Enhancing Gesture Recognition for Sign Language Interpretation in Challenging Environment Conditions: A Deep Learning Approach

Domenico Amalfitano[a], Vincenzo D'Angelo, Antonio M. Rinaldi[b], Cristiano Russo[c] and Cristian Tommasino[d]

Department of Electrical Engineering and Information Technology University of Naples Federico II, Via Claudio, 21, Naples 80125, Italy

Keywords: Gesture Recognition, Sign Language, Deep Learning, Real-Time Translation, Accessibility.

Abstract: Gesture recognition systems have gained popularity as an effective means of communication, leveraging the simplicity and effectiveness of gestures. With the absence of a universal sign language due to regional variations and limited dissemination in schools and media, there is a need for real-time translation systems to bridge the communication gap. The proposed system aims to translate American Sign Language (ASL), the predominant sign language used by deaf communities in real-time in North America, West Africa, and Southeast Asia. The system utilizes SSD Mobilenet FPN architecture, known for its real-time performance on low-power devices, and leverages transfer learning techniques for efficient training. Data augmentation and preprocessing procedures are applied to improve the quality of training data. The system's detection capability is enhanced by adapting color space conversions, such as RGB to YCbCr and HSV, to improve the segmentation for varying lighting conditions. Experimental results demonstrate the system's Accessibility and non-invasiveness, achieving high accuracy in recognizing ASL signs.

1 INTRODUCTION

Sign language is a crucial mode of communication for the deaf and hard-of-hearing community. It enables these individuals to express their thoughts and engage in fruitful interactions, giving a complete knowledge representation system. However, there are significant challenges to effective communication between sign language users and those who do not understand sign language (Wadhawan and Kumar, 2021; Rastgoo et al., 2021). Gesture recognition technology, an essential field within computer vision and machine learning, has the potential to bridge this gap. It can achieve this by precisely interpreting sign language gestures in real-time. Gesture recognition in sign language involves developing sophisticated systems that can analyze and comprehend complex hand movements, facial expressions, and body postures that constitute sign language. These systems use cutting-edge algorithms and machine learning techniques to recognize and interpret the rich visual cues in sign language gestures. Gesture recognition technology can facilitate seamless communication between sign language users and the broader community by accurately capturing and understanding these gestures (Mitra and Acharya, 2007; Khan and Ibraheem, 2012).

However, precise recognition of sign language gestures can pose significant challenges, especially under varied lighting and exposure conditions. Variations in lighting, such as harsh sunlight or dim environments, can impact the quality and visibility of the gestures. This makes it difficult for recognition systems to interpret them accurately. Furthermore, differing exposure levels, such as overexposed or underexposed images, can add complexity to the recognition process, leading to possible errors or misinterpretations. It is crucial to address the impact of lighting and exposure conditions on sign language recognition to develop robust and reliable gesture recognition systems. Acknowledging these challenges will allow researchers and practitioners to create algorithms and techniques resilient to varying lighting conditions and exposure levels. This will ensure accurate recognition

[a] https://orcid.org/0000-0002-4761-4443
[b] https://orcid.org/0000-0001-7003-4781
[c] https://orcid.org/0000-0002-8732-1733
[d] https://orcid.org/0000-0001-9763-8745

Amalfitano, D., D'Angelo, V., Rinaldi, A., Russo, C. and Tommasino, C.
Enhancing Gesture Recognition for Sign Language Interpretation in Challenging Environment Conditions: A Deep Learning Approach.
DOI: 10.5220/0012209700003598
In *Proceedings of the 15th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K 2023) - Volume 1: KDIR*, pages 395-402
ISBN: 978-989-758-671-2; ISSN: 2184-3228

