VISVESVARAYA TECHNOLOGICAL UNIVERSITY

JNANA SANGAMA, BELAGAVI – 590 018

A Project Phase 2 Report on

SIGN LANGUAGE TO TEXT TRANSLATIO IN REAL TIME

Submitted in partial fulfillment of the requirements for the VII Semester of degree of ***Bachelor of Engineering in Electronics and Communication Engineering*** *of Visvesvaraya Technological University, Belagavi*

Submitted by

Akshay Kumar Rai 1RN16EC014

Sanith Nair 1RN16EC124

Kundan 1RN16EC057

Under the Guidance of

Mr. Prakash Tunga
Assistant Professor
Department of ECE

ESTD:2001
An Institute with a Difference

Department of Electronics and Communication Engineering

RNS Institute of Technology

Dr. Vishnuvaradhana Road, Rajarajeshwari Nagar post, Channasandra, Bengaluru-560098

2019-2020

RNS INSTITUTE OF TECHNOLOGY

Dr. Vishnuvaradhana Road, RajarajeshwariNagar post,
Channasandra, Bengaluru - 560098

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

CERTIFICATE

Certified that the project work phase-II entitled ***SIGN LANGUAGE TO TEXT TRANSLATION IN REAL TIME*** has been successfully completed by **Akshay Kumar Rai (1RN16EC014), Sanith Nair (1RN16EC124) and Kundan (1RN16EC057),** bona fide students of **RNS Institute of Technology, Bengaluru** in partial fulfillment of the requirements for the award of degree in **Bachelor of Engineering in Electronics and Communication Engineering** of **Visvesvaraya Technological University, Belgaum** during academic year **2019-2020**. The project phase2 report has been approved as it satisfies the academic requirements in respect of project phase-II work for the said degree.

____________________	____________________	____________________
Mr. Prakash Tunga Project Guide Assistant Professor Department of ECE	**Dr. Vipula Singh** Professor and HoD Department of ECE RNSIT	**Dr. M K Venkatesha** Principal RNSIT

External Viva

Name of the Examiners	**Signature with Date**
1. ____________________________	**1.** ______________________
2. ____________________________	**2.** ______________________

DECLARATION

We, **AKSHAY KUMAR RAI [USN:1RN16EC014], SANITH NAIR [USN:1RN16EC124], KUNDAN [USN: 1RN16EC057]** students of VII Semester BE, in ***Electronics and Communication Engineering***, RNS Institute of Technology hereby declare that the Project phase-2 work entitled ***Sign Language Translation to Text in Real Time*** has been carried out by us and submitted in partial fulfillment of the requirements for the VII Semester degree of Bachelor of Engineering in Electronics and Communication Engineering of Visvesvaraya Technological University, Belgaum during academic year 2019-2020.

Place : Bengaluru

Date :

AKSHAY KUMAR RAI	**1RN16EC014**
SANITH NAIR	**1RN16EC124**
KUNDAN	**1RN16EC057**

ABSTRACT

Communication is the process of exchanging information, views and expressions between two or more persons, in both verbal and non-verbal manner. Hand gestures are the non verbal method of communication used along with verbal communication. A more organized form of hand gesture communication is known as sign language. In this language each alphabet of the English vocabulary is assigned a sign. The physically disabled person like the deaf and the dumb uses this language to communicate with each other. The idea of this project is to design a system that can understand the sign language accurately so that the less fortunate people may communicate with the outside world without the need of an interpreter. By keeping in mind the fact that in normal cases every human being has the same hand shape with four fingers and one thumb, this project aims at designing a real time system for the recognition of some meaningful shapes made using hands.

A gesture may be defined as a movement, usually of hand or face that expresses an idea, sentiment or emotion e.g. rising of eyebrows, shrugging of shoulders are some of the gestures we use in our day to day life. Sign language is a more organized and defined way of communication in which every word or alphabet is assigned some gesture. In American Sign Language (ASL) each alphabet of English vocabulary, A-Z, is assigned a unique gesture. Sign language is mostly used by the deaf, dumb or people with any other kind of disabilities. With the rapid advancements in technology, the use of computers in our daily life has increased manifolds. Our aim is to design a Human Computer Interface (HCI) system that can understand the sign language accurately so that the signing people may communicate with the non signing people without the need of an interpreter. It can be used to generate speech or text.

