision and shows that the ELM approach could be used in human-robot interactions.[9]. Haria et al. (2017) developed a less hand gesture recognition marker system that can detect both static and dynamic hand gestures. They utilized a webcam installed on a laptop without the use of extra cameras or hand markings like gloves. Their system translates the detected gesture into actions such as opening websites and launching applications like VLC Player and PowerPoint. Numerous approaches have been used for pre-processing the image, including algorithms and techniques for noise reduction, edge detection, smoothing, accompanied by specific segmentation techniques for boundary extraction, i.e. distinguishing the foreground from the background. They used a total of seven gestures in their gesture recognition system, six of which are static, while the seventh is a dynamic gesture. Contours, and convexity defects were used with a Haar cascade to identify the entity (hand). As applied against some clear background, the gesture recognition system was stable and operated with approximately 92.28% accuracy. For scenarios where the background was not clear, the accuracy was not strong at about 64.85 percent[10]. Simran et al., (2019), have designed system requires smooth connectivity between people and computers in the YouTube app. They introduced five gestures to manager various functionalities such as light, speed, and start or stop the app. They used the features of image processing, accompanied by neural networks, to categorize the defined gesture. The precision for the individual signing the signature is 96.03 percent[11] .

3. Proposed Model

The application is configured to obtain frames from the real-time video stream of camera that various techniques of image processing will be working on it. Next, the input frame should be transformed from RGB to Grayscale. To enhance the precision of the input gesture, the noise contained in the input frame should be eliminated. Besides, the hand segment is observed from the image, and the hand gesture is extracted from the frame taken. This processing, image, i.e. binary image mode, is then compared to the trained model. User Interface is structured to enable a consumer to catch a picture from a mobile device camera. Such recorded images are preserved in the input folder. Then, hand gesture images can be used to support the CNN training model. This dataset contains hand gesture alphabets and counting digits images. There are 2,000 pictures per class, i.e. for each letter and digit. After that, the images obtained are used to train the CNN model. In this, the images, which are transformed previously into binary mode, are fed to the CNN model for processing. For which Eighty percent of the data was provided for training and Twenty percent for the data given for testing. In the training output, the model generates a file of form h5 that stores a description of the training process. By the model, this file will be used for prediction of alphabets and digits. Eventually, the user input image is supplied to the CNN model for prediction, which compares the input images and the images recorded in the CNN model. Depending on a comparison, the CNN model generates output in text or audio format.

The basic principle of our model is proposed to classify the ASL alphabet, based on the human hand gesture. The operating method for the suggested model is seen in Figure1.

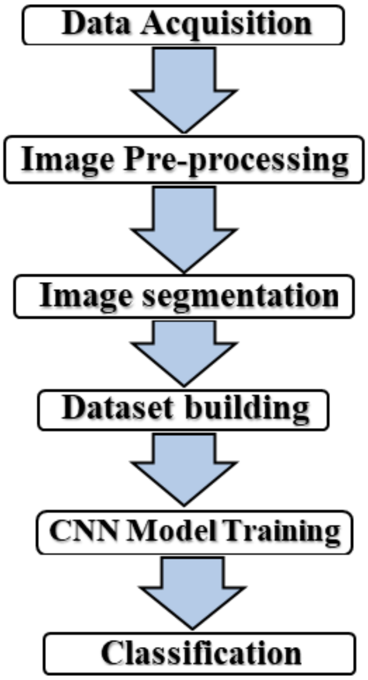

Figure 1: Operational process of the suggested model

3.1 Data Acquisition

First of all, take a hand gesture video in real-time from a handheld computer, i.e. Tablet, and get the video frames for further calculations.

3.2 Pre-processing stage

At this point. The region of interest (ROI) is separated from a frame that is identified on a border basis and is defined by a single frame marker. The value of the grayscale intensity is specified and used to divide the region into foreground and background areas depending on the pixel intensity. Intensity values of 0 (or background) or 1 (for foreground) can be assigned to the pixels in the frame. When it categorizes the region as a hand or background, disregards the unused remainder of the video frame, and resizes the frame to a specific resolution. The region of interest as shown in Figure 2. Afterwards.

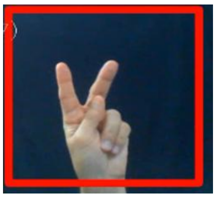

Figure 2: ROI

Licensed Under Creative Commons Attribution CC BY

3.3 Segmentation and primary feature extraction stage

We have one of two modes of pre-processing the captured images. Binary Mode is used for the plain background to convert the image to grayscale, where Outso approach is used to convert the gray images to binary images where the value of either 0 (for background) or 1 (for foreground). Whereas background subtraction is used for the compound background to get a subset of an image which it calculates the mathematics of the front mask between the frame buffer and the background image, that is, the stationary portion of the sequence or, more generally, anything that can be defined as the background according to the attributes of the scene being studied[12].In each of these, further noise removal techniques like Gaussian blur, Erosion are applied.
Morphological filtering is important to apply morphological filtering to segmented images to create a cleaner, more closed and more contoured gesture. This is accomplished by a series of erosion operations over the rotation of the invariant segmented gesture image.