regardless of the environmental constraints (Suarez and Murphy, 2012). Furthermore, variations in exposure and lighting conditions can significantly affect the contrast, color, and texture of sign language gestures. Shadows, reflections, and uneven illumination can introduce noise and distortions, making the extraction of meaningful features from the visual data challenging. These variations may also affect the recognition of subtle nuances and fine-grained movements that are essential for the accurate interpretation of sign language (Suarez and Murphy, 2012). In order to tackle the challenges presented by varied lighting and exposure conditions, our solution focuses on examining different color spaces to enhance the accuracy of the gesture recognition system. Traditional color spaces, such as RGB (Red-Green-Blue), may not be robust enough to handle variations in lighting conditions effectively. As a result, we propose investigating alternative color spaces, such as HSV (Hue-Saturation-Value) or YCbCr (Luminance-Blue Chrominance-Red Chrominance), which offer distinct advantages in separating color information from variations in illumination. By utilizing alternative color spaces, we aim to boost the system's ability to differentiate between sign language gestures and their backgrounds under different lighting conditions. These color space transformations can help mitigate the effects of lighting variations, enabling more accurate feature extraction and gesture recognition. Additionally, by exploring multiple color spaces, we can tailor the system to different environments and lighting scenarios, ensuring robust performance across diverse real-world settings. In our experimental evaluation, we will compare the performance of the gesture recognition system using different color spaces under varied exposure and lighting conditions. We will assess metrics such as recognition accuracy, robustness to lighting variations, and computational efficiency. By thoroughly investigating the impact of color space transformations, we aim to provide insights into the most effective color space choices for improving gesture recognition accuracy in different environments. Through our research, we want to contribute to advancing gesture recognition systems for sign language. We specifically aim to tackle the challenges posed by varied exposure and lighting conditions. By investigating different color spaces, we hope to enhance the system's accuracy, enabling effective communication between sign language users and non-sign users across a variety of real-world scenarios. This research has the potential to significantly improve Accessibility and inclusivity for the deaf and hard-of-hearing community, empowering them to engage more seamlessly in a wide range of environments and lighting conditions. The remainder of the paper is organized as follows: Section 2 provides an overview of the related works, Section 3 introduces our proposed approach, Section 4 presents the results obtained from our experiments, and finally, Section 5 provides conclusions and discusses future works.

2 RELATED WORKS

Gesture recognition, particularly in the context of sign language, has been extensively researched in computer vision and machine learning. Researchers have explored various approaches and techniques to interpret sign language gestures accurately. This section provides an overview of the related works in gesture recognition for sign language, highlighting significant contributions and advancements in the field. One approach focuses on utilizing 3D models for gesture recognition, which provides precise results but can be computationally expensive and less efficient for real-time systems. For example, in video surveillance, facial recognition based on 3D face modeling has shown a 40% improvement in performance when reconstructing 3D facial models from non-frontal frames (Park and Jain, 2007). However, the computational complexity of 3D modeling techniques makes them less suitable for real-time applications. Simplified versions of volumetric models rely on the representation of the human skeleton, analyzing the position and orientation of its constituent segments. These skeleton-based systems focus on key parameters, resulting in faster processing times while maintaining satisfactory recognition performance. Such approaches have found practical applications in various human-computer interaction interfaces. Bidimensional models, on the other hand, extract low-level features such as color, shape, and contour directly from images, making them suitable for gesture classification systems. Researchers have extensively employed these models to classify and interpret sign language gestures accurately. Another approach to gesture recognition involves electromyography (EMG)-based models, which classify gestures by analyzing the electrical impulses generated by muscles. This technique allows for a broader range of motion, enabling more natural and expressive gestures. Segmentation is a crucial step in the gesture recognition pipeline, involving dividing images into relevant parts for analysis. Various methodologies have been proposed to address this process. Region-based segmentation methods have been explored, including region growing and region splitting. Region growing involves selecting seed pixels representing dis-

