



e-ISSN: 2319-8753 | p-ISSN: 2347-6710

# IJRSET

International Journal of Innovative Research in  
**SCIENCE | ENGINEERING | TECHNOLOGY**



# INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

Volume 13, Issue 4, April 2024

**ISSN** INTERNATIONAL  
STANDARD  
SERIAL  
NUMBER  
INDIA

Impact Factor: 8.423

9940 572 462

6381 907 438

[ijirset@gmail.com](mailto:ijirset@gmail.com)

[www.ijirset.com](http://www.ijirset.com)

# Sign Language Recognition Using Deep Learning SSD

K. Jaya Sri Lakshmi Bai<sup>1</sup>, P. Charishma<sup>2</sup>, H. Sai Harshini<sup>3</sup>, J. Yaswanth<sup>4</sup>, A. Satwik Chand<sup>5</sup>,  
O.T. Gopi Krishna<sup>6</sup>

U.G. Student, Department of Computer Engineering, Vasireddy Venkatadri Institute of Technology, Nambur,  
Andhra Pradesh, India<sup>1,2,3,4,5</sup>

Associate Professor, Department of Information Technology, Vasireddy Venkatadri Institute of Technology, Nambur,  
Andhra Pradesh, India<sup>6</sup>

**ABSTRACT:** Sign language serves as a vital mode of communication for individuals with hearing impairments. Automated recognition of sign gestures through computer vision techniques has gained significant attention, as it enables seamless interaction between the hearing-impaired and the hearing world. This paper presents a comprehensive approach for sign language recognition using the deep learning algorithm Single Shot Multibox Detection (SSD) integrated with MobileNetV2 and is implemented in Python.

**KEYWORDS:** American signlanguage recognition (ASL), Single Shot Multibox Detector(SSD), MobileNetV2, Deep Learning, Object Detection.

## I. INTRODUCTION

Sign language is the only way of communication for the deaf and hard of hearing community, they have a rich set of signs that help them convey their thoughts to the real world. For individuals with hearing problems, sign language recognition systems play a vital role in bridging communication barriers and facilitating meaningful interactions in various domains, including education, healthcare, and social interactions.

There are many sign languages all around the world, each sign language has some common gestures like thank you, hello, I love you. For our project we are using American sign language gestures because of their easily adaptable and understandable signs. They are easy to understand and quick to learn.

The role of the deep learning algorithm also plays a vital role here Traditional approaches to sign language interpretation are often done on manual transcription or by human interpreters, this leads to so many problems rise like miscommunication, delay in finding a interpreter etc. so we are using deep learning algorithms, particularly the Single Shot Multibox Detector (SSD), this algorithm works on the detection of signs done by the user within no time. SSD's ability to perform rapid object detection and classification makes it well-suited for applications requiring speed, accuracy, and efficiency. This system could enhance and return a accurate detection of the gesture.

Many people around the world tend to learn and use ASL because of its vast range of gestures One of the primary reasons is the desire to communicate effectively with people having problems with hearing and speaking. Sign language provides a means for people with hearing problems to express themselves better.