ACKNOWLEDGEMENT

The fulfilment and rapture that go with the fruitful finishing of any assignment would be inadequate without the specifying the people who made it conceivable, whose steady direction and support delegated the endeavors with success.

We would like to profoundly thank the Management of **RNS Institute of Technology** for providing such a healthy environment to carry out this phase 2 project work.

We would like to thank our beloved Director **Dr. H N Shivashankar** for his confidence feeling words and support for providing facilities throughout the course.

We would like to express our thanks to our Principal **Dr. M K Venkatesha** for his support and inspired us towards the attainment of knowledge.

We wish to place on record our words of gratitude to **Dr. Vipula Singh**, Professor and Head of the Department, Electronics and Communication Engineering, for being the enzyme and master mind behind our phase 1 project work.

We would like to express our profound and cordial gratitude to our Project Coordinator **Mr. Prakash Tunga**, Assistant Professor, Department of Electronics and Communication Engineering for his valuable guidance, constructive comments and continuous encouragement throughout the Project work, and for his valuable guidance in preparing phase 2 project report.

We would like to express our profound and cordial gratitude to our project reviewer **Mrs. Vinaya Kusuma**, Assistant Professor, Department of Electronics & Communication Engineering for her valuable guidance in preparing phase 2 project report.

And lastly, we would hereby acknowledge and thank our parents who have been a source of inspiration and also instrumental in carrying out this project work.

AKSHAY KUMAR RAI **1RN16EC014**

SANITH NAIR **1RN16EC124**

KUNDAN **1RN16EC057**

Table of Contents

ACRONYMS

ASL	American Sign Language
BSL	British Sign Language
ISL	Indian Sign Language
ReLU	Rectified Linear Unit
ELU	Exponential Linear Unit
CNN	Convolutional Neural Network
MLP	Multilayer Perceptron
ANN	Artificial Neural Network

Chapter 1

INTRODUCTION

American Sign Language (ASL) is a whole, natural language that has the same linguistic homes as spoken languages, with grammar that differs from English. ASL is expressed by means of actions of the hands and face. It is the primary language of many North Americans who are deaf and difficult of listening to, and is utilized by many listening to human beings as properly.

There is no standard universal sign language. Different sign languages are used in specific nations or regions. For example, British Sign Language (BSL) is a different language from ASL, and Americans who know ASL won't recognize BSL. Some nations undertake capabilities of ASL in their signal languages.

Sign language is a manner of conversation via humans tormented by speech and hearing loss. Around 360 million human beings globally be afflicted by disabling hearing loss out of which 328 million are adults and 32 million kids. Hearing impairment more than 40 decibels (dB) inside the higher hearing ear is referred as a disabling hearing loss. Thus, with increasing number of human beings with deafness, there is additionally an upward thrust in demand for translators. Minimizing the communication gap among hearing impaired and everyday people turns into a need to make sure effective conversation among all.

Since India doesn't have many Institutions for growing Indian sign language [other than ISLRTC which is established last year: would be future of ISL] there's lack of knowledge among the human beings and a few Institution indicates to select ASL over ISL without proper expertise.

Chapter 2

LITERATURE REVIEW

[1] Lih-Jen Kau, Wan Lin Su, Pei-Ju Yu and Sin-Jhan Wei, "A real-time portable sign language translation system", 2015 IEEE 58th International Midwest Symposium on Circuits and Systems (MWSCAS)

In this paper, a wi-fi hand gesture recognition glove is proposed for real-time translation of Taiwanese signal language. To discriminate between one of a kind hand gestures, we've got flex and inertial sensors embedded into the glove so that the three maximum important parameters, i.e., the posture of hands, orientation of the palm, and motion of the hand, described in Taiwanese Sign Language may be diagnosed without ambiguity. The finger flexion postures obtained with the aid of flex sensors, the palm orientation received via G-sensor, and the movement trajectory acquired through gyroscope are used because of the input indicators of the proposed system. The enter indicators could be acquired and tested periodically to look if it's miles a criminal signal language gesture or now not. Once the sampled sign can final longer than a predefined clock cycle, it's far regarded as a legitimate gesture and could be dispatched to mobile phone through Bluetooth for gesture discrimination and speech translation. With the proposed architecture and set of rules, the accuracy for gesture recognition is pretty satisfactory. As we can see in experiments that an accuracy rate as much as ninety 4% on sensitivity for gesture recognition may be achieved which justifies the superiority of the proposed structure.