tinct areas and expanding them until the entire image is covered, verifying the homogeneity of each section. Region splitting adopts a top-down approach, recursively dividing the image into sub-images until only homogeneous regions remain. Thresholding, a commonly used technique, categorizes pixels into "object" or "background" based on a predetermined threshold value, resulting in a binary image. Advanced thresholding techniques handle noise and improve segmentation accuracy under challenging lighting conditions. Clustering techniques have also been applied to gesture recognition, which groups elements with similar characteristics within an image. The level of similarity is determined through distance calculations. Shape descriptors play a vital role in recognizing objects, including gesture recognition. These descriptors offer a collection of features that describe a specific shape and can be utilized for efficient image retrieval and comparison, even in the presence of transformations such as scaling, rotation, or translation. Various methodologies for shape description and representation, such as region-based or contour-based approaches, have been proposed. Edge direction histograms are essential tools for detecting objects when color information is unavailable, describing an image's texture. The input image is divided into blocks, and variables representing vertical, horizontal, diagonal, or isotropic edge nature are associated with each block. The Harris corner detector, an operator for corner detection, identifies important points for object description and can reduce the amount of data used in processing. However, its sensitivity to scale changes limits its applicability to images of different sizes. The Scale-Invariant Feature Transform (SIFT) descriptor extracts and describes many features from images, minimizing the influence of local variations on object detection. The angular partitioning-based approach (ARP) is conducted on grayscale images, where circular sections surround the edge to ensure scale invariance, and angles generated are measured for information extraction. Despite progress in gesture recognition technologies, limitations persist, particularly related to the equipment used and image noise. Factors such as camera distance, resolution, and lighting conditions can affect the quality of gesture detection. Additionally, user fatigue, known as "gorilla arm" fatigue, has been observed, particularly in mid-air gestures, where users experience arm fatigue when performing gestures over extended periods.

Gesture recognition has witnessed significant advancements with the application of artificial neural networks, which are computational models inspired by biological systems (Abiodun et al., 2018). Machine-based feature extraction has proven effective in several domains (Russo et al., 2020; Rinaldi and Russo, 2020; Rinaldi et al., 2021). In fact, different types of architectures and purposes exist in the field of neural networks, each catering to specific requirements. Convolutional neural networks (CNNs) have been widely used for image and pattern recognition tasks (Madani et al., 2023; Rinaldi et al., 2020). Their architecture comprises convolutional layers, pooling layers to reduce input parameters, and fully connected layers for classification based on information derived from previous layers. Object detection algorithms can be categorized based on the approach employed (Girshick et al., 2014). Models like R-CNN and Fast R-CNN adopt a two-stage approach: the first stage identifies possible regions of interest, and the second stage employs CNNs to detect objects within those regions. Conversely, models like YOLO and SSD utilize a fully convolutional approach, enabling single-pass detection. The former achieves higher accuracy, while the latter exhibits superior speed, making it more suitable for real-time applications. Given the need for prompt response in gesture recognition systems, single-stage approaches are favored in their implementation. Region-based convolutional networks excel in object detection tasks, distinguishing foreground objects from the background based on a region of interest. These networks aim to produce bounding boxes containing objects and specify their categories. Earlier models utilized selective search algorithms to extract regions of interest (ROIs) and subsequent convolutional operations to classify objects within the identified regions, followed by support vector machines (SVMs) for object region classification and linear regressors for bounding box refinement. However, these architectures suffered from time-consuming training due to the large number of regions identified. Subsequent advancements in object detection have led to the evolution of these models, resulting in more efficient techniques. The Fast R-CNN architecture (Girshick, 2015) directly generates feature maps from the input image, eliminating the need for region proposal stages and improving speed. Faster R-CNN (Ren et al., 2015)introduces a Region Proposal Network (RPN) that efficiently and accurately identifies regions of interest, sharing convolutional features with downstream detection networks. Region-based Fully Convolutional Networks (R-FCN) (Dai et al., 2016) further enhances detection speed by sharing computations for all region proposals. Mask R-CNN (He et al., 2017) efficiently detects objects while simultaneously generating segmentation masks for each instance. This approach replaces RoIPooling with RoIAlign for more

