International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

A Novel Method for Recognizing Hand Gestured Sign Language Using the Stochastic Gradient Descent Algorithm and Convolutional Neural Network Techniques

Dr. Vinod B. Durdi [1], Mr. Anish Kiran [2], Mr. Arvind Rao[3], Mr. Sagar S. Bhat [4], Dr. Santhosh B. [5], Dr. Mahesh Kumar N.[6]

Submitted: 03/10/2023 **Accepted :** 23/11/2023 **Accepted:** 03/12/2023

Abstract: Communication helps people to express their ideas, views, feelings and understand emotions of others. It is a process that is dynamic, systematic, transactional, adaptive and continuous. Human gestures have long been an important way of communication for people. It includes movement of hands, face or other parts of body. They are used mainly by those who have hearing impairment or are non-verbal to communicate with the rest of the world. Sign language is a uses visual-manual modality to convey meanings of messages through the employment of gestures, postures and movements. These specially-abled people who have some degree of hearing and speaking disability cannot understand oral language spoken by others vis-à-vis sign languages are not understood universally. The proposed scheme aims at implementing computer vision which takes sign language from the user and converts them into text. This way the communication barrier between deaf, mute people and the rest of the world is bridged. The work focuses primarily on the Hand Gesture Recognition system, which offers us a novel, real-time, intuitive, and user-friendly method of computer interaction that is more identical to human beings. In the current environment, security and protection are required for any online connection and network application. For this, a data hiding algorithm has been created. It uses a dynamic approach of cryptography to implement securing of the messages generated through this system. The results depicts the output of Sign Language Recognition. It executed in Jupyter Notebook environment using CNN i.e. Convolutional Neural Networks. For each of the evaluated datasets, the CNN model showed good accuracy. Despite the new dataset's volume and various settings, it produced good predictions and achieved more than 90 percentage 97% accuracy. Additionally, the fact that our dataset was produced under variable conditions such as dim light, heavy applications consuming more RAM , which increases power and cpu consumption of Stochastic Gradient Descent Algorithm, but it provides faster and efficient results. The proposed system might be considered as a promising solution in medical applications where a convolution neural network is used because of its increased accuracy.

Keywords: *Convolutional neural networks, Machine learning, OpenCV, Stochastic gradient descent algorithm, TensorFlow.*

1. Introduction

Communication is an essential life skill. Good communication leads to better understanding. Sign language is a language used by people who cannot hear or speak, using hands to make signals instead of spoken words [1]. Deaf and mute individuals who suffer with listening or speaking can communicate with each other by using sign language, which is a vital communication tool [1-3]. Not everybody understands sign language, picking up a new language is difficult and time consuming as well for the rest of us. The number of people who communicate using sign language constitute a major chunk which cannot be ignored as well. According to a survey 6.3% of the people living do not havethe ability to speak and hear. There are professional translators and interpreters who come to the rescue of those who cannot understand or use sign language, but the flip side is that their service is expensive and may not be available at all times[[4]. The minimum rate charged by is translator is very high for few words only. Many people cannot afford to pay their service. The sign languages of various nations vary. Different sign languages also frequently employ non-manual indications like body movements and facial emotions. Both hands or multiple gestures are typically used in these indications. The implementation of sign language interpretation systems is made more challenging by these issues. In order to overcome this, researchers developed systems that can understand sign language [5, 6]. This prompted us to take up this scheme and contribute to providing a solution for the difficulty they face. They can be independent by using an autonomous system. This system can also be extended to general, industrial, commercial and military environment where surrounding are loud and noisy. Communication between people working in noisy environments is highly challenging due to the noise as well as distractions all around them[7][8].The productivity and hence the efficiency of the workers will reduce because of lack of communication. They can be easily benefitted by using such systems and communicate with ease. The remaining work is further organized as follows, Section briefly stated the previous work, section 3 mentioned the proposed architecture. Section 4 gives the details of flow deign, Section 5, introduces the

[1] *Department of Electronics & Telecommunication Engineering , Dayananda Sagar College of Engineering, Bangalore-560111, Karnataka, India,*
[,2,3,4,] *Department of Electronics & Telecommunication Engineering , Dayananda Sagar College of Engineering, Bangalore-560111, Karnataka, India,*
[5]*Department of Electronics & Telecommunication Engineering , Dayananda Sagar College of Engineering, Bangalore-560111, Karnataka, India,*
[6,] *Department of Electronics & Communication Engineering , Dayananda Sagar College of Engineering, Bangalore-560111, Karnataka, India*
** Corresponding Author Email: vinoddurdi-tce@dayanandasagar.edu*

methodology. In section 6, results discussed, finally section 7 concludes the work.

2. Previous work

Brief summary of literature survey carried out has been complied in the table 1.

Table 1. Literature review

Sl. No	Title of Technical Paper & Year	Objectives	Gaps
1	Authored, Aradhana Kar et al. [2]	Minimize the processing time taken for mapping the sign language using video based approach	System designed was less accurate Only 70% success
2	Authored, Lean Karlo et al. [3]	Basic static signs, numerals, and ASL alphabets were all included in the work scope (A–Z)	Avg. time per trail of the system is more
3	Authored Shailesh Bachani et al. [9]	The systems convert gestures into simple words and also makes up a complete sentence in English	ANN(Artificial Neural network) processused is less powerful than CNN 2020
4	Deepraj Pradhan et al. [10]	This paper used Bit Rotation technique to encrypt the given data for more secured Communication	The system can be usefulfor static ISL numerical signs only
5	Authored B. Akiwate et al. [11]	The unicode DNA technology encrypts all forms of media, including text, messaging, audio, and video.	Presents Computational complexities
6	Authored Mohamed Aktham Ahmed al et. [12]	Conduct analyses of the glove systems for SLR device characteristics, develop a roadmap for technology evolution, discuss its limitations, and provide valuable insights into technology	Help researchers to understand the current options and gaps in this area, thus contributing to this line of research.