[2] **Brandon Garcia and Sigberto Alarcon Viesca, "Real-time American Sign Language Recognition with Convolutional Neural Networks", Stanford University Journal Year - 2012**

A real-time signal language translator is a crucial a milestone in facilitating verbal exchange among the hearing impaired and most of the people. We hereby gift the development and implementation of an American Sign Language (ASL) fingerspelling translator primarily based on a convolutional neural network. We make use of a preskilled GoogLeNet structure educated at the ILSVRC2012 dataset, in addition to the Surrey University and Massey University ASL datasets on the way to apply switch mastering to this assignment. We produced a robust version that consistently classifies letters a-e efficiently with first-time customers and another that successfully classifies letters a-k in a majority of instances. Given the constraints of the datasets and the encouraging consequences, we are assured that with further and greater information, we can produce a completely generalizable translator for all ASL letters.

[3] Jestin Roy and Kannan Balakrishnan, "A prototype Malayalam to Sign Language Automatic Translator", Cornell University Journal Year – 2014

A prototype Malayalam to Sign Language Automatic Translator. Sign language, that's a medium of communication for deaf human beings, uses guide verbal exchange and body language to carry meaning, instead of the usage of sound. This paper affords a prototype Malayalam text to signal language translation device. The proposed gadget takes Malayalam textual content as enter and generates corresponding Sign Language. Output animation is rendered using a laptop-generated version. This device will help to disseminate facts to the deaf humans in public utility places like railways, banks, hospitals and so forth. This will also act as an educational device in learning Sign Language.

[4] Manny Rayner, Pierrette Bouillon, Sarah Ebling, Johanna Gerlach, Irene Strasly, Nikos Tsourakis , "An Open Web Platform for Rule-Based Speech-to-Sign Translation", Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)

An open internet platform for developing, compiling, and going for walks rule-based speech to sign language translation programs. Speech reputation is performed using the Nuance Recognizer 10.2 the toolkit, and signed output, which includes each guide and non-manual additives, is rendered the usage of the JASigning avatar gadget. The platform is designed to make the element technologies quite simply on hand to signal language professionals who are not always computer scientists. Translation grammars are written in a version of Synchronous Context-Free Grammar tailored to the peculiarities of signal language. All processing is finished.

On a faraway server, with content uploaded and accessed through a web interface, initial stories display those simple translation grammars may be implemented on a time-scale of some hours to three days and bring signed output conveniently understandable to Deaf informants. Overall, the platform notably lowers the barrier to access for researchers interested in constructing packages that generate tremendous signed language.

[5] R.San Segundo, J.M.Pardo, J.Ferreiros,V.Sama, R.Barra-Chicote , "Spoken Spanish generation from sign language", 2010

This paper describes the improvement of a Spoken Spanish generator from sign-writing. The signal language considered was the Spanish signal language (LSE: Lengua de Signos Española). This gadget consists of an advanced visible interface (where a deaf character can specify a chain of symptoms in sign-writing), a language translator (for producing the sequence of phrases in Spanish), and sooner or later, a textual content to speech converter. The visual interface lets in a signal collection to be described using numerous signal-writing options. The paper details the method for designing the visible interface proposing solutions for HCI-precise demanding situations whilst working with the Deaf (i.e. Crucial problems in writing Spanish or confined sign coverage for describing summary or conceptual ideas). Three strategies had been evolved and blended for language translation to put in force the very last model of the language translator module.

The summative assessment, carried out with Deaf from Madrid and Toledo, consists of objective measurements from the machine and subjective statistics from questionnaires. The paper also describes the first Spanish-LSE parallel corpus for language processing studies centered on particular domain names. This corpus consists of extra than 4000 Spanish sentences translated into LSE. These sentences focused on two limited domain names: the renewal of the identification report and driving force's license. This corpus additionally contains all sign descriptions in several signal-writing specifications generated with a new version of the eSign Editor. This new edition consists of a grapheme to phoneme system for Spanish and a SEA-HamNoSys converter.