accurate pixel-level segmentation. Single Shot MultiBox Detector (SSD) combines object classification and bounding box prediction in a single pass, utilizing predefined bounding boxes of varying sizes. By evaluating object categories and adapting the boxes to their shape, SSD handles objects of different scales effectively. MobileNet SSD v2 (Sandler et al., 2018), designed for real-time applications on mobile devices, achieves high-speed processing. YOLO (You Only Look Once) (Redmon et al., 2016) proposes a novel object detection approach by recognizing image regions with high probabilities of containing objects, enabling single-pass evaluation. YOLO is a global reasoning network that reasons about the entire image and all objects within it, dividing the input image into an S×S grid. Each grid cell is responsible for detecting an object if its center falls within that cell. The approach is extremely fast, capable of processing 45 frames per second, with a faster version reaching 155 frames per second. However, YOLO may exhibit more localization errors compared to other detection systems. YOLOv7 (Wang et al., 2022), introduced in 2022, outperforms previous detection models in terms both speed and accuracy. It requires significantly less expensive hardware compared to other neural networks and is trained solely on the MS COCO dataset without the use of pre-trained models. The cited study (Bharati and Pramanik, 2020), conducted in 2020, provides a detailed performance comparison of various object detection models. It is important to note that performance depends on factors such as input image resolution, dataset, and training configurations. Model accuracy is measured using mean average precision (mAP). From the reported data, it can be inferred that SSD and R-FCN are among the fastest models but do not match the precision and accuracy of Faster R-CNN. While SSD is less affected by the choice of feature extractors, it is less accurate in detecting small objects. YOLO remains the fastest architecture, with a speed of approximately 21-155 frames per second, while Mask R-CNN exhibits the highest accuracy, with an average precision of 47.3.

3 SIGN LANGUAGE TRANSLATION SYSTEM

Our approach targets developing a real-time American Sign Language (ASL) translation system. ASL stands as the principal sign language for deaf communities in America, Canada, and various countries in West Africa and Southeast Asia. The system captures visual input from a camera, executes gesture detection and recognition, and then exhibits the corresponding textual translation along with a reliability score.

Figure 1 showcases the architecture of our system, which is composed of the following key modules:

- **Capture Module:** This module represents the eyes of the system. It uses the webcam to capture a video frame, which becomes the initial raw data for the entire translation process. The ability to effectively capture frames in different lighting conditions and at varying distances underpins its functionality.
- **Preprocessing Module:** This module transforms the raw video frame into a format that the system can more efficiently analyze. This includes image transformations like gamma correction, which can adjust the brightness of the image and improve visibility. Noise reduction and normalization might also occur at this stage to improve the accuracy of subsequent detection and classification tasks. Additionally, the prepossessing pipeline also applies image resizing to dimensions of 320x320.
- **Detection Module:** This is where the actual sign detection happens. This module analyzes the preprocessed frame and identifies the presence of any signs. The output of this module is a set of regions in the frame where a sign is likely present, often represented by bounding boxes.
- **Classification Module:** The identified signs are then fed into this module. Here, a class is assigned to each detected sign based on the trained model. The Classification Module's role is to translate the identified signs into their equivalent meanings in spoken or written language.
- **Visualization Module:** The final module takes the classified signs and presents them to the user in an easy-to-understand format. This involves displaying the corresponding textual translation and the screen's reliability score. This module provides critical feedback to users, allowing them to understand how the system interprets their signs.

Each module plays a critical role in the system, contributing to the overall goal of effective and efficient real-time ASL translation.

3.1 Sign Detection Implementation

We implemented our system using Python 3.8, drawing from the power of the TensorFlow 2 Object Detection API and the OpenCV library. The training process embraces a supervised learning approach, leveraging labeled data that we created with the open-source software LabelImg. Once we load the trained

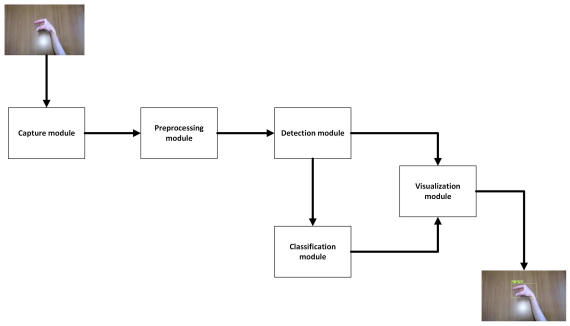

Figure 1: System Architecture.