As seen earlier, communicating with persons who have hearing and language impairments is extremely difficult. Systems that can identify various indications and provide information to common people are thus necessary.

There exist many systems already which can recognize sign language. Identification of sign gestures is mainly done by glove-based methods, which can be divided into static and dynamic recognition. The signer must wear a hardware glove while the hand movements are being recorded in this glove-based method [13].

Methods based on vision can be further divided into static and dynamic recognition. Statics deals with the identification of static gestures, whereas dynamic is a real actual recording of the motions (2d-images). This entails using the camera to record movement [14].

The user of the glove-based technique, as depicted in Figure 1, had to put on a specific glove that could detect the position as well as the orientation of their hands. Despite being over 90% accurate, it looked a little unwieldy for everyday use [15]. The equipment are also expensive.

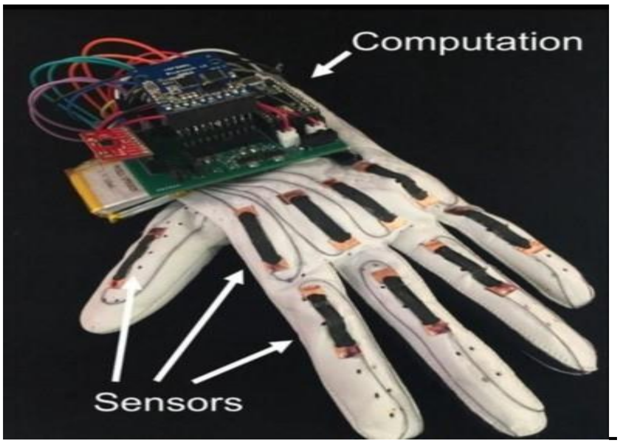


Fig. 1 Glove based approach

The demerits of vision based methods are that it used ANN instead of CNN which is less stable and time consuming [16]. Worldwide, there are numerous varieties of sign languages. One of the sign languages is American Sign Language (ASL) [2], along with Bangala Sign Language (BSL).

The majority of the work done on sign language focused with ASL and BSL. The difference between ASL and BSL has been shown in figure 2. The BSL which employs the usage of both the hands to convey the message instead of one. Hence the accuracy and precision of the system were drastically reduced. The work was solely limited to include recognition of alphabets.

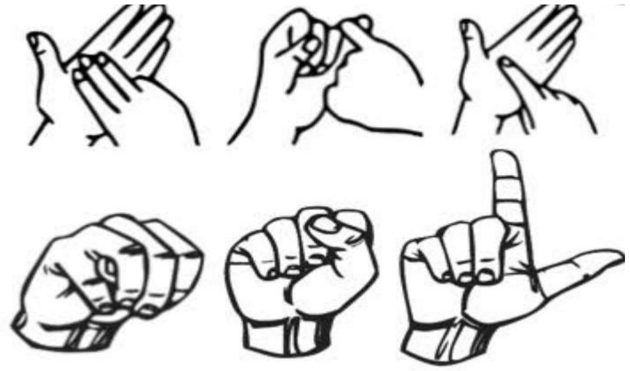


Fig. 2 Usage of ASL vs BSL

It did not take into account numbers and alphanumeric characters. Also they did not predict any words or sentences based upon the letters identified. The above mentioned short-comings have been considered to overcome in this work. For the analysis ASL has been considered.

3. Proposed Architecture

The most commonly used algorithm for ASL Recognition is Batch Gradient Descent algorithm. This is implemented for getting the best optimal result with more computation time.

Table 2: Comparison between Existing and Proposed Design

Sl No	**Batch gradient Descent (Existing Method)**	**Stochastic Gradient Descent(SGD) (Proposed Method)**
1	By using entire Training sample, compute the gradient.	Calculate the gradient using only one training data set
2	Slow and expensive algorithm in terms of computati on	faster and more efficient computationally than Batch gradient
3	Not advised for large training samples	Large training sample sizes are possible to use
4	Deterministic in nature	Stochastic in nature
5	Provides the best result if convergence is given enough time.	provides a good solution, but not the best

The proposed work[17][18] is based upon Stochastic Gradient Descent as it is computation is fast and is easy to fit the memory due to a single training sample being processed by the network. Comparison [1] between Existing and Proposed algorithm is compiled in Table 2:

Machine learning algorithms frequently use the optimization method gradient descent to get the weights or coefficients. [19,20]. The figure 3 depicts the compressional analysis between various algorithms SGD is much faster because there are far fewer data to modify at once because it chooses a random instance of training data at each step before computing the gradient. SGD attempts to address the fundamental issue of using the entire training set to determine gradients at each step. The SGD error function is noisy or moves in zigzags in the direction of global minima. As fewer recordings are chosen for each epoch, less computing memory is needed.

$for\ i\ in\ range\ (m)$:

$$\theta j = \theta j - \alpha(\widehat{y^i} - y^i)X_j^i \qquad (1)$$

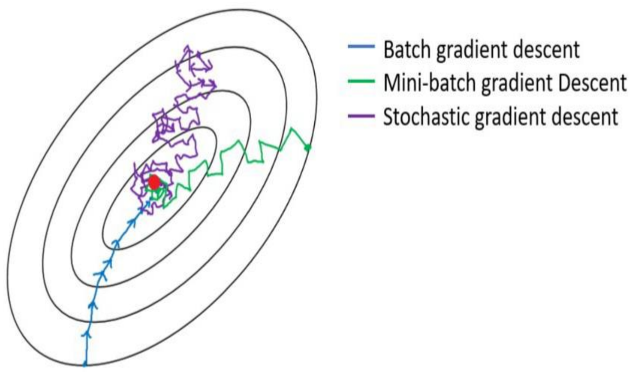


Fig. 3 2D diagram to represent global minima points with loss as a function of 2 weights for gradient descent, mini-batch gradient descent and stochastic gradient

For each run of this algorithm's optimization phase, random samples from the training data are selected. By doing this, it not only fixes computed errors and updates weights in faster rounds, but it also aids in the quicker approach to a minimum. It also helps to improve computation a whole lot faster. Large datasets often can't be held in RAM, which makes vectorization much less efficient. At every epoch, it takes in one record, finds out the derivative and updates error in the weight. By a technique called mini batch SGD, K data points are chosen which is less than n i.e., total number of records for every iteration. Stochastic gradient descent with momentum is used to remove noise that is involved during the training phase of the mode.