[6] Daniel Kelly, John Mc Donald and Charles Markham, "Weakly Supervised

Training of a Sign Language Recognition System Using Multiple Instance Learning Density Matrices", IEEE Year - 2010

A machine for mechanically education and recognizing signs from continuous sign language sentences is offered. We suggest a singular multiple instance gaining knowledge of density matrix algorithm which robotically extracts remoted signs from full sentences the usage of the weak and noisy supervision of text translations. The mechanically extracted remoted samples are then utilized to train our spatiotemporal gesture and hand posture classifiers. The experiments have been carried out to evaluate the performance of the automated sign extraction, hand posture classification, and spatiotemporal gesture recognizing systems. We then perform a full assessment of our usual sign recognizing gadget which changed into mechanically trained on 30 specific sign

Chapter 3

METHODOLOGY

The Data

There are 2 data sets utilized in this model:

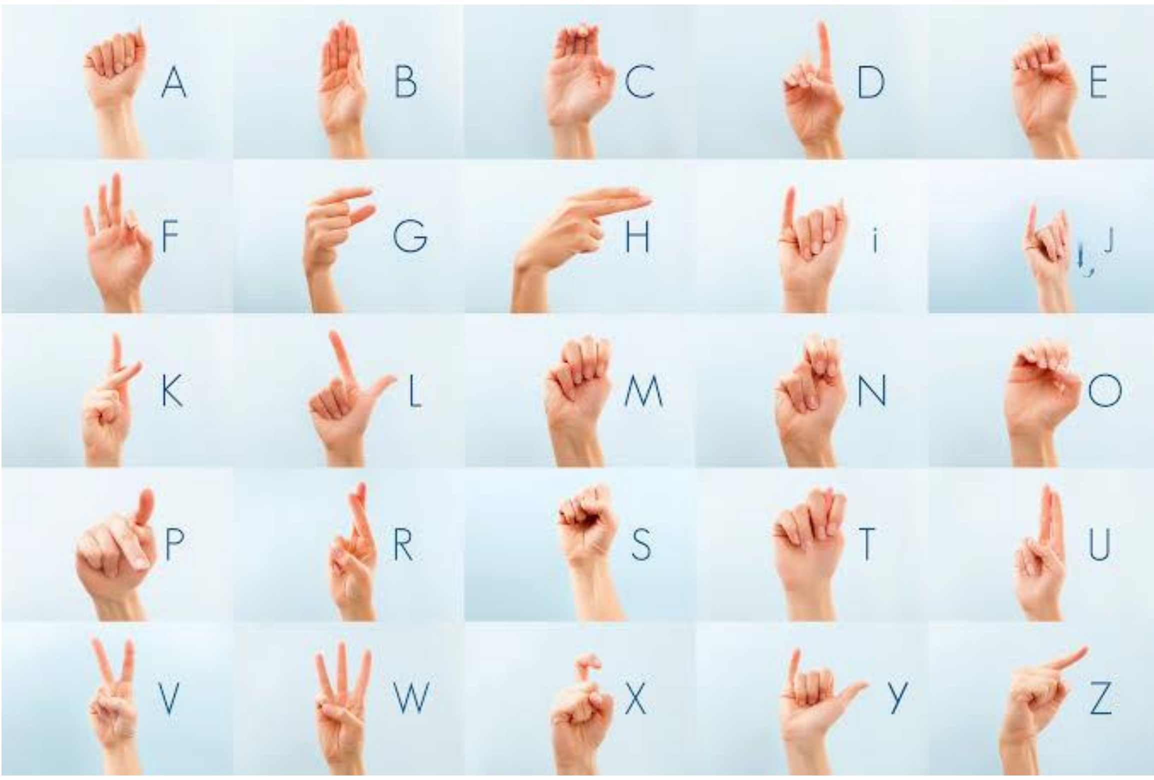
ASL Alphabet - This data set is the basis for the model. It has around 80k images for training.