model, we start the detection process. The system captures the video stream from the webcam with the help of the OpenCV library and starts displaying real-time detections on the screen. Each detection comes defined with bounding boxes, class labels, and confidence scores. We have set up the system to display a maximum of two detections simultaneously. We have established a confidence threshold that dictates the minimum score for reliable predictions. This threshold essentially functions as a filter, discarding results that lack sufficient confidence. Experiments in different visual conditions have set the value of the threshold. To ensure robust performance, we put the system through rigorous testing using a variety of webcams, each with different resolutions. We experimented with several color spaces and altered the positioning distance between the webcam and the target objects. We ran these tests under an array of lighting conditions and with both simple and complex backgrounds to mimic real-world scenarios. In the forthcoming section, we will delve into the results of these tests. Capitalizing on the trained model and real-time video input, our system strives to accurately detect and identify the specified signs across a multitude of situations. The bounding boxes and class labels serve as intuitive visual feedback, giving users insights into how the system interprets their signs. Confidence scores, on the other hand, offer a measure of the system's prediction certainty. These scores empower users to evaluate the reliability of the sign interpretations.

4 EXPERIMENTAL RESULTS

In this section, we present our experimental strategy, report the results, and go into deep related discussions.

To evaluate our approach, we constructed our dataset, recognizing that the phase of dataset creation plays a crucial role in achieving pleasing results. In this study, we selected a restricted set of signs. Such a strategy initially allows the model to learn essential sign recognition before gradually expanding the training set. This approach can be useful when dealing with complex sign language systems, providing a stepping stone to incorporate more signs into the model progressively.

After all, the quality and volume of input data directly steer the detector's accuracy. We harnessed four different webcam models to capture images for each sign. To boost system robustness, we portrayed each sign with slight variations in hand poses and interchangeably used hands in each image.

Following the data collection, we annotated each image using the LabelImg software (Tzutalin, 2015). During this annotation process, we matched each sign with its appropriate label and highlighted the region of interest with a bounding box. Consequently, we generated an XML file specifying the image information, the assigned label, and the bounding box coordinates.

Constructing the dataset required us to engage in careful labeling to depict the nuances in sign expressions accurately. This variation in the dataset helps ensure that our model learns to recognize and generalize different instances of the same sign effectively. Additionally, we optimized the size of the dataset, striking a balance between the requirement for ample samples and the practical limitations of data collection.

We use the annotated dataset to train the sign language translation model. By exposing the model to a wide array of sign variations, we aim to enhance its generalization capabilities and its accuracy in recognizing signs in real-world scenarios. By constructing an effective dataset, we set the stage for training a resilient and dependable sign language translation system.

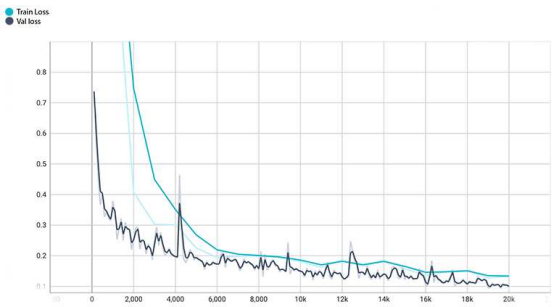

Figure 2: Loss trend over epochs.

4.1 Training

To augment the efficiency of the training procedure, we leveraged the well-established technique of transfer learning (Cook et al., 2013). This approach enables adapting a pre-trained artificial intelligence model to a task distinct from its initial training objective. Specifically, we extracted the lower layers from the pre-trained network for reuse within our newly designed network to recognize a set of five distinct signs.