4. Flow Design

The figure 4 depicts the flow chart of sign language recognition, which starts by capturing the image. The image captured is then labelled and a label map is created, for this label mapTFRecords is generated and then updated in pipeline.config. The updated file is trained as custom model which is used to detect the SLR input.

The figure 5 shows the flow of encryption. The output from SLR block is in string format which is fed as input to the Cryptography machine module which first performs RAA (to generate a partially encrypted keyword.

To make the keyword more secured character rotation technique, ASCII and Binary conversion is carried out.

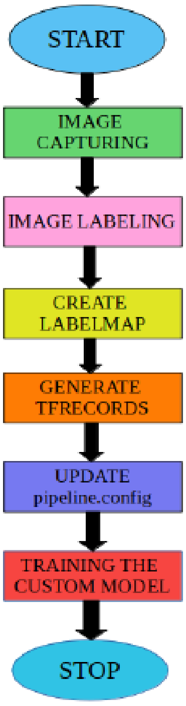


Fig. 4 Flow chart of sign language recognition

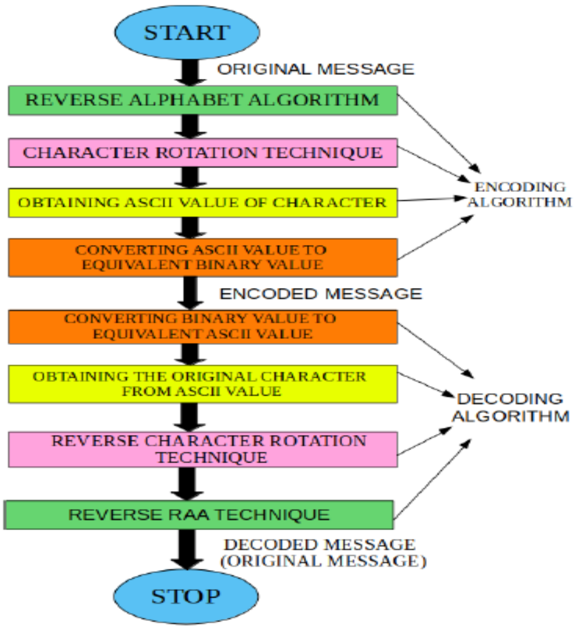


Fig. 5 Flow Chart of Crypto Machine

In decryption inverse of encryption takes place to retain back the original message.

5. Methodology

The figure 6 depicts the block diagram of the SLR system. The system is already trained with pre-captured images and tested so that it works under different conditions. Labelling is the model used to train the model. It crops and generates xml files for each image. The image captured by the webcam is fed as input to the SLR block, which consists of pretrained and tested models. After the input is fed it compares with pre-trained gesture and based on the comparison output text is displayed on Monitor which will be in string format. This output string is then encrypted using RBA i.e., Reverse Binary Algorithm and a binary equivalent encrypted keyword is generated[21]. Then by inverse RBA technique the encrypted keyword is retained back.

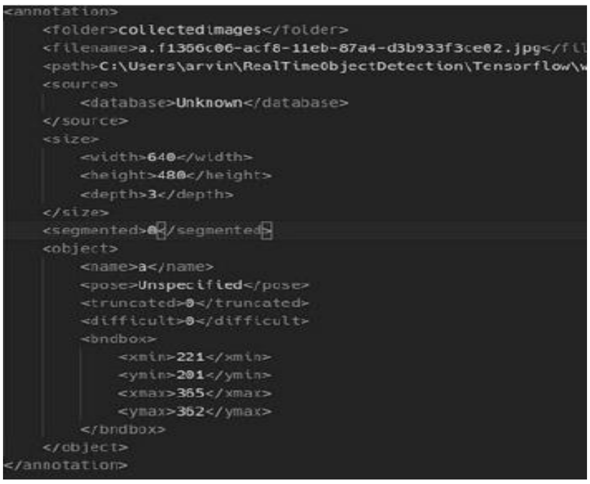
5.1 Image Capturing

The very first step is capturing images through the webcam or a static camera as shown in figure 7 with help of OpenCV library. OS, time and uuid dependencies are used in this step to capture image and provide distinct names for each image. Each sign language pose is allotted a specific label and number of images that has to be captured and later trained upon a pre-trained model using transfer learning. Collecting images with different orientations is ideal to get a better efficiency, and to remove noise in the images Gaussian filter is used whose impulse response is Gaussian function. These captured images are stored in a centralized database at first and eventually partitioned.