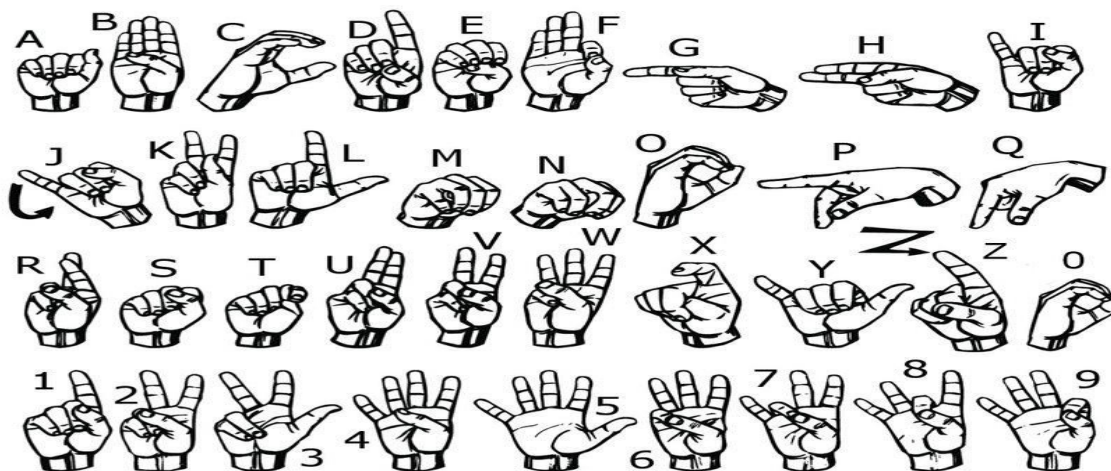


Figure-1: American Sign Language Alphabets and Numbers

## II. LITERATURE SURVEY

Sign language recognition (SLR) plays a crucial role in bridging communication gaps for the deaf and hard of hearing communities. Traditional SLR systems often face challenges in real-time processing and accuracy. To address these issues, researchers have explored deep learning techniques, including the Single Shot Multibox Detector (SSD), for improving SLR systems.[1] Liu et al. (2020): Multi-User Real-time SLR with Transfer Learning - Liu et al. proposed a multi-user real-time SLR system using transfer learning. Their work emphasizes the importance of real-time recognition, which aligns with the requirements of many practical applications. By leveraging transfer learning, their system achieves competitive accuracy while reducing training time and resource requirements.[2] Tolentino et al. (2019): Static SLR Using Deep Learning - Tolentino et al. focused on static SLR using deep learning techniques. Their research showcases the effectiveness of deep learning in recognizing static signs, which are fundamental components of sign language. This work highlights the potential for deep learning to enhance the accuracy and efficiency of SLR systems.

[4]. Bantupalli & Xie (2018): SLR with Deep Learning and Computer Vision Bantupalli and Xie presented an SLR system leveraging deep learning and computer vision methodologies. Their work demonstrates the synergy between these two domains in improving SLR accuracy and robustness. By combining deep learning models with computer vision techniques, they achieve significant advancements in SLR technology. [5]. Kothadiya et al. (2022): Deepsign - SLR Using Deep Learning Kothadiya et al. introduced "Deepsign," a comprehensive SLR system based on deep learning. Their system, built upon the SSD architecture, excels in both sign language detection and recognition tasks. By employing deep learning techniques, Deepsign offers state-of-the-art performance in SLR, making significant contributions to the field.[6]. Lin et al. (2021): Improved YOLOv5 Algorithm for SLR [1] Lin et al. proposed an improved YOLOv5 algorithm specifically tailored for SLR applications. Their work focuses on enhancing recognition accuracy and efficiency, addressing key challenges faced by existing SLR systems. The improved YOLOv5 algorithm showcases promising results in real-time SLR tasks.[7]. Al-Hammadi et al. (2020): Efficient SLR with Deep Learning Al-Hammadi et al. developed an efficient SLR approach using deep learning techniques. Their emphasis on efficient hand gesture representation contributes to faster and more accurate SLR systems. By optimizing model architectures and training strategies, they achieve a balance between performance and computational efficiency.

In Conclusion, The literature review highlights the diverse approaches and methodologies adopted in SLR using SSD and other deep learning techniques. From real-time multi-user recognition to efficient hand gesture representation, researchers continue to innovate and improve SLR systems, addressing key challenges and paving the way for more accessible communication for the deaf and hard of hearing communities.



### III. PROPOSED METHODOLOGY AND DISCUSSION

The proposed approach integrates the Single Shot Multibox Detector (SSD) algorithm with the MobileNetV2 architecture to enhance sign language recognition systems. This combination aims to address the limitations often encountered in CNN-based systems while providing accurate and real-time gesture recognition capabilities. MobileNetV2's lightweight design and efficient operations make it well-suited for deployment on resource-constrained devices, reducing the computational load and leading to faster inference times for real-time gesture recognition. SSD's multi-scale object detection capability complements MobileNetV2 by efficiently handling the diverse range of hand shapes and movements in sign language gestures. The integration of SSD with MobileNetV2 improves accuracy in gesture recognition by capturing fine-grained details and variations. This approach enables real-time interaction between users and the sign language recognition system, crucial for applications requiring immediate feedback. Additionally, the scalability and ease of deployment on various devices make the combined SSD-MobileNetV2 approach accessible and usable across different platforms, contributing to inclusive technology for the deaf and hard of hearing community. Previous approaches to sign language recognition are done in many ways, however they are not really so accurate and hard to please we went forward with these machine learning algorithms to deep learning algorithms. These deep learning algorithms usually have 4-5 layers of processing the given input which helps us to have the data set totally satisfied we are moving forward with the increasing new technology. For that we need to have shown strengths in terms of accuracy, especially when trained on large and diverse datasets.