ASL Alphabet Test - This data set was made specifically for validating the model created using the above data set, and is intended to be used to improve the feature engineering and modeling process to make it more versatile in "the wild" with less contrived images. It has around 8k images for testing.

Loading the dataset

The ASL Alphabet dataset is conveniently provided to us as part of the Keras library, so we can easily load the dataset. Out of the 88,000 images provided in the dataset, 80,000 are given for training and 8,000 are given for testing.

When we load the dataset below, X_train and X_test will contain the images, and y_train and y_test will contain the alphabets that those images represent.



Exploratory data analysis

We are going to label the 29 classes of images into their respective alphabetical and numerical order into a list["unique labels"].Plot them using the matplotlib library.

labels_dict={'A':0,'B':1,'C':2,'D':3,'E':4,'F':5,'G':6,'H':7,'I':8,'J':9,'K':10,'L':11,'M':12, 'N':13,'O':14,'P':15,'Q':16,'R':17,'S':18,'T':19,'U':20,'V':21,'W':22,'X':23,'Y':24,'Z':25,'space':26,'del':27,'nothing':28}

Data pre-processing

Next, we need to reshaping our dataset inputs (X_train and X_test) to the shape that our model expects when we train the model. The first number is the number of images (80,000 for X_train and 8,000 for X_test). Then comes the size of each image (64x64).

Building the model

Now we are ready to build our model.

The model type that we will be using is Sequential. Sequential is the easiest way to build a model in Keras. It allows you to build a model layer by layer.

We use the ‘add()’ function to add layers to our model.

There are six Conv2D layers. These are convolution layers that will deal with our input images, which are seen as 2-dimensional matrices.16 in first layer ,32 in second and third layer, 64 in fourth ,128 in fifth and 256 in sixth layer are the number of nodes in each layer.This number can be adjusted to be higher or lower, depending on the size of the dataset.

Kernel size is the size of the filter matrix for our convolution. So a kernel size of 3 means we will have a 3x3 filter matrix.A convolution multiplies a matrix of pixels with a filter matrix or ‘kernel’ and sums up the multiplication values. Then the convolution slides over to the next pixel and repeats the same process until all the image pixels have been covered.

Activation is the activation function for the layer. The activation function we will be using for our all six layers is the ReLU, or Rectified Linear Activation. This activation function has been proven to work well in neural networks.

The ReLU function is f(x)=max(0,x).

Usually this is applied element-wise to the output of some other function, such as a matrix-vector product. In MLP usages, rectifier units replace all other activation functions except perhaps the readout layer.

One way ReLUs improve neural networks is by speeding up training. The gradient computation is very simple (either 0 or 1 depending on the sign of xx). Also, the computational step of a ReLU is easy: any negative elements are set to 0.0 -- no exponentials, no multiplication or division operations.

Gradients of logistic and hyperbolic tangent networks are smaller than the positive portion of the ReLU. This means that the positive portion is updated more rapidly as training progresses. However, this comes at a cost. The 0 gradient on the left-hand side is

has its own problem, called "dead neurons," in which a gradient update sets the incoming values to a ReLU such that the output is always zero; modified ReLU units such as ELU (or Leaky ReLU, or PReLU, etc.) can ameliorate this.

We have used padding as "same' in the program.The same padding makes the size of outputs be the same with that of inputs when s=1. If s=1, the number of zeros padded is (k-1).

MaxPool: The pooling layer is usually placed after the Convolutional layer. The utility of pooling layer is to reduce the spatial dimension of the input volume for next layers. Note that it only affects weight and height but not depth.The max pool layer is similar to convolution layer, but instead of doing convolution operation, we are selecting the max values in the receptive fields of the input, saving the indices and then producing a summarized output volume. The implementation of the forward pass is pretty simple.

In between the Conv2D layers and the dense layer, there is a ‘Flatten’ layer. Flatten serves as a connection between the convolution and dense layers.

