IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

ISO 3297:2007 Certified ⁙ Impact Factor 7.105 ⁙ Vol. 9, Issue 6, June 2022

DOI: 10.17148/IARJSET.2022.96109

Sign Language Detection system using artificial intelligence

Samarth S Naik[1], Rajeshwari N[2]

Dept. of MCA, student, Bangalore Institute of Technology, Bengaluru, India[1]

Dept. of MCA, Assistant Professor, Bangalore Institute of Technology, Bengaluru, India[2]

Abstract: The act of communicating or sharing information, knowledge, or emotions is defined as communication. All of the participants must understand and be able to comprehend a common language in order to identify communication between two or more persons. The methods of communication, however, differ in the case of dumb and deaf persons. The inability to hear is referred to as deafness, whereas deafness is the inability to communicate. They interact vocally among themselves and with others, but most people do not believe signing to be vital. Communication between a regular person and someone who is dumb or deaf is difficult since not everyone has linguistic skill or comprehension. One can create the model aided by machine learning to get beyond this obstacle. The model will be taught to recognise and translate various linguistic gestures. Many individuals will benefit from this while engaging and communicating with deaf and dumb persons. The single- and double-handed signs used in these Indian Sign Language Recognition systems are constructed utilizing machine learning methods, they are not, however, real-time. We present a method for building an Indian language dataset using a camera in this study. followed by the training of a TensorFlow model utilizing transfer learning to develop a real-time linguistic communication recognition system. Even with a little dataset, the algorithm manages to attain an honest degree of accuracy.

I. INTRODUCTION

The act of conveying information from one location, person, or group to another is sometimes referred to as communication. It has three components: -the speaker, the message being communicated, and the audience. It is typically judged successful only after the audience hears and understands the speaker's intended message. It is commonly classified as informal and formal communication, spoken (distance and face-to-face), written, nonverbal, gossip, feedback, and communication, and, as a result, active listening.
Nonverbal communication helps almost completely illiterate people communicate with one another and with others. While being dumb is an obstruction that diminishes one's capacity to communicate and renders them unsuitable to talk, being hard of hearing may be a disability that impairs one's ability to hear and renders them unfit to hear. It's challenging to organise correspondence with others when you can't choose whether to talk or listen. A person can express themselves without using words thanks to gesture-based communications, which are frequently used in this situation. However, there is a difficulty in that very few people are knowledgeable of gesture-based communication.
Due to a lack of knowledge about gesture-based communication, people who are
hearing-impaired may need to be able to communicate with one another by signing, but those who are challenged still find it difficult to converse with others who can
hear them. A technology-driven structure is frequently used to resolve this issue. Using such a method, one may easily translate motions used as communication tokens into English, the language most commonly used for communication.

The present framework for Indian Sign Language Recognition was developed using MATLAB and AI computations [4]. Creators spoke about being bold and made two moves. They used the Back Propagation Algorithm and the K Nearest Neighbors Algorithm to organize their system.

Their framework achieved an accuracy of 93-96%. To be quite clear, it's not even close to an ongoing SLR framework. The goal of this study is to create a continuous SLR framework utilizing TensorFlow's article location API and train it on webcam-generated datasets. The remainder of this article is organized as follows after the presentation. Area 2 presents the associated work on the SLR framework. Segment 3 shows the age and level of expertise. Segment 4 focuses on the creation process of the framework. The trial evaluation of the framework is presented in Area 5, and the article is finally concluded in Section 6.

DOI: 10.17148/IARJSET.2022.96109

II. RELATED WORK

Their framework has a 93-96 percent accuracy rate. To be clear, it is not even close to a functioning SLR framework. The goal of this study is to create a continuous SLR framework utilising TensorFlow's article location API and train it on webcam-generated datasets. The remainder of this article is organised as follows after the presentation. Area 2 presents the associated work on the SLR framework. The third segment displays the age and level of skill. The fourth segment focuses on the framework's creation process. The framework's trial evaluation is described in Area 5, and the article is eventually ended in Section 6.

A coordinated array of hand gestures with clear connotations that are used by people who are meeting weaker people to communicate in daily life are referred to as signing communications [3]. Because they are visual dialects, they employ and, facial, and body motions as communication tools. SLR demonstrates how to communicate via gestures without even realising it. It notices movement and interprets it into a language that is widely spoken, like English, etc.

SLR has been the subject of extensive investigation, but there is still much more to be done in this vast area. Thanks to AI techniques, the electronic frameworks are able to demand judgments based on interaction, for instance, information. The arrangement computations need two datasets: a testing dataset and a preparation dataset. Once the classifier has gained experience from the preparation set, the model is tested [6]. Several innovators have created efficient information order and purchasing mechanisms [3][7]. Vision-based methodologies and instantaneous estimation techniques are widely used to categorise past research on information security solutions [3].

For the instant estimation processes, movement information gloves, movement recording frameworks, or sensors are needed.

It is possible to accurately monitor hands, fingers, and other body parts because to the division of the movement information, which has prompted the creation of complex SLR systems.