Fig. 7: Capturing an Image using OpenCV

5.2 Image labelling:

After capturing the images, each images are labelled manually. It is the process of identifying and marking necessary details of an image. LabelImg is a graphical image annotation tool that is written in python[22][23] and Qt for its graphical interface. This package is used to help setting up label for every image. It creates XML files as shown in figure 8 which contains all meta information like size, folder path representing an object. The given object can be cropped and assigned label accordingly as shown in figure 9.



```
<annotation>
    <folder>collectedimages</folder>
    <filename>a.f1356c06-acf8-11eb-87a4-d3b933f3ce02.jpg</fil
    <path>C:\Users\arvin\RealTimeObjectDetection\Tensorflow\w
    <source>
        <database>Unknown</database>
    </source>
    <size>
        <width>640</width>
        <height>480</height>
        <depth>3</depth>
    </size>
    <segmented>0</segmented>
    <object>
        <name>a</name>
        <pose>Unspecified</pose>
        <truncated>0</truncated>
        <difficult>0</difficult>
        <bndbox>
            <xmin>221</xmin>
            <ymin>201</ymin>
            <xmax>365</xmax>
            <ymax>362</ymax>
        </bndbox>
    </object>
</annotation>
```

Fig. 8: A Sample XML file

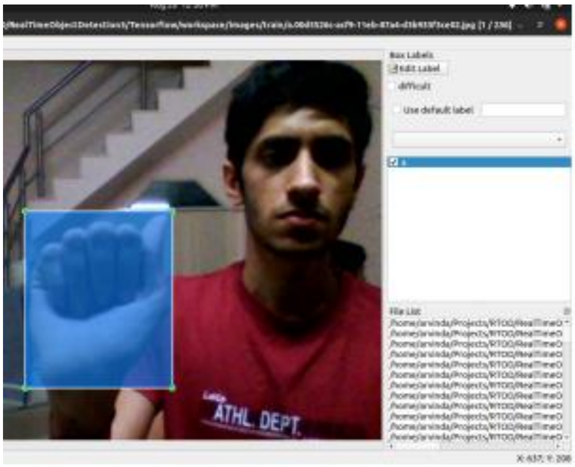


Fig. 9: Create a label for each sign language pose

```
labels = [{'name':'a', 'id':1},
          {'name':'b', 'id':2},
          {'name':'c', 'id':3},
          {'name':'d', 'id':4},
          {'name':'e', 'id':5}
         ]

with open(ANNOTATION_PATH + '\label_map.pbtxt', 'w') as f:
    for label in labels:
        f.write('item { \n')
        f.write('\tname:\'{}\'\n'.format(label['name']))
        f.write('\tid:{}\n'.format(label['id']))
        f.write('}\n')
```

Fig. 10: Sample script for creating labelmap.pbtxt

5.2 Create labelmap:

Once all images are labeled as depicted in figure 10, they *are partitioned into training and testing folders [24-26]*. The model will be allowed to train on certain set of data and evaluated on different data which ideally helps in reducing the chances of overfitting. A labelmap provides a means to represent an item with *name* and *id* attributes. This file is usually created to make an object detection model recognize the particular sign language pose using Stochastic Gradient Descent Algorithm. This is stored in *pbtxt* file format which is understood by Tensorflow library

5.4 Generate TfRecords:

An object detection model named ssd_mobNet is always trained using TFRecord. The TFRecord is a Tensorflow format that is used for storing a sequence of binary records. It is special file format used by Tensorflow Object Detection API[27]. We generate records using generate_tfrecord script by importing it from the official repository and generate TFRecord scripts for each of train and test data partitions as shown in figure 11

Fig. 11 TFRecords used to store and process images in files

To use data extracted from TFRecord for training a model, an iterator will be created on the dataset object. The data extracted from TFRecords is later sent to train our model.

5.5 Update pipeline.config:

Download the ssd_mobileNet_v2 (pre-trained models) from tensorflow. Baseline configuration is the set of steps and model information that is used in order to train the model . With the help of ssd_mobileNet_v2 pre-trained model, we leverage the power of transfer learning and provide the trained data information into the pre-trained model and they are stored as shown in the figure 11. The captured images are originally in RGB format. They are converted into their respective binary images depending on pixelation values. The baseline configuration consists of all necessary parameters like num_classes, the batch size (data processed within each epoch), path of label_map, and tfrecords, etc

5.6 Training the custom model:

The very last step is to train deep learning model by feeding all the custom images [28, 29]. The training model is used to run the input data through the algorithm to correlate the processed output against the sample output as shown in figure 12.The result from this correlation is used to modify the model. There is a provision to change the step size for the given pipeline configuration. This can be adjusted to optimal efficiency. The learning rate for each epoch can be varied so as to get least minimum error cost function.

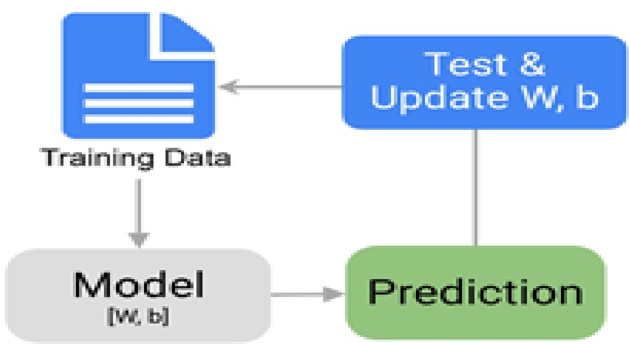


Fig. 12 Image that depicts how to train deep learning model

5.7 Encryption:

The text generated from the Sign Language recognition needs to be safeguarded[30]. Hence this work uses a Reverse Binary Algorithm (RBA) technique to encrypt the data. The above generated text is given as input to the encryption module. The encryption module consists of pre-defined keys that are used to produce different character set for the input string based on RAA technique which uses the concept of reverse string. Now the text generated is partially encrypted but can be easily cracked by an intruder. Hence to make it more secured and maintain the secrecy of the text some more parameters are included, which are - Character Rotation, Converting the character to their respective ASCII value and Obtaining Equivalent Binary Number of the ASCII value.

