

Sign Language Recognition using Depth Data and CNN

Lakshman Karthik Ramkumar^{#1}, Sudharsana Premchand^{*2}, Gokul Karthi Vijayakumar^{#3}

[#]Undergraduate, Department of Computer Science and Engineering, Dr. Mahalingam College of Engineering and Technology, Pollachi, India

Abstract

Sign Language Recognition is a project which is primarily focused on recognizing gestures (used by Deaf and Dumb) and process them into sentences, thereby making communication of such people easier with the common people. The project uses "Intel Real Sense" technology (Intel Real Sense Robotic Development Kit), which captures depth images of gestures made the user. Depth images are useful for making highly precise predictions on what is being communicated by the user. Early works used different kind of image convolution to form feature vectors based on a single RGB image of a hand. The authors use wavelet families, computed on edge images, as features to train a Neural Network for 24 sign classification. Haarlet-like features, computed on grey-scale images and on silhouettes were used for classification of 10 hand shapes. Principle Component Analysis (PCA) was applied directly on images to derive a subspace of hand poses, which is then used to classify the hand poses. A modification of HOG descriptors is employed to recognize static signs of the British Sign Language. SIFT-feature based description was used to recognize signs of ASL. All these methods depend heavily on the lighting conditions, subject's appearance and Background. This project uses Convolutional Neural Networks for classifying gestures using depth data and its features for more accuracy

Keywords - ASL, Recognition, Depth Data, RealSense, Gesture Recognition.

I. INTRODUCTION

There are a lot of people who are unable to hear and speak, which makes it harder for them to carry out a normal life. Sign Language acts as a tool for such speech-impaired people to effectively communicate with others. Sign Language comprises of several gestures that makes it easier for them to carry out a conversation. Unfortunately, there are only few populations who are aware and know sign language. This in turn makes the life of speech-impaired people harder as they can't communicate effectively.

Sign Language Recognition is a project that recognizes the gestures made and parse them into appropriate sentences. It's primarily focused on real-time gesture recognition, which can be deployed in

public places where communication is necessary (E.g. Hospitals, Teaching methodologies, News reading, etc.,).

A mundane life of speech-impaired person is so hard that he/she can't do anything they would like as they could not communicate with common people. They become more dependent on people who translate for them. It would be easy if there is a Digital Assistant which fills the gap between the two by translating the gestures into sentences, which can be later transformed into voice to make the communication possible. Sign Language Project addresses this gap and acts as a bridge between speech-impaired and common people. The system can also be integrated with chat-bots and other voice assistants, so that they could use them without any hurdles.

The study focuses on developing a digital assistant that recognizes the gestures (Sign Language) and translate them to proper sentences, which can be then used for enabling proper communication between speech-impaired and common people.

Speech-impaired people feel difficult to communicate with common people through Sign Language in their regular routine. They understand only those gestures, which in turn makes it difficult for them to communicate effectively. The gestures vary between different countries and regions. A system that facilitates people to interpret the gestures irrespective of the region is to be built

A. Objective

- To develop a system that recognizes the gestures and translate them into structured sentences.
- The system is useful for bridging the communication gap between speech-impaired people and common people.
- To help speech-impaired people effectively converse with speech assistants like Alexa, Google assistant.
- To help common people to understand sign language, hence making a hassle-free communication.

B. Overview

Chapter 2 presents a brief overview of the existing literature on this topic. CNN approaches to

classify gestures and feature extraction from raw image using convex hull and contours are discussed. Chapter 3 describes the methodology used to recognize real-time sign language gestures. This includes pre-processing the input data to a form more suitable for training Neural Networks, and ways to initialize weights effectively using unsupervised learning methods such as max-pooling. An overview of the experiments conducted on different lighting conditions are also described. Chapter 4 presents the results of the experiments. The performance of the Networks using various architectures and weight initialization methods is tabulated. Chapter 5 describes the interpretation of the results and gives a formal conclusion to the project.