Two datasets are required for the arrangement calculations: a testing dataset and a preparation dataset. The testing set is used to test the model once the classifier has experience from the preparation set [6]. Effective information order and purchase techniques have been developed by several inventors [3][7]. Past research on information security strategies is frequently divided into two categories: vision-based approaches and instantaneous estimating techniques [3]. The movement information gloves, movement capturing frameworks, or sensors are required for the instant estimation procedures.

It is possible to accurately monitor hands, fingers, and other body parts because to the division of the movement information, which has prompted the creation of complex SLR systems.

The authors used filtering techniques like Kalman and molecular channels to achieve precise and robust hands following, particularly in situations of deterrents [10][12]. Various devices should be used for information acquisition using either the instantaneous estimation or the vision-based techniques. The camera is the primary device used as an information expert cess in SLR systems [13].

A number of tools are utilised to collect data, including the Microsoft Kinect, which combines video transmission for depth and variety. Knowledge that is insightful helps to create division. In addition to technology, tactile gloves and accelerometers are alternative techniques for information collecting. Leap Motion Controller (LMC) is another information security framework [14]. [15] - It was developed by "Jump Motion," a San Francisco-based innovation firm that is now known as "Ultraleap." It can recognise and track hands, fingers, and things that resemble the opposite of the same digit while processing 200 instances per second. While it might be challenging to locate a gesture-based communication dataset, the majority of professionals record their preparation dataset from their underwriter [2].

Different handling methods or application frameworks have produced results with varying degrees of accuracy. The Light-HMM produced an accuracy result of 83.6
percent, the MSHMM of 86.7 percent, the SVM of 97.5 percent, the Eigen Value of 97 percent, and the Wavelet Family of 100 percent [2][31][22][32]. Although many models have given results with high accuracy, the precision is dependent on a number of factors, such as the dataset's size, the photos in the dataset's clarity based on information security approaches, the devices employed, and so on.

There are 2 types of SLR systems: continuous SLR and isolated SLR. A single gesture in isolated SLR is taught to the system.

Each picture is identified as standing for a letter of the alphabet, a number, or a particular motion. In contrast to single gesture categorization, continuous SLR is continuous. Continuous SLR enables the system to recognise and translate entire words rather than just a single gesture [33][34]. Despite all of the SLR research that has been done, there are still numerous gaps that need to be filled by more study. The following are some of the problems and obstacles that need to be resolved: [33][2][4][6].

• Since expensive data collecting equipment is required, commercialising SLR systems will require a less expensive

DOI: 10.17148/IARJSET.2022.96109

approach.
• Web cameras are an alternative to cameras with greater specifications, but the quality is reduced since the image is blurry.
• There are other problems with data collecting from sensors, such as noise, poor human handling, poor ground connection, etc. When hands and fingers are covered, vision-based algorithms can detect mistakes; however, large datasets are not yet accessible.
• There are myths regarding signing, such as the idea that sign language is universal, despite the fact that signing is dependent on
spoken language.
• Continuous SLR techniques use SLR isolated systems as building blocks, with non-trivial temporal segmentation as the pre-processing step and sentence synthesis as the post-processing step.
• Each word must undergo laborious labelling using isolated SLR approaches.

III. DATA ACQUISTION

For the Indian language, a real-time language identification system is being created. Python and OpenCV are used to collect webcam photos for data gathering. Real-time computer vision is the primary focus of the functionalities offered by OpenCV. It speeds up the incorporation of AI into commercial products and provides a solid foundation for computer vision applications. The OpenCV library contains more than 2500 powerful computer vision and machine learning algorithms that can be applied to a wide range of tasks, such as face detection and recognition, object identification, categorization of human actions, tracking camera and object movements, extracting 3D object models, and much more [35].
The produced dataset is composed of images that, as seen in Fig. 1, reflect the Indian alphabets used for communication [36]. To create the dataset, 25 photos for each letter of the alphabet are taken. In order to go from the sign of one alphabet to the sign of a particular alphabet, a clear stage of five seconds is presented between the two signs. Every 2 seconds, images are taken, enabling opportunity to record gestures with a minute difference in time. The photographs that were collected are kept in the appropriate folder.

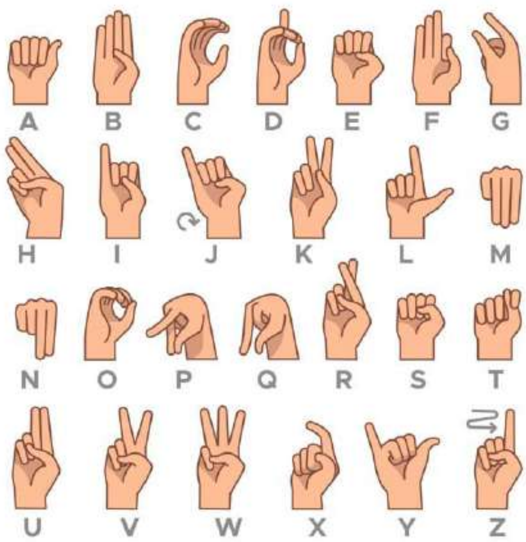


Fig. 1. Indian Sign Language Alphabets

For data collection, dependencies such as cv2, i.e., Time, OS, OpenCV, and uuid have been imported. The dependency Os is used to help with file path work. It is a common Python utility module that offers routines for interfacing with operating systems. Thanks to the time module, time in Python may be represented in code in a number of forms, including strings, integers, and objects. It may be used to assess code efficiency or wait during code execution in addition to representing time. It is used here to insert gaps between picture captures to allow for hand motions. The uuid library is used to name the image files. It aids in the creation of random objects of 128 bits as ids, which provide uniqueness due to the use of time and computer hardware.