3.4 Customized Dataset

HGR screenshots for 26 letter signs are obtained for ASL, Ten Counting digits, and three unique characters for three distinct people. There are 2,000 images for each sign and human. There are 3x2000 x images (26 alphabets + ten numbers + three characters).

Afterwards, frames are processed as images in folders. The folder name is being used to mark the images according to the category of gestures in the frame (i.e. mark A for alphabet A, and so on for certain categories of gestures). Any representation of the sign language can be used, the interpretation proposed is for the American Sign Language Alphabet, see Figure 3.

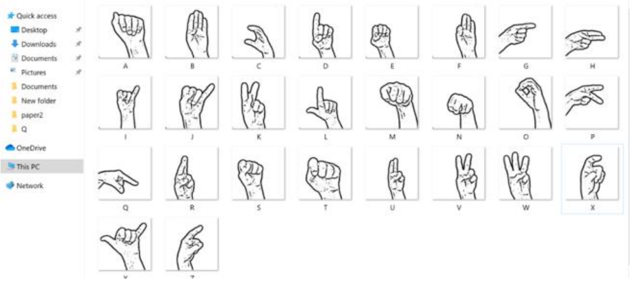

Figure 3: Alphabet ASL dataset Images

3.5 Feature selection &Training stage

At this point, the extracted features are chosen for classification. For this suggested model of feature extraction, the vector element is derived from the frame of a video sequence utilizing the CNN. Many of the extracted features of the image are kept after extraction in the file. Powerful algorithms of machine learning are used to extract features. One of the strongest deep learning strategies is CNN. A broad of different images varies that CNN is used. CNN will collect possible features for the classification model across a wide variety of pictures. The proposed network is four hidden layers. The input dimension is specified for the first layer. In this paper, the resolution of the representation of the data is 300x300x1. See Figure 4.

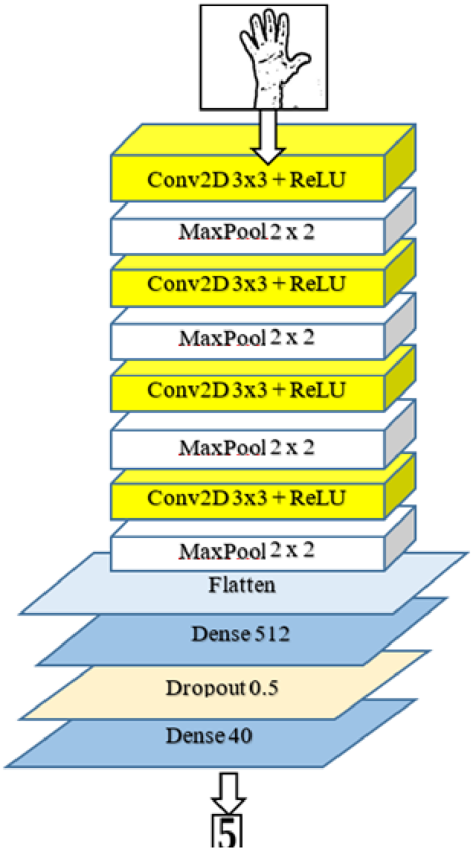

Figure 4: Convolutional Neural Network architecture

CNN generates an activation form, which is used as an input image for every layer. The process continues layer after layer. Indeed, there are just several layers appropriate for the extraction of image features throughout CNN. Inside this suggested model, hidden layers are assumed as the extraction of features. With these layers, the simple image features of the layers are obtained at the network started. Deeper layers of the network process these essential features and merge the preliminary feature to construct image features of the higher level. These are all features at a higher level are well-designed for categorization activities. Since deeper layers of the network incorporate all these features of rudimentary into a better image exemplification.

3.6 Classification stage

Finally, the SoftMax function is used to quantify every alphabetical sign during the final part of this suggested model. That is a generalization of logistic regression insofar as it can be implemented to continuous data (rather than binary classification) which can include many decision boundaries. It deals with multinomial labeling mechanisms. Softmax is the function that we always consider the output layer of the classifier. The softmax activation function provides the distribution of the likelihood between mutually exclusive output classes. SoftMax is commonly utilized as a tool of learning for the description of derived regression and features. SoftMax is a classifier based on a supervised learning method for categorizing data in several classes. The central operating method of SoftMax is to assign the sample

Licensed Under Creative Commons Attribution CC BY

data entered into several separated classes. In SoftMax, one class is distinguished from the other and a final decision is made for this procedure by choosing the maximum output SoftMax value.