II. METHODOLOGY

The communication between the user and the system occurs as follows:

- The user makes the gestures in-front of the camera which captures the Depth Image and separates the gestures from the entire scene.
- The important features such as position of hand and fingers are extracted to make decision on what is being gestured.
- The Gesture Recognition is performed using Convolutional Neural Networks, by classifying the gestures.
- The sentences are interpreted from the gesture and finally sent to the user interface which displays/speaks the sentence.

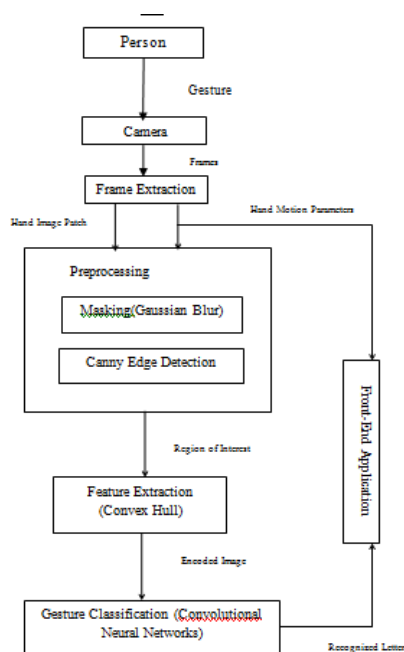


Fig. 1. Block Diagram for Sign Language Recognition


















A. Pre-processing of input data

Each frame of image is pre-processed by cropping it by 50x50 pixels and then Gaussian blur is applied for smoothing the image frames. The gaussian

blurred images are then further smoothed using median blur function to obtain a black and white image. These frames are then processed for obtaining the contours, which can be used to extract the features.

TABLE I. GESTURES AND THEIR LABELS

GESTURE ID	PHRASE	DATA SET
0	A	
1	B	
2	C	
3	D	
4	E	
5	F	
6	G	
7	H	
8	I	
9	J	
10	K	
11	L	
12	M	
13	N	

14	O	
15	P	
16	Q	
17	R	
18	S	
19	T	
20	U	
21	V	
22	W	
23	X	
24	Y	
25	Z	
26	0	
27	1	
28	2	
29	3	
30	4	














31	5	
32	6	
33	7	
34	8	
35	9	
36	Best of Luck	
37	You	
38	I/Me	
39	Like	
40	Remember	
41	Love	
42	Hello	
43	I love you	

Table I, shows a list of words and their labels for classification that will be used for predicting the gestures. The words will be constructed based upon the labels listed in the above table which is stored in an SQLite database.

B. Gaussian Blur

To perform a smoothing operation, we will apply a filter to our image. The most common type of filters is linear, in which an output pixel’s value (i.e. $g(i, j)$) is determined as a weighted sum of input pixel values (i.e. $f(i + k, j + l)$):

$$g(i, j) = \sum_{k,l} f(i + k, j + l)h(k, l) \tag{3.1}$$

$h(k, l)$ is called the kernel, which is nothing more than the coefficients of the filter.

It helps to visualize a filter as a window of coefficients sliding across the image. Gaussian filtering is done by convolving each point in the input array with a Gaussian kernel and then summing them all to produce the output array.

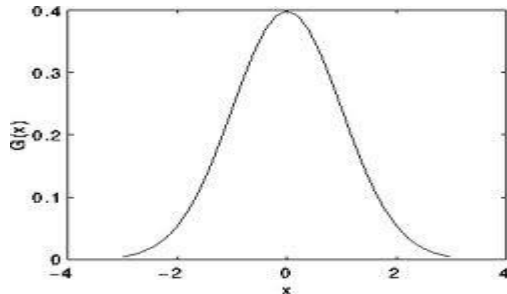


Fig. 2. Example of 1D Gaussian Curve

A 2D Gaussian can be represented as:

$$G_0(x, y) = Ae^{-\frac{(x - \mu_x)^2}{2\sigma_x^2} - \frac{(y - \mu_y)^2}{2\sigma_y^2}} \quad (3.2)$$

where μ is the mean (the peak) and σ represents the standard deviation (per each of the variables x and y)

C. Architecture of Convolutional Neural Network

Convolutional Neural Networks (also referred to as CNN or ConvNet) are a class of deep Neural Networks that have seen widespread adoption in a number of computer vision and visual imagery applications. The overall architecture of a CNN consists of an input layer, hidden layer(s), and an output layer. They are several types of layers, for e.g. Convolutional, Activation, Pooling, Dropout, Dense, and SoftMax layer. Fig.3 represents a generic architecture of CNN.