We elected to use the *SSD MobileNet V2 FPN Lite* model, with an input size of 320x320 pixels pre-trained on the COCO 2017 dataset (Lin et al., 2014). The rationale for this selection is the neural network's impressive processing speed of 22ms and a COCO mAP of 22.2. These performance metrics, coupled with the simplicity of the network's architecture that facilitates its operation even on low-power devices, deemed it an apt choice for our application.

In the training phase, we employed augmentation techniques, specifically random cropping and horizontal flipping, to enhance the network's generalization capabilities. The model's training leverages a momentum optimizer (Ruder, 2016), featuring a cosine decay learning rate strategy (Loshchilov and Hutter, 2016). This optimizer initiates with a base learning rate of 0.08, diminishing gradually per a cosine decay schedule, thereby promoting smoother convergence and limiting the risk of overshooting. A warm-up phase within the initial 1000 steps gradually escalates the learning rate from 0.026, a measure that mitigates the risk of premature divergence. A momentum optimizer value of 0.9 encourages the optimizer's acceleration in the correct direction and dampens oscillations, proving beneficial in complex optimization landscapes. We employ a loss function that is a linear combination of localization and classification losses, represented in Equation 1.

$$\mathcal{L} = w_1 \cdot \mathcal{L}_{loc} + w_2 \cdot \mathcal{L}_{class} \quad (1)$$

In our configuration, we deemed it appropriate to assign equivalent importance to both components localization loss ($\mathcal{L}_{loc}$) and classification loss ($\mathcal{L}_{class}$), thus setting $w_1 = w_2 = 1.0$. We specifically utilize Smooth-L1 as the localization loss, whereas Focal Loss, with a gamma of 2 and alpha of 0.25, serves as our classification loss. We conducted the training phase over 20,000 epochs.

As illustrated in Figure 2, both the training and validation data exhibit a decrease in loss functions, signaling the system's excellent adaptation to the new data.

4.2 Evaluation

One of the goals of this study is to obtain a system for sign recognition that gives robust predictions under different environmental conditions, such as brightness variations and background complexity. In low-light conditions, HSV (Hue, Saturation, Value) color space is often considered more advantageous compared to YCbCr (Luma, Chroma Blue, Chroma Red) color space due to the distinct characteristics of their respective components. The Value component in HSV directly represents the brightness or intensity of a color, making it particularly suitable for object detection in low-light environments. Despite reduced overall illumination, the Value component still exhibits distinguishable variations in brightness, enabling effective differentiation of objects. Conversely, YCbCr separates the color information from the brightness information, with the Luma component representing the brightness. However, the Luma component may not provide sufficient contrast for robust object detection in low-light conditions. This discrepancy arises from the fact that the Luma component is less sensitive to variations in brightness under low-light conditions, potentially leading to decreased detection accuracy. Consequently, HSV, emphasizing the Value component, is generally favored over YCbCr for improved object detection performance in low-light conditions. We observed that predicted classes were correct during tests conducted in non-optimal environmental conditions for RGB input frames. However, their probability scores were not predominant concerning other classes. For these reasons, we compared RGB against HSV under bright environment conditions and RGB against YCbCr under low-light environment conditions.

The results of tests conducted for each sign are presented in Table 1 and Table 2 and summarized graphically in Figure 3 and Figure 4.

For example, in tests carried out in brightly lit environments, an improvement in confidence scores was observed when converting to the HSV color space, as demonstrated in the examples shown in Figure 5 where the confidence score for the "Hello" sign increases from 56.08% to 99.24%. Similarly, in tests conducted in low-light conditions, detection quality was improved by using the YCbCr color space, as shown in Figure 6 where it can be observed that the confidence score increases from 51.45% to 93.29% for the "No" sign.

Table 1: Comparison of Probability Scores for RGB and HSV in low-light environments.

Sign	RGB (%)	HSV (%)
"Yes"	46.46	99.39
"No"	50.01	96.02
"Hello"	54.05	99.42
"I Love You"	49.89	96.72
"Thank You"	68.39	97.14

Table 2: Comparison of Probability Scores for RGB and YCbCr in bright environments.