4. Result and Estimation

Python with the Keras and TensorFlow backend libraries was used to apply the suggested deep learning algorithm. The suggested model is evaluated through an interconnected data collection composed of 40 symbols of three distinct individuals. Each symbol consists of 2000 images of each individual. So, in all, there are 3 by 2000 by 40 images. The dataset is split into two groups. The first branched collection comprises 80 percent for training images and the second one includes the remainder of the 20 percent for testing images. Figure 5 displays the signs of the ASL alphabet.

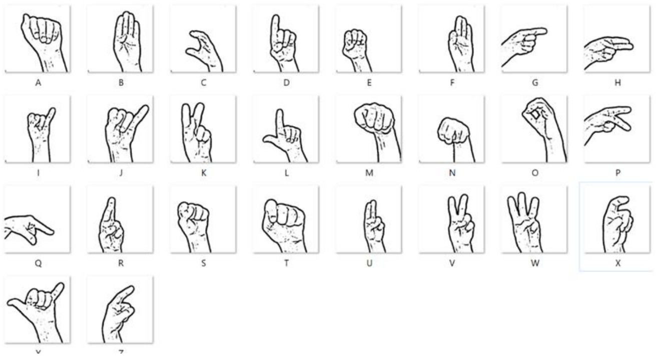

Figure 5: Signs in the ASL language

For the extraction of a feature that CNN is used. After using CNN for the feature extraction of the images, then, we consider the number of training features and the number of testing features. Those are all features are helpful, that is contributing to distinguish the classes in each individual sign. See figure 6 that show model accuracy about 98 % in an average.

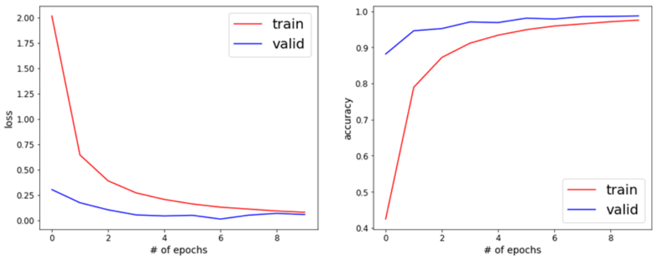

Figure 6: Graph of loss & accuracy against number of epochs for proposed model.

With these more detailed features of training and testing, each of the signs individually performed by the SoftMax has been identified. The quality of the classification has been accepted, which is 96.68%. The accuracy of the person sign is seen in Table 1 & 2 under varying luminous intensity impact.

Table 1: Accuracy under the influence of light intensity 900 LUX

Under the influence of light intensity 900 LUX				
Gesture	*Samples*	*Correct*	*Success Rate*	*Accuracy Score Rate*
A	20	20	100%	99.99 %
B	20	20	100%	99.99 %
C	20	20	100%	100.00 %
D	20	20	100%	98.93 %
E	20	20	100%	98.95 %
F	20	20	100%	92.90 %
G	20	20	100%	99.99 %
H	20	20	100%	99.99 %
I	20	20	100%	99.99 %
J	20	20	100%	99.99 %
K	20	20	100%	99.99 %
L	20	20	100%	99.99 %
M	20	17	85%	87.99 %
N	20	18	90%	89.99 %
O	20	19	95%	90.09 %
P	20	19	95%	98.01 %
Q	20	20	100%	99.99 %
R	20	20	100%	99.99 %
S	20	20	100%	99.99 %
T	20	20	100%	99.99 %
U	20	20	100%	99.12 %
V	20	19	95%	91.00 %
W	20	20	100%	98.29 %
X	20	20	100%	99.96 %
Y	20	20	100%	99.99 %
Z	20	20	100%	99.99 %
Average Rate			98%	97.89%

Table 2: Accuracy under the influence of light intensity 150 LUX

Under the influence of light intensity 150 LUX				
Gesture	Samples	Correct	Success Rate	Accuracy Score Rate
A	20	20	100%	99.97 %
B	20	20	100%	99.93 %
C	20	20	100%	100.00 %
D	20	20	100%	98.93 %
E	20	20	100%	96.15 %
F	20	20	100%	88.95 %
G	20	20	100%	96.07 %
H	20	20	100%	97.33 %
I	20	20	100%	98.25 %
J	20	20	100%	98.95 %
K	20	20	100%	98.11 %
L	20	20	100%	97.99 %
M	20	17	85%	65.49 %
N	20	18	90%	71.16 %
O	20	19	95%	85.36 %
P	20	19	95%	95.66 %
Q	20	20	100%	98.87 %
R	20	20	100%	98.99 %
S	20	20	100%	97.89 %
T	20	20	100%	97.77 %
U	20	19	95%	98.12 %
V	20	20	100%	90.06 %
W	20	20	100%	96.10 %
X	20	20	100%	92.99 %
Y	20	20	100%	98.03 %
Z	20	20	100%	99.07 %
Average Rate			98%	94.47%

The average results of our suggested model are seen in the Table 3.