5.8 Decryption:

After obtaining the encrypted keyword it has to be converted back to its original form. Hence Reverse RBA technique is used to decrypt the data. As encryption uses a very complicated algorithm, decryption algorithm used is also equally complicated and follows 4 steps - Obtaining Equivalent ASCII value of the Binary Number, Converting the ASCII value to their respective character, Reverse Character Rotation, Reverse RAA Technique

6. Results

The following images depicts the output of Sign Language Recognition. As mentioned above it has been executed in Jupyter Notebook environment using CNN i.e. Convolutional Neural Networks. In the figure 13 SLR is shown recognizing letter "A" with an accuracy of 97%. If the accuracy of recognition is above 95% then it indicated through a green colored frame and if the accuracy is below 95% it is appears in a white colored frame

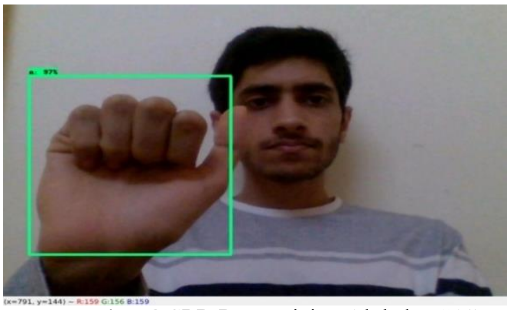


Fig. 13 SLR Recognizing Alphabet "A"

Fig. 14 SLR Recognizing Alphabet "D"

In the above figure 14 SLR is shown recognizing letter "D" with an accuracy of 90%. As the accuracy is below 95% it is appears in a white coloured frame.

This scheme not only recognizes alphabets but also recognizes numbers and common phrases used in everyday life.

Fig. 15 SLR Recognizing Number 0

The above fig 15 shows SLR Recognizing Number 0. As mentioned above the output of SLR is in string format. Hence, we get the output in terms of string as "Zero" for number 0.

The figure 16 shows SLR Recognizing Number 4. As mentioned above the output of SLR is in string format. Hence, we get the output in terms of string as "Four" for number 4. These are some of the commonly used phrases recognized by SLR .

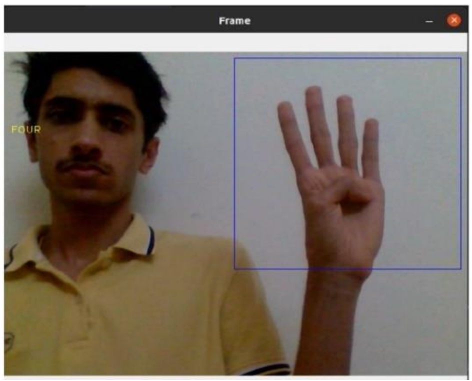


Fig. 16 SLR Recognizing Number 4

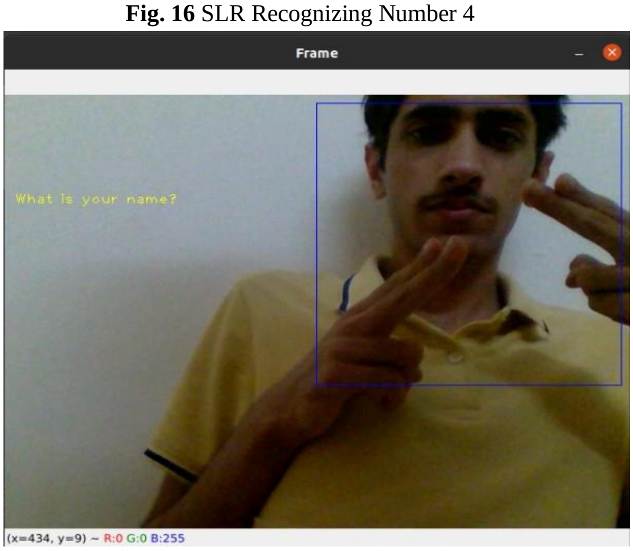


Fig. 17: SLR Recognizing the Phrase "What is your name?"

The above figure 17 depicts SLR detecting the Phrase "What is your name?", which is fed as a input to the cryptoMachine which encodes the word and converts it into its binary equivalent based on the ASCII value of the characters. The output of both encryption and decryption is shown in figure 18.

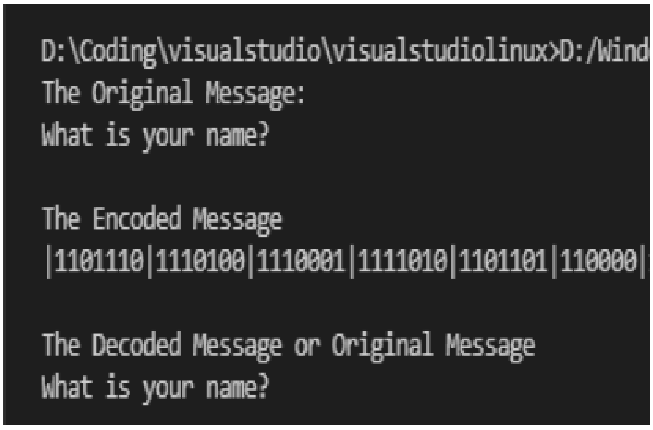


Fig. 18 : CryptoMachine output for the Phrase "What is your name?"

.**Fig. 19:** SLR Recognizing the Phrase "How are you?"

The above figure 19 depicts SLR detecting the Phrase "How are you?" which is fed as a input to the cryptoMachine which encodes the word and converts it into its binary equivalent based on the ASCII value of the characters. The output of both encryption and decryption is shown in figure 20.

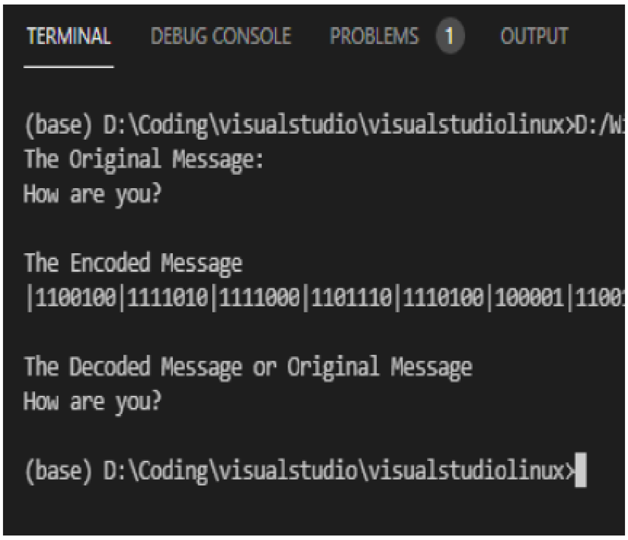


Fig. 20 CryptoMachine output for the Phrase "How are you?"