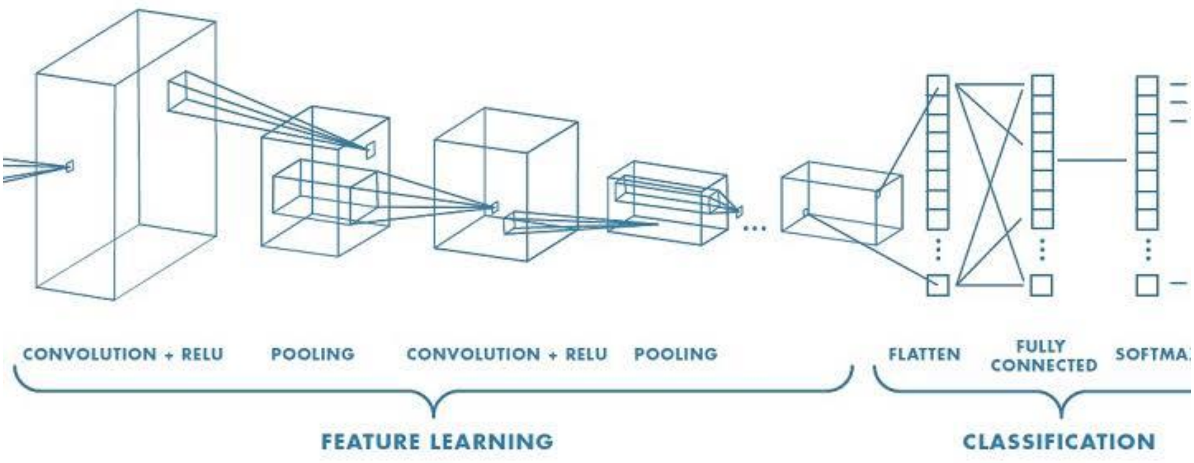
"Dense" is the layer type we will use in for our output layer. Dense is a standard layer type that is used in many cases for neural networks.A dense layer is just a regular layer of neurons in a neural network. Each neuron recieves input from all the neurons in the previous layer, thus densely connected. The layer has a weight matrix W, a bias vector b, and the activations of previous layer a.

"Dropout" is a a technique used to tackle Overfitting. The Dropout method in keras layers module takes in a float between 0 and 1, which is the fraction of the neurons to drop. Below is the docstring of the Dropout method from the documentation. Dropout consists in randomly setting a fraction rate of input units to 0 at each update during training time, which helps prevent overfitting.

The activation is ‘Softmax’. Softmax makes the output sum up to 1 so the output can be interpreted as probabilities. The model will then make its prediction based on which option has the highest probability.

Batch Normalization is used to normalize the input layer as well as hidden layers by adjusting mean and scaling of the activations. Because of this normalizing effect with additional layer in deep neural networks, the network can use higher learning rate without vanishing or exploding gradients. Furthermore, batch normalization regularizes the network such that it is easier to generalize, and it is thus unnecessary to use dropout to mitigate overfitting.

Model Architecture



Compiling the model

Next, we need to compile our model. Compiling the model takes three parameters: optimizer, loss and metrics.

The optimizer controls the learning rate. We will be using ‘adam’ as our optmizer. Adam is generally a good optimizer to use for many cases. The adam optimizer adjusts the learning rate throughout training.

The learning rate determines how fast the optimal weights for the model are calculated. A smaller learning rate may lead to more accurate weights (up to a certain point), but the time it takes to compute the weights will be longer.

We will use ‘categorical_crossentropy’ for our loss function. This is the most common choice for classification. A lower score indicates that the model is performing better.

To make things even easier to interpret, we will use the ‘accuracy’ metric to see the accuracy score on the validation set when we train the model.

Training the model

Now we will train our model. To train, we will use the ‘fit()’ function on our model with the following parameters: training data (train_X), target data (train_y), validation data, and the number of epochs.

For our validation data, we will use the test set provided to us in our dataset, which we have split into X_test and y_test.

The number of epochs is the number of times the model will cycle through the data. The more epochs we run, the more the model will improve, up to a certain point. After that point, the model will stop improving during each epoch. For our model, we will set the number of epochs to 5.