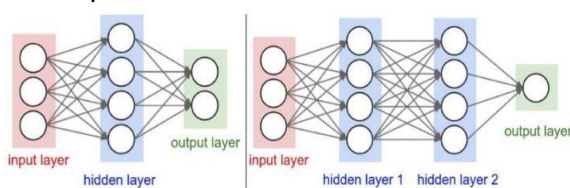


Fig. 3. CNN Architecture

D. Gradient Descent Algorithm

The gradient descent algorithm, along with the backpropagation technique, is used to optimize the weights of the Neural Network. During the forward pass, the Network uses the weights to predict the output. The “cost” or error value i.e the difference between the actual and predicted output is backpropagated through the Network and the gradients are used to update the weight matrices. During each step, the gradient descent algorithm takes

a small step in the direction which has the lowest slope. This is repeated several times until the global minimum is reached, and thus, the Network is optimized. Equation 1 and 2 show the computations to get the gradients for each layer, from right to left. $\delta(l)$ denotes the error values of nodes in layer l . $\Theta(l)$ denotes the weight matrix from layer l to layer $l+1$. g is the activation function, $z(l)$ denotes the input values to layer l , and $a(l)$ is the activation at layer l .

$$\delta(l) = (\Theta(l))^T \delta^{(l+1)} * g'(z(l)) \quad (3.3)$$

$$g'(z(l)) = a(l) * (1-a(l)) \quad (3.4)$$

Figure 4 shows the visualization of cost where the global minimum is at the centre. $J(w)$ denotes the cost for the weights w . The steps taken towards reaching the minimum are highlighted in black.

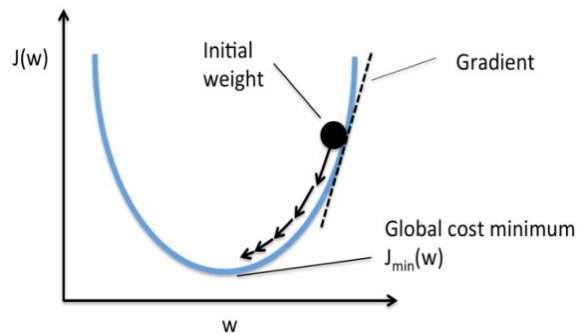


Fig. 4. Illustration of Gradient Descent

E. Activation Functions

The Soft Max function squashes the outputs of each unit to be between 0 and 1, just like a sigmoid function. But it also divides each output such that the total sum of the outputs is equal to 1 (check it on the figure above).

The output of the Soft Max function is equivalent to a categorical probability distribution, it tells you the probability that any of the classes are true.

Mathematically the Soft Max function is shown below, where z is a vector of the inputs to the output layer (if you have 10 output units, then there are 10 elements in z). And again, j indexes the output units, so $j = 1, 2, \dots, K$.

$$\sigma(z)_j = \frac{e^{z_j}}{\sum_{k=1}^K e^{z_k}} \quad (3.5)$$

F. Hardware and Software Requirements

HARDWARE	
----------	--

REQUIREMENTS	
Hardware	Purpose
Intel RealSense R200 3D depth image sensor	Capturing Depth & RGB Images.
Single Board Computer with 4GB RAM, 8GB ROM, Quad-Core CPU	Processing Depth Images and Prediction of gestures.
SOFTWARE REQUIREMENTS	
Software	Purpose
Python (V. 3.6)	Programming Language used for developing the system.
OpenCV	Open-source Image processing library used for Image Pre-processing.
TensorFlow	Open-source Machine Learning Library for Classification of gestures.
Keras	Open-source Deep Learning platform for improved accuracy of models generated by TensorFlow
Realsense SDK 1.0	An API for interacting with the Intel Realsense R200 Depth camera.