Fig. 21: SLR Recognizing the Phrase "Greetings!"

The figure 21 depicts SLR detecting the Phrase "Greetings!" which is fed as a input to the crypto Machine which encodes the word and converts it into its binary equivalent based on the ASCII value of the characters. The output of both encryption and decryption is shown in figure 22.

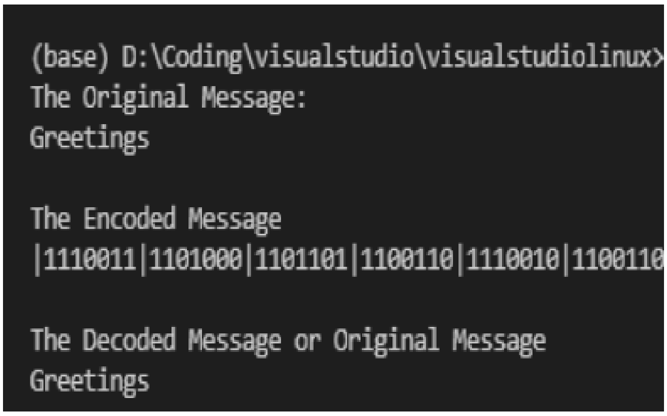


Fig. 22: CryptoMachine output for the Phrase "Greetings!"

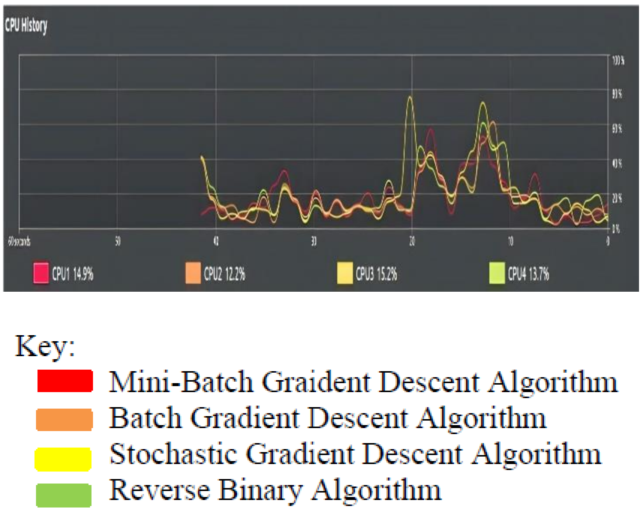


Fig. 23: Performance analysis of different SLR algorithms

Based on the above figure 23, depicts the performance analysis of different algorithms.

The below table 3 concludes that though power and cpu consumption of Stochastic Gradient Descent Algorithm is more but it provides faster and efficient results.

Parameters	**Mini Batch Gradient Descent Algorithm**	**Batch Gradient Descent Algorithm**	**Stochastic Gradient Descent Algorithm**
Power	22.03%	26.42%	43.4%
Timer	2.1s	3.2s	0.9s
CPU consumption	14.9%	12.2%	15.2%
Efficiency	≤ 75%	≤ 70%	≥ 90%

Note: the following analysis have been deduced after integrating SLR algorithms with Reverse Binary Algorithm

7. Conclusion

This model proposed reflects the design of next-generation sign language recognizers. A new system designed with a output consistent with the provided data input. The system has used hand postures for communication in input, and gives output in English alphabets. This design can be used in various industrial, commercial, military and personal environments.

The system is a faster way of recognizing sign language poses. The model has used images and their corresponding xml files as the input and ssdmobilenet_v2 as a pre-trained model to render the output with help of transfer learning. Tensorflow and OpenCV are run using python. The system can be used outdoors and can provide sign for different poses which the user enacts.

This is useful to automate the work of interpreters and minimizing human workforce. It also saves time and reduces the risk of misinterpretation because the system is more accurate when compared to manual prediction.

This way the communication barrier between deaf, mute people and the rest of the world is overcome. The system also aims at encrypting the data using secure and advanced algorithms so that there is no data breach to malicious third parties.

The future scope of the scheme can be expanded to include other sign languages such as Indian sign language, Bangla sign language etc. The time taken to give output by the system can be further reduced by using better algorithm designs which are faster, hence taking less computational time. Speed of the system can be increased by reducing latency i.e. the delay from input into a system to desired outcome. More number of images can be included in dataset and trained to increase accuracy and precision of the system therefore the efficiency of the process will be enhanced.

Acknowledgment

The following paper would not have been possible without the contribution of the following references. We are immensely thankful to the authors of the paper.