This work is licensed under a Creative Commons Attribution 4.0 International License

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

ISO 3297:2007 Certified ⁘ Impact Factor 7.105 ⁘ Vol. 9, Issue 6, June 2022

DOI: 10.17148/IARJSET.2022.96109

Fig. 2. Selecting a portion of the image to label

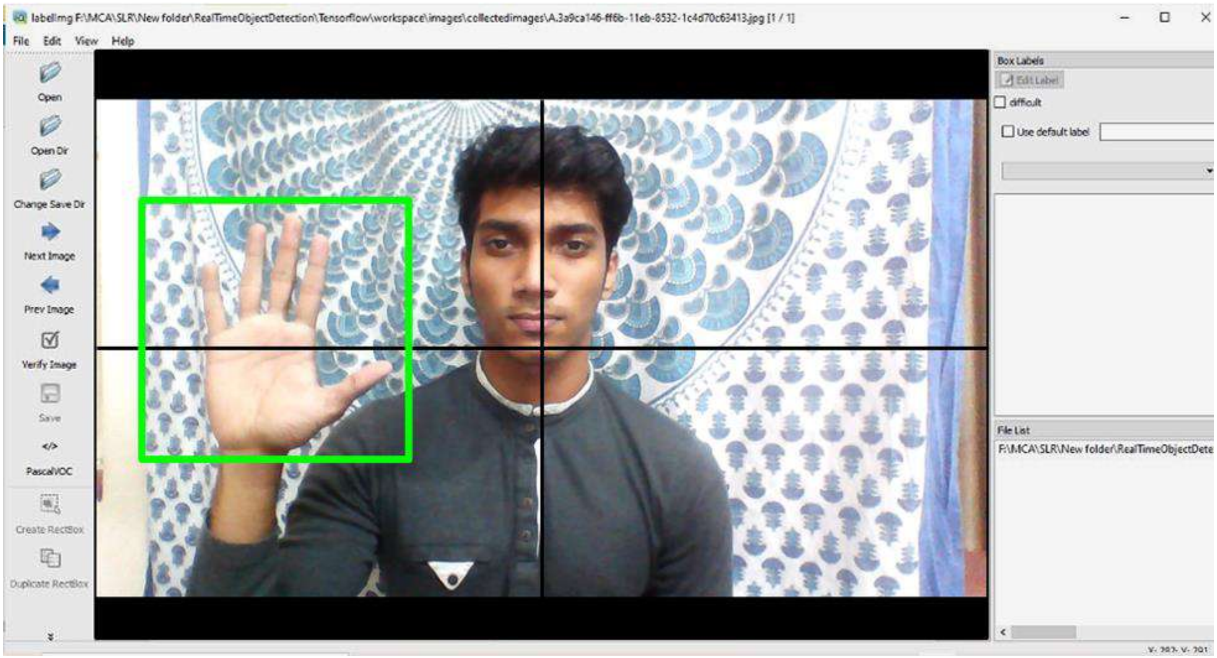


Fig. 3. Labelling the selected portion

After all of the photos have been acquired, they are labelled one by one with the LabelImg package. LabelImg is a free open-source program for labelling pictures visually. As illustrated in Fig. 1, the hand gesture section of the image is labelled by what the gesture in the box or the symbol indicates. 2 as well as Fig. 3. When you save a labelled image, an XML file is produced. The XML files contain all of the picture information, including the tagged part. Once the images have been labelled, the XML files for all of them are now available. This is used to create TensorFlow (TF) records. The images and their XML files are then divided into training and validation data in an 80:20 ratio. Twenty (20%) of the 25 photos of an alphabet were taken and saved as a training dataset, while the remaining 5 (20%) were taken and stored as a validation dataset. This work was completed for all 26 alphabet pictures.