III.RESULTS

A. Dataset

The CNN is tested with the dataset that has been generated. A total of 1,03,000 datasets comprising 2400 samples for 43 gestures is used for classification and training. The Dataset is derived from a Finger Spelling Library, which is an open-source project for collection of Sign Language Datasets.

(src:<http://empslocal.ex.ac.uk/people/staff/np331/index.php?section=FingerSpellingDataset>)

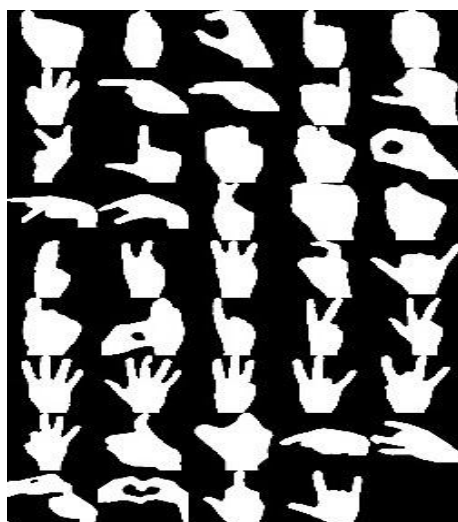


Fig. 5. Sample Dataset for Sign Language

B. Evaluation Metric

The Networks predict the labels and their corresponding confidence/support values are used for evaluation. A minimum threshold of 360 is used for determining the accuracy of the prediction.

C. Experiments and Results

In our experiment, a confusion matrix has been used to analyse the accuracy of each gesture and 30% of dataset sample is used for this purpose.

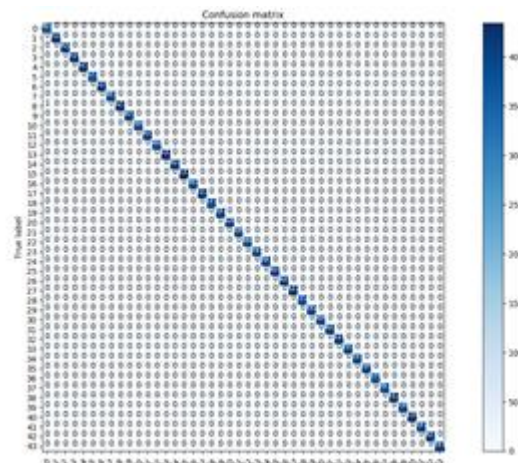


Fig. 6. Confusion Matrix for Gesture classification

D. Summary of Results

From the above table, we can observe that the accuracy for the alphabets J & Z is less compared to other alphabets, as they involved motion parameters such as swing and need more time to be recognized. The test accuracies are tested by F1 scores for which precision and recall values are considered (p is the number of correct positive results divided by the number of all positive results returned by the classifier, and r is the number of correct positive results divided by the number of all relevant samples). Though, CNN proves to be a faster and effective classifier in terms of speed and accuracy. Almost every gesture with support values greater than 370 is predicted correctly.

any professional title (e.g. Managing Director), any academic title (e.g. Dr.) or any membership of any professional organization (e.g. Senior Member SSRG International Journals).the family name must be written as the last part of each author name (e.g. John A.K. Smith). Every word in a heading must be capitalized except for short minor words as listed in Section III-B.heading “III. Page Style” of this document. The two level-1 headings which must not be numbered are “Acknowledgment” and “References”. (Size 10 & Bold & Italic) indented, in Italic and numbered with an Arabic numeral followed by a right parenthesis. The level-3 heading must end with a colon. The body of the level-3 section immediately follows the level-3 heading in the same paragraph. For example, this paragraph begins with a level-3 heading.SOLID FILL colors which contrast well both on screen and on a black-and-white

hardcopy, as shown in Fig. 1. Please check all figures in your paper both on screen and on a black-and-white hardcopy. When you check your paper on a black-and-white hardcopy, please ensure that: Figures must be numbered using Arabic numerals. Figure captions must be in 8 pt Regular font. Captions of a single line (e.g. Fig. 2) must be centered whereas multi-line captions must be justified (e.g. Fig. 1). Captions with figure numbers must be placed after their associated figures, as shown in Fig. 1.