However, previous approaches have some limitations, such as:

- Data scarcity
- Computational complexity
- Generalization

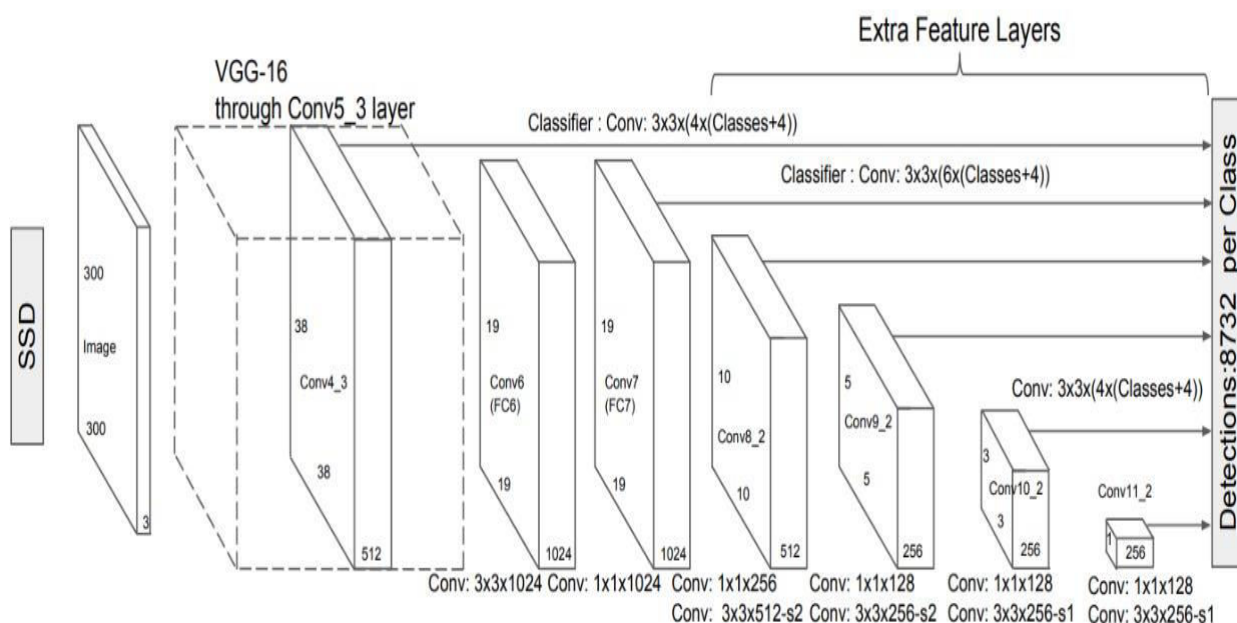


Figure -2 Proposed system architecture



This is the where we got our database and model form:

```
@misc{
  american-sign-language-letters-gxpdm_dataset,
  title = {American sign language letters Dataset},
  type = {Open Source Dataset},
  author = {Sign Language},
  howpublished = {
\url{ https://universe.roboflow.com/sign-language-oyykt/american-sign-language-letters-gxpdm
} },
  url = {
https://universe.roboflow.com/sign-language-oyykt/american-sign-language-letters-gxpdm
},
  journal = {Roboflow Universe},
  publisher = {Roboflow},
  year = {2024},
  month = {mar},
  note = {visited on 2024-04-05},
}
```

#### IV. EXPERIMENTAL RESULTS

The dataset we used in this project is from roboflow[3] that gave the trained model that helped us to get the accurate results. Coming to the model configuration the model we used is “ssdlite\_mobilenet\_v2\_coco\_2018\_05\_09” that trained the model for accurate results.