Table 3: Average accuracy

Average Rate at LUX = 900	*Average Rate at LUX = 150*	*Total average*
98.89%	94.47%	96.68%

Licensed Under Creative Commons Attribution CC BY

Table 4: Comparing the state of the art with the suggested approach

Author	Method	Accuracy	Static/dynamic gestures recognition	No. of recognized gestures	Processing time
Kilioz et al. (2015) [7]	A six-degree position sensor for gathering trajectory data and representing motions as an ordered sequence of spatial motions in 2D.	73%	both	10	1.50 sec
Haria et al.(2017) [10]	Background subtraction to find contours, and convexity defects were used with a Haar cascade to identify hand	92.28%	both	7	Not reported
Modanwal et al. (2019) [8]	dactylology based on a concept similar to the Brailler concept	97.53 %	static	31 gestures of each hand	Not reported
Josiane Uwineza et al. (2019) [9]	Hu Times, the structure of Haralick, and the color histogram for the extraction of features. Extreme Learning Machine for Categorization (ELM)	98.7%	static	9	109.7 s
(Simran et al. (2019)[11]	Image processing features(Image Preprocessing, Segmentation, Gaussian blur, Morphological filtering), followed by CNN for classification	Val acc:98.98%, Test Acc:96.03 %	static	5	Not reported
Our Proposed System	threshold process, background subtraction to detect object and CNN for feature extraction and classification	Val acc:98.11%, Test Acc:96.68	static	40	0.50 sec.

5. Conclusion

Hand gesture recognition is a big problem in real-life implementations concerning the precision and reliability correlated with it. This paper introduces the hand gesture identification in ASL without touching, the input gestures are recorded utilizing a mobile device camera. A still-hand shot taken from a real-time video stream frame and using CNN to search for more insightful features. Eventually, recognizing the sign of the alphabet by SoftMax. About the validity of the model that is proposed, our designed dataset is used in compliance with the ASL conventions. Classification accuracy was attained by 96.68%, which is notable with the implementation of ASL Sign Language Recognition with Disabled Persons as the production of HCI. The gesture has to be formed in front of the camera and the output is given in the form of text or audio.

References

[1] J. F. and H. H. Jonathan Lazar, *Research Methods in Human Computer Interaction*. Morgan Kaufmann, 2017.

[2] A. Mantri and M. Ingle, *A Comparative Study of Various Techniques Used in Current HGRSs*. Springer Singapore, 2018.

[3] R. A. Bhuiyan, A. K. Tushar, A. Ashiquzzaman, J. Shin, and R. Islam, “Reduction of Gesture Feature Dimension for Improving the Hand Gesture Recognition Performance of Numerical Sign Language,” pp. 22–24, 2017.

[4] B. A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet Classification with Deep Convolutional Neural Networks,” 2012.

[5] A. Z. Karen Simonyan, “VERY DEEP CONVOLUTIONAL NETWORKS FOR LARGE-SCALE IMAGE RECOGNITION,” pp. 1–14, 2015.

[6] J. Donahue *et al.*, “DeCAF : A Deep Convolutional Activation Feature for Generic Visual Recognition,” vol. 32, 2014.

[7] U. G. Nurettin Çagˇrı Kılıboz, “A hand gesture recognition technique for human–computer interaction,” vol. 28, pp. 97–104, 2015, doi: 10.1016/j.jvcir.2015.01.015.

[8] G. Modanwal and K. Sarawadekar, “Utilizing gestures to enable visually impaired for computer interaction,” *CSI Trans. ICT*, no. June, 2019, doi: 10.1007/s40012-019-00251-w. and Y. J. , Hongbin Ma, Baokui Li, *Static Hand Gesture Recognition for Human Robot Interaction*. Springer International Publishing, 2019.

[9] A. Haria, A. Subramanian, N. Asokkumar, S. Poddar, and J. S. Nayak, “Hand Gesture Recognition for Human Computer Interaction,” *Procedia Comput. Sci.*, vol. 115, pp. 367–374, 2017, doi: 10.1016/j.procs.2017.09.092.

[10] P. P. M. C. Simran Shah, Ami Kotia, Kausha Nisar, Aneri Udeshi, “A Vision Based Hand Gesture Recognition System using Convolutional Neural Networks,” *Int. Res. J. Eng. Technol.*, vol. 463, no. 6, pp. 2570–2575, 2019, doi: 10.1007/978-981-10-6571-2_132.

[11] N. Umadevi and I. R. Divyasree, “Development of an Efficient Hand Gesture Recognition system for human computer interaction,” no. September, 2018, doi: 10.18535/Ijecs/v4i12.5.