IARJSET

ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

ISO 3297:2007 Certified ⁜ Impact Factor 7.105 ⁜ Vol. 9, Issue 6, June 2022

DOI: 10.17148/IARJSET.2022.96109

IV. METHODOLOGY

The suggested system is designed to create a real-time sign language detector using a TensorFlow object identification API and train it on the given dataset using transfer learning [37]. Images from a camera are acquired for data acquisition using Python and OpenCV in accordance with the steps outlined in Section 3. Following data collection, a labelled map is created that represents all of the model's objects and contains their id as well as the label for each letter of the alphabet. Each of the 26 labels on the label map corresponds to a letter of the alphabet. Each label has a unique ID that ranges from 1 to 26. To seek up the class name, use this as a reference. The TensorFlow object identification API is then trained by using generate TF record, which generates TF records from testing and training data TensorFlow uses a binary storage format called TF record. The use of binary files for data storage has a substantial influence on the performance of the import pipeline and, as a result, the model's training time. It copies quickly, uses less disc space, and reads data from the drive effectively. The open-source framework TensorFlow object detection API makes it straightforward to develop, train, and utilise an object identification model. The open-source framework TensorFlow object detection API makes it straightforward to develop, train, and utilise an object identification model. They provide a framework known as TensorFlow detection model zoo, which has a collection of detection models that have previously been trained on the COCO 2017 dataset. The SSD MobileNet v2 Object recognition model is coupled with the FPN-lite feature extractor, shared box predictor, and focal loss on training images scaled to 320x320. Pipeline setup is the process of setting up and changing the pre-trained model's configuration for transfer learning in order to train it using the freshly created dataset. TensorFlow, config util, pipeline pb2, and text format have been imported as configuration dependencies.

The most notable change is that the number of classes has been reduced from 90 to 26, as has the number of signs (alphabets) on which the model will be trained. The model was trained in 12000 steps after the settings was set up and updated. The model's training step count was determined by the hyper-parameter employed during training, which was set at 12,000. The model suffers from classification loss, regularisation loss, and localization loss during training.The predicted bounding box correction and the real values differ on the localization loss. The localization loss [38] formula is provided by Equations (1)-(5).

$$L_{loc}(x,l,g) = \sum_{i \in Pos}^{N} \sum_{m \in \{cx,cy,w,h\}} x_{ij}^{k} \, smooth_{L1}\left(l_i^m - \hat{g}_j^m\right)$$

$$\hat{g}_j^{cx} = (g_j^{cx} - d_i^{cx})/d_i^{w} \qquad (2)$$

$$\hat{g}_j^{cy} = (g_j^{cy} - d_i^{cy})/d_i^{h} \qquad (3)$$

$$\hat{g}_j^{w} = \log(g_j^{w}/d_i^{w}) \qquad (4)$$

$$\hat{g}_j^{h} = \log(g_j^{h}/d_i^{h}) \qquad (5)$$

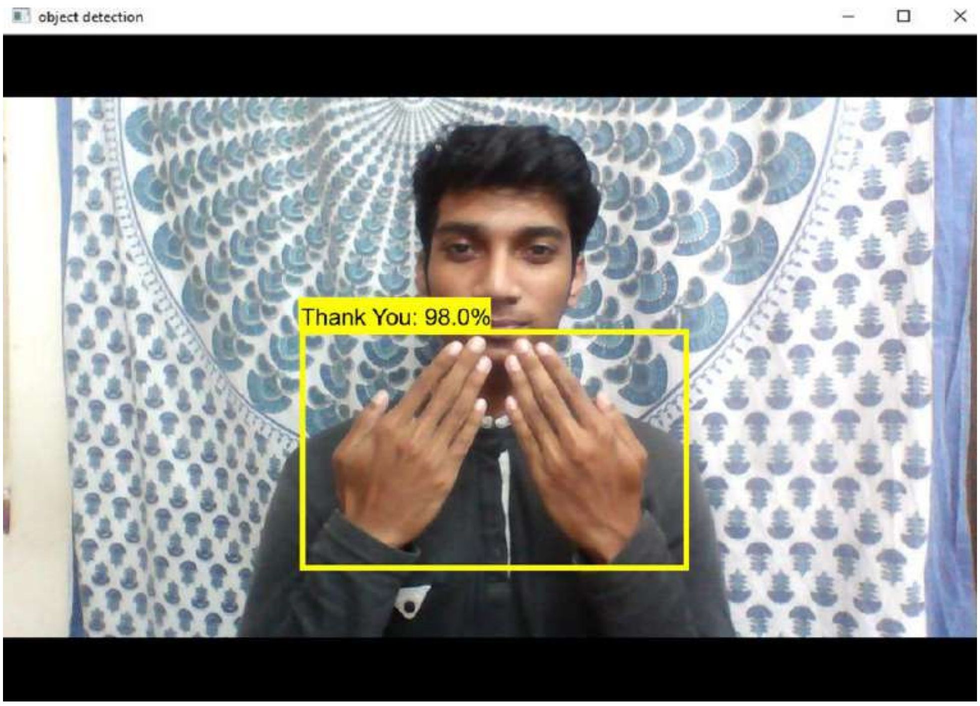


Fig. 4. Real-Time Sign Language Detection

DOI: 10.17148/IARJSET.2022.96109

l corresponds to the expected bounding box, g represents the ground truth bounding box, and $\hat{g}$ represents the boundary box of the encoded ground truth. and x_{ij}^{k} is the pointer of correspondence between category k's ground truth box j and default box i. N is the number of predefined boxes of default boxes that have been matched.
The softmax loss over several classes is referred to as the classification loss. The classification loss formula [38] is Eq. (6).

$$L_{lcon\,f}(x,c) = \sum_{i\in Pos}^{N} x_{ij}^{p} \log\left(\hat{c}_i^{p}\right) - \sum_{i\in Neg} \log(\hat{c}_i^{0})$$