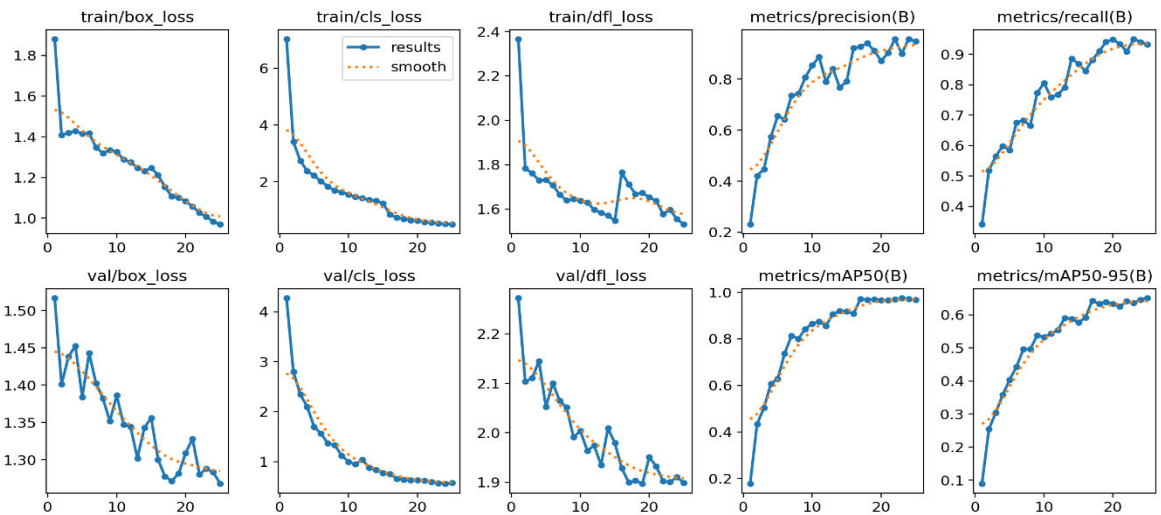
```
Downloading https://ultralytics.com/assets/Arial.ttf to 'C:\Users\lenovo1\AppData\Roaming\Ultralytics\Arial.ttf'...
100%|██████████| 755k/755k [00:00<00:00, 6.84MB/s]
Overriding model.yaml nc=80 with nc=26

   from  n  params module                                arguments
   ----  -  -
0         -1  1    1856 ultralytics.nn.modules.conv.Conv                    [3, 64, 3, 2]
1         -1  1    73984 ultralytics.nn.modules.conv.Conv                    [64, 128, 3, 2]
2         -1  3    279808 ultralytics.nn.modules.block.C2f                    [128, 128, 3, True]
3         -1  1    295424 ultralytics.nn.modules.conv.Conv                    [128, 256, 3, 2]
4         -1  6    2101248 ultralytics.nn.modules.block.C2f                    [256, 256, 6, True]
5         -1  1    1180672 ultralytics.nn.modules.conv.Conv                    [256, 512, 3, 2]
6         -1  6    8396800 ultralytics.nn.modules.block.C2f                    [512, 512, 6, True]
7         -1  1    2360320 ultralytics.nn.modules.conv.Conv                    [512, 512, 3, 2]
8         -1  3    4461568 ultralytics.nn.modules.block.C2f                    [512, 512, 3, True]
9         -1  1    656896 ultralytics.nn.modules.block.SPPF                    [512, 512, 5]
10        -1  1         0 torch.nn.modules.upsampling.Upsample                [None, 2, 'nearest']
11        [-1, 6] 1         0 ultralytics.nn.modules.conv.Concat                    [1]
12        -1  3    4723712 ultralytics.nn.modules.block.C2f                    [1024, 512, 3]
13        -1  1         0 torch.nn.modules.upsampling.Upsample                [None, 2, 'nearest']
14        [-1, 4] 1         0 ultralytics.nn.modules.conv.Concat                    [1]
15        -1  3    1247744 ultralytics.nn.modules.block.C2f                    [768, 256, 3]
16        -1  1    590336 ultralytics.nn.modules.conv.Conv                    [256, 256, 3, 2]
17        [-1, 12] 1         0 ultralytics.nn.modules.conv.Concat                    [1]
18        -1  3    4592640 ultralytics.nn.modules.block.C2f                    [768, 512, 3]
19        -1  1    2360320 ultralytics.nn.modules.conv.Conv                    [512, 512, 3, 2]
20        [-1, 9] 1         0 ultralytics.nn.modules.conv.Concat                    [1]
21        -1  3    4723712 ultralytics.nn.modules.block.C2f                    [1024, 512, 3]
22        [15, 18, 21] 1    5602846 ultralytics.nn.modules.head.Detect                    [26, [256, 512, 512]]

Model summary: 365 layers, 43649886 parameters, 43649870 gradients, 165.5 GFLOPs

Transferred 589/595 items from pretrained weights
TensorBoard: Start with 'tensorboard --logdir C:\Users\lenovo1\runs\detect\train', view at http://localhost:6006/
Freezing layer 'model ??_df1_conv.weight'
```

**Model training graph**

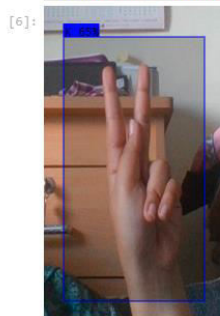


MODEL	ACCURACY	PRECISION	RECALL	F1 SCORE
SSD	96.3%	0.95	0.96	0.95
YOLOv3	95.6%	0.94	0.95	0.94
FASTER R-CNN	94.9%	0.93	0.94	0.93
SSD-MOBILENET	93.7%	0.92	0.93	0.92