Using our model to make predictions

If you want to see the actual predictions that our model has made for the test data, we can use the predict function.If we have new data, we can input our new data into the predict function to see the predictions our model makes on the new data. Since we don’t have any new unseen data, we will show predictions using the test set for now.


prediction : [R]
R_test.jpg
prediction : [U]
U_test.jpg
prediction : [A]
A_test.jpg
prediction : [E]
E_test.jpg
prediction : [F]
F_test.jpg
prediction : [space]
space_test.jpg
prediction : [D]
D_test.jpg
prediction : [T]
T_test.jpg
prediction : [L]
L_test.jpg
prediction : [V]
V_test.jpg
prediction : [S]
S_test.jpg
prediction : [C]
C_test.jpg
prediction : [H]
H_test.jpg
prediction : [I]
I_test.jpg
prediction : [K]
K_test.jpg
prediction : [Z]
Z_test.jpg
prediction : [B]
B_test.jpg
prediction : [J]
J_test.jpg

Chapter 4

CONCLUSION

A CNN based Sign language recognition model was implemented for all 26 classes of alphabets. The model was trained using the dataset containing around 3000 images per alphabet for 26 alphabets (78000). The dataset would be divided in the ratio 9:1. Where 90% of the dataset (70000) is used for training the model, while 10% of the dataset (8000) was used for the validation of the trained model. The model was trained for 5 epoch which resulted in an accuracy of 97.3%.

FUTURE SCOPE

The project involves distinguishing among the hand gesture of all English alphabets in real time and translating into audio. Future work may include taking live data from webcam/camera and translating it simultaneously. Further, we may move on to recognising words by detecting facial expression, body movement with hand gesture, from as large a dictionary as possible, for American Sign Language. Another method to improve the performance is by using a more accurate method for sign language detection.

REFERENCES

[1] Beifang Yi Dr. Frederick C. Harris ‘A Framework for a Sign Language Interfacing System’, A dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Computer Science and Engineering May 2006 University of Nevada, Reno.

[2] Helene Brashear & Thad Starner ‘Using Multiple Sensors for Mobile Sign Language Recognition’, ETH - Swiss Federal Institute of Technology Wearable Computing Laboratory 8092 Zurich, Switzerland flukowicz, junker.

[3] Jose L. Hernandez-Rebollar1, Nicholas Kyriakopoulos1, Robert W. Lindeman2 ‘A New Instrumented Approach For Translating American Sign Language Into Sound And Text’, Proceedings of the Sixth IEEE International Conference on Automatic Face and Gesture Recognition (FGR’04) 0-7695-2122-3/04 $ 20.00 © 2004 IEEE.

[4] K. Abe, H. Saito, S. Ozawa: Virtual 3D Interface System via Hand Motion Recognition from Two Cameras. IEEE Trans.Systems, Man, and Cybernetics, Vol. 32, No. 4, pp. 536–540, July 2002.

[5] T.Starner and A. Pentland. Real-Time American Sign Language Recognition from Video Using Hidden Markov Models. Computational Imaging and Vision, 9(1); 227-243, 1997.

[6] M. Jeballi et al. Extension of Hidden Markov Model for Recognizing Large Vocabulary of Sign Language. International Journal of Artificial Intelligence & Applications 4(2); 35-42, 2013

[7] H. Suk et al. Hand gesture recognition based on dynamic Bayesian network framework. Patter Recognition 43 (9); 3059-3072, 2010.

[8] P. Mekala et al. Real-time Sign Language Recognition based on Neural Network Architecture. System Theory (SSST), 2011 IEEE 43rd Southeastern Symposium 14-16

March 2011.

[9] Y.F. Admasu, and K. Raimond, Ethiopian Sign Language Recognition Using Artificial Neural Network. 10th International Conference on Intelligent Systems Design and Applications, 2010. 995-1000.

[10] J. Atwood, M. Eicholtz, and J. Farrell. American Sign Language Recognition System. Artificial Intelligence and Machine Learning for Engineering Design. Dept. of Mechanical Engineering, Carnegie Mellon University, 2012.

[11] L. Pigou et al. Sign Language Recognition Using Convolutional Neural Networks. European Conference on Computer Vision 6-12 September 2014

[12] Y. Jia. Caffe: An open source convolutional architecture for fast feature embedding. http://caffe.berkeleyvision.org/, 2014.

[13] Lifeprint.com. American Sign Language (ASL) Manual Alphabet (fingerspelling) 2007.