Where, Softmax enabled class score for category p with default box i is, $\hat{c}\,_i^{p} = \exp\,(c_i^p)/\sum_p exp\,(c_i^p\,)$, x_{ij}^{p} is the indication that corresponds to the default box i and the ground truth box j of category p. The next section discusses the many losses that occurred throughout the experiment. The model is loaded from the most recent checkpoint after training, allowing for real-time detection. The model will be prepared for training after configuration setup and updating. The most recent checkpoint, which is made during model training, is used to load the learned model. The model has now been completed and is capable of detecting sign language in real time. The camera and OpenCV are used again for real-time detection. For real-time detection, cv2 and NumPy dependencies are employed.
As seen in Figure. 5, the system recognises signals in real time and translates what each movement means into English. The system
is checked on a regular basis by creating and presenting different indications. Each sign's level of assurance, or how certain the system is about detecting a sign (alphabet), is inspected, documented, and tallied for the output.

V. EXPERIMENT

The dataset was built for the Indian language, with the signs represented by English alphabets. The dataset is constructed following the process provided in Section 3 of this paper for gathering information.
The tests were carried out using a Windows 10 PC equipped with an Intel i5 7th generation 2.70 GHz CPU, 8 GB of RAM, and an HP True Vision HD camera with a 0.31 MP and 640x480 resolution for the webcam. TensorFlow Object Detection API, Jupyter Notebook, OpenCV (version 4.2.0), and Python are used in the development environment (version 3.7.3).

VI. RESULT AND DISCUSSION

The Tensor Flow Object Detection API is used in this system. This developed system is capable of recognising Indian signing alphabets in real time. SSD MobileNet v2 320x320 is a pre-trained model obtained from the TensorFlow model zoo. Transfer learning was used to train the newly created dataset, which comprises of 650 photographs in total, 25 images for each word.
According to Fig. 4, The overall loss sustained during the final 10,000 step training phase was 0.25; the localization loss was 0.18; the classification loss was 0.13; and the regularisation loss was 0.10. Fig. 4 further shows that step 9900 resulted in a loss of just 0.17.
The outcome of the framework was 85.45%, based on the confidence rate and normal certainty speed of the framework. The knowledge rate is noted and categorised in the results, which are shown in Table 1, for each letter in order. By increasing the dataset size, the knowledge pace of the framework may be increased, which will improve the framework's recognition capability. Performing on the framework's outcome and enhancing it as a result.

Table 1. Confidence rate of alphabets

A	94%	J	58%	S	90%
B	98%	K	88%	T	83%
C	90%	L	95%	U	88%
D	95%	M	94%	V	87%
E	70%	N	55%	W	88%
F	96%	O	78%	X	89%
G	73%	P	95%	Y	93%
H	97%	Q	95%	Z	84%
I	95%	R	83%		

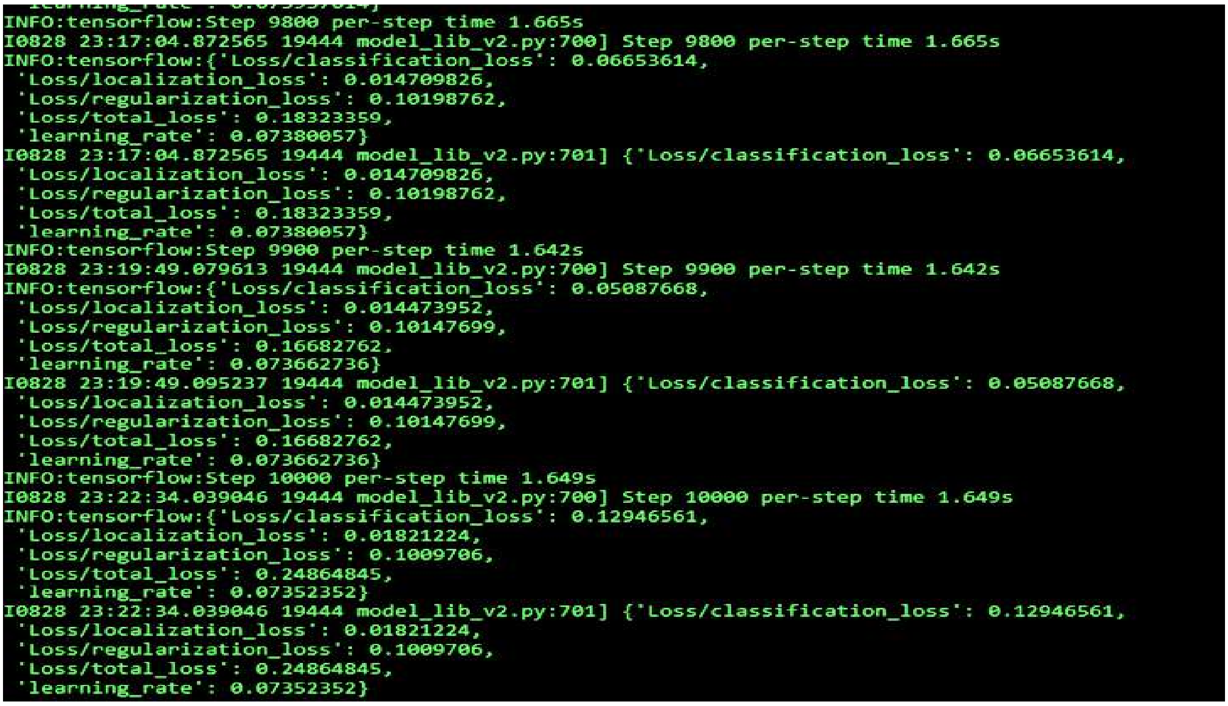
 This work is licensed under a Creative Commons Attribution 4.0 International License