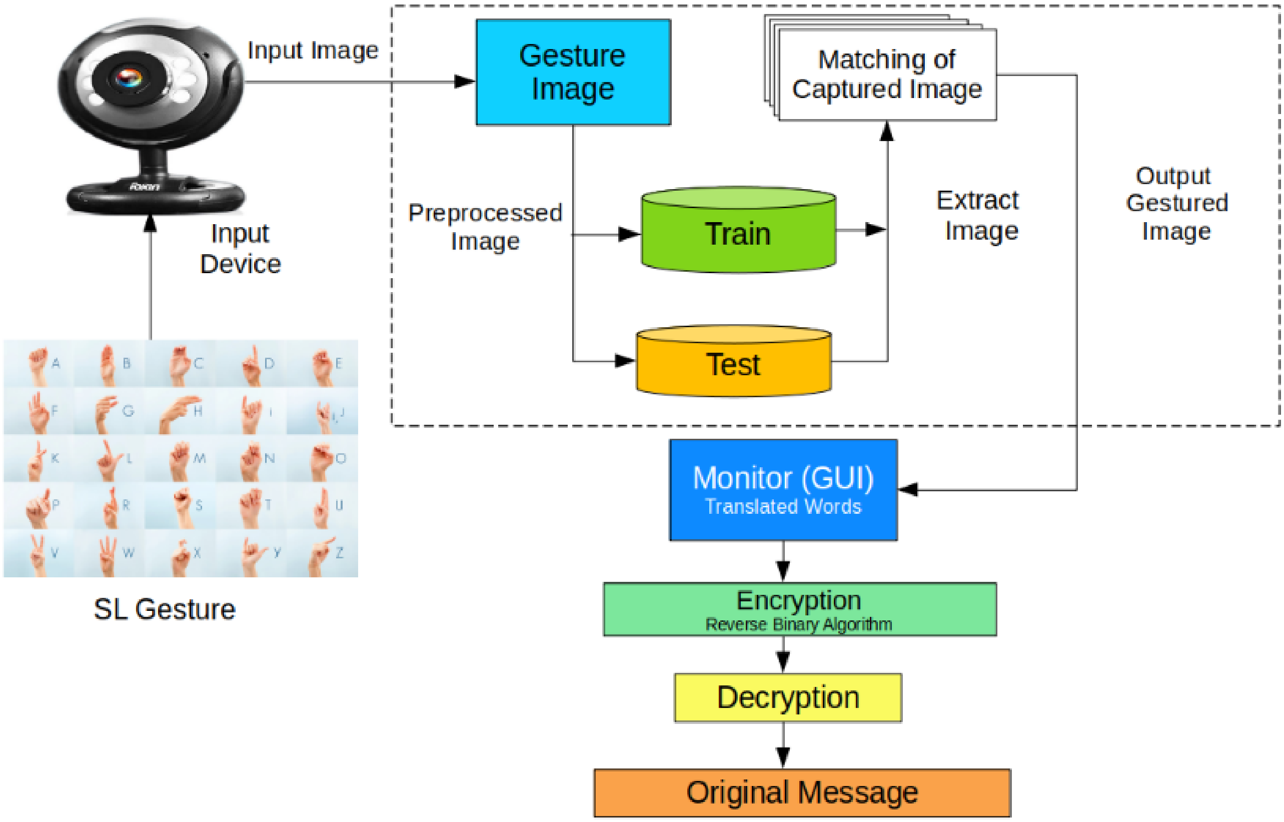


Fig. 6 Block diagram of proposed system

References

[1] Radha S Shirbhate, Mr Vedant D Shinde, Ms. Sanam A Metkari, Ms Pooja U Norkar , "Sign Language Recognition Using Machine Learning Algorithm", *International Research Journal of Engineering and Technology* , vol. 07,Issue 03, pp 2122-2125, March 2020.

[2] Kar and P. S. Chatterjee, "An Approach for Minimizing the Time Taken by Video Processing for Translating Sign Language to Simple Sentence in English," *in Proc. 2015 in International Conference on Computational Intelligence and Networks*, Odisha, India, pp. 172-177, 2015, doi: 10.1109/CINE.2015.40.

[3] Lean Karlo S. Tolentino, Ronnie O. Serfa Juan, August C. Thio-ac, Maria Abigail B. Pamahoy, Joni Rose R. Forteza, and Xavier Jet O. Garcia "Static Sign Language Recognition Using Deep Learning" *International Journal of Machine Learning and Computing*, vol. 9, No. 6, pp.821-827, December 2019,doi: 10.18178/ijmlc.2019.9.6.879.

[4] Mohammed Zakariah, , Yousef Ajmi Alotaibi, Deepika Koundal, Yanhui Guo, and Mohammad Mamun Elah ,"Sign Language Recognition for Arabic Alphabets Using Transfer Learning Technique", Comput Intell Neurosci. 2022; 2022: 4567989.Published online 2022 Apr 22. doi: 10.1155/2022/4567989

[5] M. Ahmed, M. Idrees, Z. ul Abideen, R. Mumtaz and S. Khalique, "Deaf talk using 3D animated sign language: A sign language interpreter using Microsoft's kinect v2," in proc. of *2016 SAI Computing Conference (SAI)*, London, UK, pp. 330-335, 2016, doi: 10.1109/SAI.2016.7556002.

[6] Ahmed Kasapbaşi, Ahmed Eltayeb Ahmed Elbushra, Omar Al-Hardanee, Arif Yilmaz,"Deepaslr: A CNN Based Human Computer Interface For American Sign Language recognition for hearing- impaired individuals", *Computer Methods and Programs in Biomedicine,* vol. 2,2022,100048, ISSN 2666- 9900

[7] Ibarguren A., Maurtua I., Sierra B. "Layered architecture for real time sign recognition: Hand gesture and movement. " *Eng. Appl. Artif. Intell.* 2010;23:1216–1228. doi: 10.1016/j.engappai.2010.06.001.

[8] Das P., De R., Paul S., Chowdhury M., Neogi B. "Analytical study and overview on glove based Indian Sign Language interpretation technique"; *in proc. Proceedings of the Michael Faraday IET International Summit 2015*; Kolkata, India. 12–13 September 2015.

[9] Shailesh Bachani, Shubham Dixit, Rohin Chadha, Prof. Avinash Bagul "Sign Language Recognition Using Neural Network", *International Research Journal of Engineering and Technology*, vol: 07 Issue: 04 | Apr 2020,pp.583-586, 2020.

[10] D. Pradhan, S. Som and A. Rana, "Cryptography Encryption Technique Using Circular Bit Rotation in Binary Field," 2020 8th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) *(ICRITO)*, Noida, India, 2020, pp. 815-818, doi: 10.1109/ICRITO48877.2020.9197845.