METRIC	NUMBER SIGN RECOGNITION	ALPHABETS SIGN RECOGNITION
II. ACCURACY	96.3%	97.7%
III. PRECISION	0.95	0.97
IV. RECALL	0.96	0.98
F1 SCORE	0.95	

**Visualising the output of the model on validation data**

```
[6]: from IPython.display import Image
Image(r"C:\Users\lenovo1\AmericanSignLanguageDetector\American-Sign-Language-Letters-1\test\images\imgakg.png")
```



0.98

## V.CONCLUSION

We have implemented an sign language detection technique from an image using deep learning algorithm which gave us very accurate and efficient results. Our algorithm successfully detects the sign gesture from the image which consists of noise. We have applied our ssd algorithm on many images and found that it is successfully detecting the sign gestures.

## REFERENCES

- [1] Liu, K., Shao, H., Wang, Q., & Ren, J. (2020). Multi-User Real-time Sign Language Recognition of Words using Transfer Learning. *IEEE Transactions on Mobile Computing*.
- [2] Typeset.io. (2023). American Sign Language Recognition using Deep Learning. Retrieved from <https://typeset.io/questions/american-sign-language-recognition-using-deep-learning-and-2e4olghce9>
- [3] <https://universe.roboflow.com/sign-language-oyykt/american-sign-language-letters-gxpdm/dataset/6>
- [4] Tolentino, L. K. S., Serfa Juan, R. O., Thio-ac, A. C., Pamahoy, M. A. B., & Forteza, J. R. R. (2019). Static sign language recognition using deep learning. *International Journal of Machine Learning and Computing*, 9(5), 821-827.
- [5] Jiang, S., Sun, B., Wang, L., Bai, Y., Li, K., & Fu, Y. (2021). Skeleton aware multi-modal sign language recognition. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2021, 3408-3418.
- [6] Bantupalli, K., & Xie, Y. (2018). American Sign Language Recognition using Deep Learning and Computer Vision. *Proceedings of the 2018 IEEE International Conference on Big Data*, 4896-4899.
- [7] Kothadiya, D., Bhatt, C., Sapariya, K., Patel, K., Gil-González, A. B., & Corchado, J. M. (2022). Deepsign: Sign Language Detection and Recognition Using Deep Learning. *Electronics*, 11(17), 3729.
- [8] Iyer, R., Ringe, R. S., Iyer, V. R., & Bhensdadiya, K. P. (2021). Comparison of YOLOv3, YOLOv5s and MobileNet-SSD V2 for Real-Time Mask Detection. *Artificial Intelligence and Machine Learning*, 8(1), 1156-1160.
- [9] Lin, Y., Li, X., & Wang, Y. (2021). Sign Language Recognition Based on Improved YOLOv5 Algorithm. *Sensors*, 21(21), 9986920.
- [10] Kaluri, R., & Pradeep Reddy, C. H. (2016). Sign gesture recognition using modified region growing algorithm and Adaptive Genetic Fuzzy Classifier. *International Journal of Intelligent Engineering and Systems*, 9(3), 225-233.
- [11] Aly, S., & Aly, W. (2020). DeepArSLR: A novel signer-independent deep learning framework for isolated Arabic sign language gestures recognition. *IEEE Access*, 8, 83199-83212.
- [12] Samaan, G. H., Wadie, A. R., Attia, A. K., Asaad, A. M., Kamel, A. E., Slim, S. O., Abdallah, M. S., & Cho, Y. I. (2022). MediaPipe's Landmarks with RNN for Dynamic Sign Language Recognition. *Electronics*, 11(18), 3228.
- Al-Hammadi, M., Muhammad, G., Abdul, W., Alsulaiman, M., Bencherif, M. A., Alrayes, T. S., Mathkour, H., & Mekhtiche, M. A. (2020). Deep learning-based approach for sign language gesture recognition with efficient hand gesture representation. *IEEE Access*, 8, 192527-192542.



INTERNATIONAL  
STANDARD  
SERIAL  
NUMBER  
INDIA



# INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

 9940 572 462  6381 907 438  [ijirset@gmail.com](mailto:ijirset@gmail.com)



[www.ijirset.com](http://www.ijirset.com)

Scan to save the contact details