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

ISO 3297:2007 Certified ⁘ Impact Factor 7.105 ⁘ Vol. 9, Issue 6, June 2022

DOI: 10.17148/IARJSET.2022.96109



```
INFO:tensorflow:Step 9800 per-step time 1.665s
I0828 23:17:04.872565 19444 model_lib_v2.py:700] Step 9800 per-step time 1.665s
INFO:tensorflow:{'Loss/classification_loss': 0.06653614,
 'Loss/localization_loss': 0.014709826,
 'Loss/regularization_loss': 0.10198762,
 'Loss/total_loss': 0.18323359,
 'learning_rate': 0.07380057}
I0828 23:17:04.872565 19444 model_lib_v2.py:701] {'Loss/classification_loss': 0.06653614,
 'Loss/localization_loss': 0.014709826,
 'Loss/regularization_loss': 0.10198762,
 'Loss/total_loss': 0.18323359,
 'learning_rate': 0.07380057}
INFO:tensorflow:Step 9900 per-step time 1.642s
I0828 23:19:49.079613 19444 model_lib_v2.py:700] Step 9900 per-step time 1.642s
INFO:tensorflow:{'Loss/classification_loss': 0.05087668,
 'Loss/localization_loss': 0.014473952,
 'Loss/regularization_loss': 0.10147699,
 'Loss/total_loss': 0.16682762,
 'learning_rate': 0.073662736}
I0828 23:19:49.095237 19444 model_lib_v2.py:701] {'Loss/classification_loss': 0.05087668,
 'Loss/localization_loss': 0.014473952,
 'Loss/regularization_loss': 0.10147699,
 'Loss/total_loss': 0.16682762,
 'learning_rate': 0.073662736}
INFO:tensorflow:Step 10000 per-step time 1.649s
I0828 23:22:34.039046 19444 model_lib_v2.py:700] Step 10000 per-step time 1.649s
INFO:tensorflow:{'Loss/classification_loss': 0.12946561,
 'Loss/localization_loss': 0.01821224,
 'Loss/regularization_loss': 0.1009706,
 'Loss/total_loss': 0.24864845,
 'learning_rate': 0.07352352}
I0828 23:22:34.039046 19444 model_lib_v2.py:701] {'Loss/classification_loss': 0.12946561,
 'Loss/localization_loss': 0.01821224,
 'Loss/regularization_loss': 0.1009706,
 'Loss/total_loss': 0.24864845,
 'learning_rate': 0.07352352}
```

Fig. 5. Loss incurred at different steps

The cutting-edge technique for the Indian signature Recognition framework achieved 93-96 percent accuracy. Despite being highly precise, it is not a continuous SLR framework. This research looks at the issue. Despite the little dataset, our approach managed to achieve a conventional confidence rate of 85.45 percent.

For the Indian language, a real-time language identification system is being created. Python and OpenCV are used to collect webcam photos for data gathering. Real-time computer vision is the primary focus of the functionalities offered by OpenCV. It accelerates the deployment of artificial intelligence in commercial products and provides a reliable infrastructure for computer vision applications. The OpenCV package comprises over 2600 powerful computer vision and machine learning algorithms that may be used for a wide range of tasks such as face and object recognition, object identification, action categorization, tracking camera and object motions, and extracting 3D object models [35].

The produced dataset is composed of images that, as seen in Fig. 1, reflect the Indian alphabets used for communication [36]. To create the dataset, 26 photos are taken for each letter of the alphabet. The images are taken every two seconds, giving time to record gestures with a minor change, after five seconds, a transparent stage of five seconds is supplied between two unique signs, i.e., to transition from the sign of one alphabet to the sign of a specific alphabet. The collected photos are saved in the relevant folder.

VII. CONCLUSION AND FUTURE WORK

A form of visual language called sign language makes use of facial, body, and hand gestures to convey messages. The use of technology enables persons with special needs to interact, communicate, and share their experiences, emotions, ideas, and thoughts with others. The disadvantage of this is that it is not a language that the majority of people are ready to learn, which restricts how widespread it can be and serves as a barrier to communication. This problem may currently be remedied by using automatic sign language recognition technology, which can properly translate designated language gestures into voice communication.