[11] B. Akiwate and L. Parthiban, "A Dynamic DNA for Key-based Cryptography," *in proc. 2018 International Conference on Computational Techniques, Electronics and Mechanical Systems (CTEMS)*, Belgaum, India, 2018, pp. 223-227,doi:10.1109/CTEMS.2018.8769267.

[12] Mohamed Aktham Ahmed, Bilal Bahaa Zaidan, Aws Alaa Zaidan, Mahmood Maher Salih, Muhammad Modi bin Lakulu, "A Review on Systems-Based Sensory Gloves for

Sign Language Recognition State of the Art between 2007 and 2017”,*Sensors* 2018, *18*(7),2208; https://doi.org/10.3390/s18072208

[13] S.A. Mehdi; Y.N. Khan, ”Sign language recognition using sensor gloves”, *in proc. of the 9th International Conference on Neural Information Processing, vol 5,pp 2204-2206,* 2002. ICONIP '02. Doi 10.1109/ICONIP.2002.1201884

[14] Suharjito,Anderson R., Wiryana F., Ariesta M.C., Kusuma G.P. Sign Language Recognition Application Systems for Deaf-Mute People: A Review Based on Input-Process-Output.”, *Procedia Comput. Sci. , vol 116, pp: 441-448,* 2017

[15] Oudah, M.; Al-Naji, A.; Chahl, J. “Hand Gesture Recognition Based on Computer Vision:A Review of Techniques.” *J.Imaging* , vol. *6*, issue 8, pp 1-29, 2020 https://doi.org/10.3390/jimaging6080073

[16] Alzubaidi, L., Zhang, J., Humaidi, A.J. et al. “ Review of deep learning: concepts, CNN architectures, challenges, applications, future directions” *Springer Open Access*, Page 1-74, J Big Data 8, 53, 31 March 2021, https://doi.org/10.1186/s40537-021-00444-8

[17] Latz, J. “Analysis of stochastic gradient descent in continuous time.”, *Stat Comput* 31, article 39, 09 May 2021 ,https://doi.org/10.1007/s11222-021-10016-8.

[18] S. Suresh, H. T. P. Mithun and M. H. Supriya, "Sign Language Recognition System Using Deep Neural Network,", *in proc. 2019 5th International Conference on Advanced Computing & Communication Systems (ICACCS)*, Coimbatore, India, pp. 614-618, 2019, doi: 10.1109/ICACCS.2019.8728411.

[19] S. Subburaj, S. Murugavalli, “Survey on sign language recognition in context of vision-based and deep learning, Measurement: Sensors”, vol. 23, pp. 1-22 Oct 2022, 100385,ISSN 2665-9174

[20] Anuraganand Sharma,” Guided Stochastic Gradient Descent Algorithm for inconsistent datasets”, Applied Soft Computing”, vol. 73, Pages 1068-1080, 2018, ISSN 1568-4946

[21] S. Shrenika, M. Madhu Bala, "Sign Language Recognition Using Template Matching Technique," *in proc. 2020 International Conference on Computer Science, Engineering and Applications (ICCSEA)*, Gunupur, India, 03 July 2020, **DOI:** 10.1109/ICCSEA49143.2020.9132899

[22] Prof. Supriya Pawar, Sainath Muluk, Sourabh Koli "Real Time Sign Language Recognition using Python", *International Journal of Innovative Research in Computer and Communication Engineering, vol 6, issue 3, March* 2018.

[23] Sadaf Ikram , Namrata Dhanda ” American Sign Language using Convolutional Neural Network” *in proc. 2021 IEEE 4th International Conference on Computing , Power and Communication Technologies (GUCON)*,2021,Kuala Lumpur, Malaysia

[24] Kamal Preet Kour, Dr. Lini Mathew" Sign Language Recognition Using Image Processing", *International Journals of Advanced Research in Computer Science and Software Engineering*, vol. 7, Issue 8, 2017

[25] Neha Poddar, Shrushti Rao, Shruti Sawant, Vrushali Somavanshi, Sumita Chandak "Study of Sign Language Translation using Gesture Recognition", *International Journal of Advanced Research in Computer and Communication Engineering*, vol. 4, Issue 2, 2015, DOI 10.17148/IJARCCE.2015.4258

[26] D. Naglot, M. Kulkarni, "Real time sign language recognition using the leap motion controller," *in proc.* 2016 International Conference on Inventive Computation Technologies (ICICT), pp. 1-5, Coimbatore, India, 2016, doi:10.1109/INVENTIVE.2016.7830097.

[27] Sharvani Srivastava, Amisha Gangwar, Richa Mishra, Sudhakar Singh ,“Sign Language Recognition System using TensorFlow Object Detection API”, in proc. International Conference on Advanced Network Technologies and Intelligent Computing (ANTIC-2021), part of the book series ‘Communications in Computer and Information Science (CCIS)’, page 634-646, Feb 2022, Springer.: https://doi.org/10.1007/978-3-030-96040-7_48

[28] R. S. Trofin, C. Chiru, C. Vizitiu, A. Dinculescu, R. Vizitiu and A. Nistorescu, "Detection of Astronauts Speech and Language Disorder Signs during Space Missions using Natural Language Processing Techniques," in proc. *2019 E-Health and Bioengineering Conference (EHB)*, Iasi, Romania, 2019, pp. 1-4, doi: 0.1109/EHB47216.2019.8969950.

[29] S. S. Shanta, S. T. Anwar and M. R. Kabir, "Bangla Sign Language Detection Using SIFT and CNN," *in proc. 9th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, Bengaluru, India, 2018, pp. 1-6, doi: 10.1109/ICCCNT.2018.8493915.

[30] Bashar Saadoon Mahdi , Mustafa Jasim Hadi and Ayad Rodhan Abbas “Intelligent Security Model for Password Generation and Estimation Using Hand Gesture Features”, Big Data Cognitive. Computing. 6, 116, 2022, https://doi.org/10.3390/bdcc6040116