IV. CONCLUSION

Thus, CNN can be used to predict the gestures. The CNN predicts the gestures accurately under better lighting conditions. However, it does not work well under varying lighting conditions, which let us calibrate the histogram every time. The training dataset performed well during prediction with a probability of 1. Anyhow, this is not the case in real-time environment.

In future, more gestures with Depth-trained data models will be added and the accuracy of the system will be improved.

ACKNOWLEDGMENT

We express our sincere thanks to our honourable Secretary Dr.C.Ramaswamy for providing us with required amenities.

We sincerely thank our director Dr. RangaPalaniswamy, for his moral support and encouragement for our project.

We wish to express our hearty thanks to Dr.A. Rathinavelu, Principal of our college, for his constant motivation and continual encouragement regarding our project work.

We are grateful to Dr. G. Anupriya, Head of the Department, Computer Science and Engineering, for her direction delivered at all times required. We also thank her for her tireless and meticulous efforts in bringing out this project to its logical conclusion.

Our hearty thanks to our guide Ms. A. Brunda, AP(SS) for her constant support and guidance offered to us during our project by being one among us and all the noble hearts that gave us immense encouragement towards the completion of our project.

REFERENCES

- [1] Lopez, S. Rio, J.M. Benitez and F. Herrera, "Real-time sign language recognition using a consumer depth camera", IEEE International Conference on Computer Vision Workshops, 2013
- [2] Ankit Chaudhary, J.L. Raheja, "Light invariant real-time robust hand gesture recognition", IEEE, 2016
- [3] Fabio M. Caputo, Pietro Prebianca, Andrea Carcangiu, Lucio D. Spano, Andrea Giachetti, "Comparing 3D trajectories for simple mid-air gesture recognition", Published on Science Direct in December, 2017

- [4] Aashni Haria, Archanasri Subramanian, Nivedhitha Asokkumar, Shristi Poddar, Jyothi S Nayak, "Hand Gesture Recognition for Human Computer Interaction", 7th International Conference on Advances in Computing & Communications, ICACC-2017, Cochin, India, 2017
- [5] Chen, Stanford University, "Sign Language Recognition with Unsupervised Feature Learning", IEEE, 2016
- [6] Anup Kumar, Karun Thankachan and Mevin M. Dominic, "Sign Language Recognition", IEEE, 2016
- [7] Brandon Garcia, Sigberto Alarcon Viesca, "Real-time American Sign Language Recognition with Convolutional Neural Networks", IEEE, 2016
- [8] Lionel Pigou, Sander Dieleman, Pieter-Jan Kindermans, Benjamin Schrauwen, "Sign Language Recognition using Convolutional Neural Networks", IEEE, 2015
- [9] Purva C. Badhe, Vaishali Kulkarni, "Indian Sign Language Translator Using Gesture Recognition Algorithm", Computer Graphics, Vision and Information Security (CGVIS), 2015 IEEE International Conference, 2015
- [10] Nagendraswamy H S, Chethana Kumara B M, Lekha Chinmayi R, "Indian Sign Language Recognition: An Approach Based on Fuzzy-Symbolic Data", IEEE
- [11] Guillaume Plouffe and Ana-Maria Cretu, Member, IEEE, "Static and Dynamic Hand Gesture Recognition in Depth Data Using Dynamic Time Warping", Published in International Conference on Advances in Computing, Communications and Informatics, 2016
- [12] Pichao Wang, Wanqing Li, Song Liu, Zhimin Gao, Chang Tang, Philip Ogunbona, "Large-scale Isolated Gesture Recognition Using Convolutional Neural Networks", 23rd International Conference on Pattern Recognition (ICPR), IEEE, 2016
- [13] Chenyang Zhang, Yingli Tian, Matt Huenerfauth, "Multi-Modality American Sign Language Recognition", Published in IEEE International Conference on Image Processing (ICIP), 2016
- [14] Lihong Zheng, Bin Liang, "Sign Language Recognition using Depth Images", Published in 14th International Conference on Control, Automation, Robotics and Vision (ICARCV), 2016
- [15] M. Mahadeva Prasad, "Gradient Feature based Static Sign Language Recognition", Published in IJCSE, Vol.6, Issue-12, December, 2018.