It was carried out in this research using the TensorFlow object detection API. The technique was developed using a dataset of Indian language alphabets. The technology instantly recognises signature. Python and OpenCV are used to acquire data while utilising a camera to collect images, which lowers the cost. The developed strategy

had a mean confidence rating of 85.45 percent. Despite the system's high average confidence rate, the dataset on which it was trained is tiny and confined. The algorithm will be able to recognise more motions in the future as the dataset grows. The model that has been utilised, TensorFlow, is frequently replaced for another model. By altering the dataset, the system is often developed for other sign languages.

DOI: 10.17148/IARJSET.2022.96109

REFERENCES

1. Kapur, R.: The Types of Communication. MIJ. 6, (2020).
2. Suharjito, Anderson, R., Wiryana, F., Ariesta, M.C., Kusuma, G.P.: Sign Language Recognition Application Systems for Deaf-Mute People: A Review Based on InputProcess-Output. Procedia Comput. Sci. 116, 441–448 (2017). https://doi.org/10.1016/J.PROCS.2017.10.028.
3. Konstantinidis, D., Dimitropoulos, K., Daras, P.: Sign language recognition based on hand and body skeletal data. 3DTV-Conference. 2018-June, (2018). https://doi.org/10.1109/3DTV.2018.8478467.
4. Dutta, K.K., Bellary, S.A.S.: Machine Learning Techniques for Indian Sign Language Recognition. Int. Conf. Curr. Trends Comput. Electr. Electron. Commun. CTCEEC 2017. 333–336 (2018). https://doi.org/10.1109/CTCEEC.2017.8454988.
5. Bragg, D., Koller, O., Bellard, M., Berke, L., Boudreault, P., Braffort, A., Caselli, N., Huenerfauth, M., Kacorri, H., Verhoef, T., Vogler, C., Morris, M.R.: Sign Language Recognition, Generation, and Translation: An Interdisciplinary Perspective. 21st Int. ACM SIGACCESS Conf. Comput. Access. (2019). https://doi.org/10.1145/3308561.
6. Rosero-Montalvo, P.D., Godoy-Trujillo, P., Flores-Bosmediano, E., Carrascal-Garcia, J., 12 Otero-Potosi, S., Benitez-Pereira, H., Peluffo-Ordonez, D.H.: Sign Language Recognition Based on Intelligent Glove Using Machine Learning Techniques. 2018 IEEE 3rd Ecuador Tech. Chapters Meet. ETCM 2018. (2018). https://doi.org/10.1109/ETCM.2018.8580268.
7. Zheng, L., Liang, B., Jiang, A.: Recent Advances of Deep Learning for Sign Language Recognition. DICTA 2017 – 2017 Int. Conf. Digit. Image Comput. Tech. Appl. 2017- Decem, 1–7 (2017). https://doi.org/10.1109/DICTA.2017.8227483.
8. Rautaray, S.S.: A Real Time Hand Tracking System for Interactive Applications. Int. J. Comput. Appl. 18, 975–8887 (2011).
9. Zhang, Z., Huang, F.: Hand tracking algorithm based on super-pixels feature. Proc. - 2013 Int. Conf. Inf. Sci. Cloud Comput. Companion, ISCC-C 2013. 629–634 (2014). https://doi.org/10.1109/ISCC-C.2013.77.
10. Lim, K.M., Tan, A.W.C., Tan, S.C.: A feature covariance matrix with serial particle filter for isolated sign language recognition. Expert Syst. Appl. 54, 208–218 (2016). https://doi.org/10.1016/J.ESWA.2016.01.047.
11. Lim, K.M., Tan, A.W.C., Tan, S.C.: Block-based histogram of optical flow for isolated sign language recognition. J. Vis. Commun. Image Represent. 40, 538–545 (2016). https://doi.org/10.1016/J.JVCIR.2016.07.020.
12. Gaus, Y.F.A., Wong, F.: Hidden Markov Model - Based gesture recognition with overlapping hand-head/hand-hand estimated using Kalman Filter. Proc. - 3rd Int. Conf. Intell. Syst. Model. Simulation, ISMS 2012. 262–267 (2012). https://doi.org/10.1109/ISMS.2012.67.
13. Nikam, A.S., Ambekar, A.G.: Sign language recognition using image based hand gesture recognition techniques. Proc. 2016 Online Int. Conf. Green Eng. Technol. IC-GET 2016. (2017). https://doi.org/10.1109/GET.2016.7916786
14. Mohandes, M., Aliyu, S., Deriche, M.: Arabic sign language recognition using the leap motion controller. IEEE Int. Symp. Ind. Electron. 960–965 (2014).
15. Enikeev, D.G., Mustafina, S.A.: Sign language recognition through Leap Motion controller and input prediction algorithm. J. Phys. Conf. Ser. 1715, 012008 (2021).https://doi.org/10.1088/1742-6596/1715/1/012008.
16. Cheok, M.J., Omar, Z., Jaward, M.H.: A review of hand gesture and sign language recognition techniques. Int. J. Mach. Learn. Cybern. 2017 101. 10, 131–153 (2017). https://doi.org/10.1007/S13042-017-0705-5.
17. Wadhawan, A., Kumar, P.: Sign Language Recognition Systems: A Decade Systematic Literature Review. Arch. Comput. Methods Eng. 2019 283. 28, 785–813 (2019). https://doi.org/10.1007/S11831-019-09384-2.
18. Camgöz, N.C., Koller, O., Hadfield, S., Bowden, R.: Sign language transformers: Joint end-to-end sign language recognition and translation. Proc. IEEE Comput. Soc. Conf.Comput. Vis. Pattern Recognit. 10020–10030 (2020). https://doi.org/10.1109/CVPR42600.2020.01004.
19. Cui, R., Liu, H., Zhang, C.: A Deep Neural Framework for Continuous Sign Language Recognition by Iterative Training. IEEE Trans. Multimed. 21, 1880–1891 (2019). https://doi.org/10.1109/TMM.2018.2889563
20. Bantupalli, K., Xie, Y.: American Sign Language Recognition using Deep Learning and Computer Vision. Proc. - 2018 IEEE Int. Conf. Big Data, Big Data 2018. 4896–4899 13 (2019). https://doi.org/10.1109/BIGDATA.2018.8622141.
21. Hore, S., Chatterjee, S., Santhi, V., Dey, N., Ashour, A.S., Balas, V.E., Shi, F.: Indian Sign Language Recognition Using Optimized Neural Networks. Adv. Intell. Syst. Comput. 455, 553–563 (2017). https://doi.org/10.1007/978-3-319-38771-0_54.
22. Kumar, P., Roy, P.P., Dogra, D.P.: Independent Bayesian classifier combination-based sign language recognition using facial expression. Inf. Sci. (Ny). 428, 30–48 (2018). https://doi.org/10.1016/J.INS.2017.10.046.
23. Sharma, A., Sharma, N., Saxena, Y., Singh, A., Sadhya, D.: Benchmarking deep neural network approaches for Indian Sign Language recognition. Neural Comput. Appl. 2020 3312. 33, 6685–6696 (2020). https://doi.org/10.1007/S00521-020-05448-8.

DOI: 10.17148/IARJSET.2022.96109

24. Kishore, P.V.V., Prasad, M. V.D., Prasad, C.R., Rahul, R.: 4-Camera model for sign language recognition using elliptical fourier descriptors and ANN. Int. Conf. Signal
Process. Commun. Eng. Syst. - Proc. SPACES 2015, Assoc. with IEEE. 34–38 (2015). https://doi.org/10.1109/SPACES.2015.705828
25. Tewari, D., Srivastava, S.K.: A Visual Recognition of Static Hand Gestures in Indian Sign Language based on Kohonen Self-Organizing Map Algorithm. Int. J. Eng. Adv. Technol. 165 (2012).
26. Gao, W., Fang, G., Zhao, D., Chen, Y.: A Chinese sign language recognition system based on SOFM/SRN/HMM. Pattern Recognition 37, 2389–2402 (2004). https://doi.org/10.1016/J.PATCOG.2004.04.00
27. Quocthang, P., Dung, N.D., Thuy, N.T.: A comparison of SimpSVM and RVM for sign language recognition. ACM Int. Conf. Proceeding Ser. 98–104 (2017). https://doi.org/10.1145/3036290.3036322.
28. Pu, J., Zhou, W., Li, H.: Iterative alignment network for continuous sign language recognition. Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. 2019-June, 4160–4169 (2019). https://doi.org/10.1109/CVPR.2019.00429.
29. Kalsh, E.A., Garewal, N.S.: Sign Language Recognition System. Int. J. Comput. Eng. Res.
30. Singha, J., Das, K.: Indian Sign Language Recognition Using Eigen Value Weighted Euclidean Distance Based Classification Technique. IJACSA) Int. J. Adv. Comput. Sci. Appl. 4, (2013).
31. Liang, Z., Liao, S., Hu, B.: 3D Convolutional Neural Networks for Dynamic Sign Language Recognition. Comput. J. 61, 1724–1736 (2018). https://doi.org/10.1093/COMJNL/BXY049
32. Pigou, L., Van Herreweghe, M., Dambre, J.: Gesture and Sign Language Recognition with Temporal Residual Networks. Proc. - 2017 IEEE Int. Conf. Comput. Vis. Work. ICCVW 2017. 2018-Janua, 3086–3093 (2017). https://doi.org/10.1109/ICCVW.2017.365.
33. Huang, J., Zhou, W., Zhang, Q., Li, H., Li, W.: Video-based Sign Language Recognition without Temporal Segmentation.
34. Cui, R., Liu, H., Zhang, C.: Recurrent convolutional neural networks for continuous sign language recognition by staged optimization. Proc. - 30th IEEE Conf. Comput. Vis. Pattern Recognition, CVPR 2017. 2017-Janua, 1610–1618 (2017). https://doi.org/10.1109/CVPR.2017.175.
35. About - OpenCV.
36. Poster of the Manual Alphabet in ISL | Indian Sign Language Research and Training Center 14 (ISLRTC), Government of India.
37. Transfer learning and fine-tuning | TensorFlow Core.
38. Wu, S., Yang, J., Wang, X., Li, X.: IoU-balanced Loss Functions for Single-stage Object Detection. (2020).