Sign	RGB (%)	YCbCr (%)
"Yes"	52.09	92.95
"No"	58.30	95.9
"Hello"	50.43	99.81
"I Love You"	56.70	99.01
"Thank You"	59.92	99.03

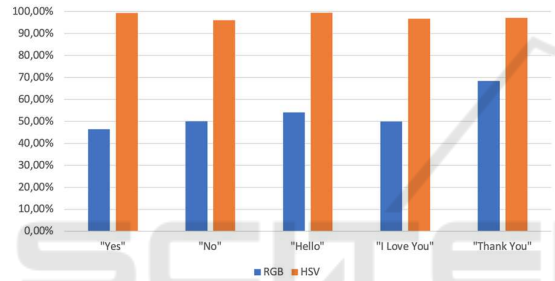

Figure 3: Accuracy comparison between each sign's RGB and HSV color spaces.

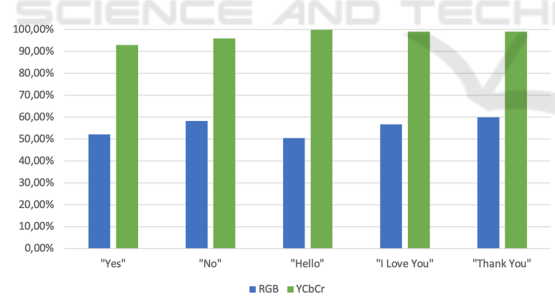

Figure 4: Accuracy comparison between each sign's RGB and HSV color spaces.

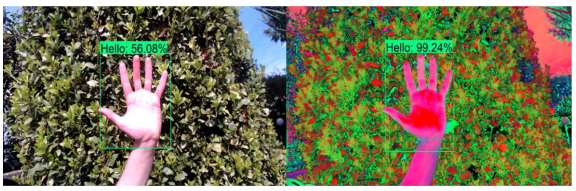

Figure 5: Sign recognition comparison between RGB and HSV color spaces in a bright environment. HSV outperforms RGB in this condition.

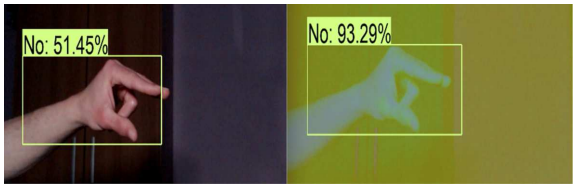

Figure 6: Sign recognition comparison between RGB and YCbCr color spaces in low-light environment. YCbCr outperforms RGB in this condition.

5 CONCLUSIONS

Gestures provide a simple and effective method of communication, which is why gesture recognition systems are gaining popularity. Depending on the application domain, it is important to consider the choice of technologies and training architectures carefully. While more sophisticated sensors can provide more accurate detections, they are often less affordable and accessible.

Our proposed system is capable of real-time translation of five symbols from American Sign Language (ASL), which is the predominant language among deaf communities in America, Canada, West Africa, and Southeast Asia. By analyzing the data stream from the webcam, the system displays the detection on the screen, providing bounding boxes, classes, and confidence scores. It is based on the SSD Mobilenet architecture, designed to deliver real-time performance on low-power devices while achieving high levels of accuracy under optimal conditions. In challenging lighting conditions, the detection quality has been significantly improved through color space conversions, enabling better image segmentation. The implemented technology demonstrates Accessibility and non-invasiveness.

Overall, this research contributes to advancing gesture recognition systems, particularly for sign languages, by leveraging Deep Learning techniques. The proposed system shows promise in real-time translation and has the potential to facilitate communication between deaf individuals and the wider community. Future work may focus on expanding the vocabulary using other knowledge sources (Caldarola et al., 2015; Muscetti et al., 2022) and improving the system's robustness under various environmental conditions, ultimately aiming to make gesture recognition more accurate, efficient, and inclusive.

ACKNOWLEDGMENTS

We acknowledge financial support from the project PNRR MUR project PE0000013-FAIR.

REFERENCES

Abiodun, O. I., Jantan, A., Omolara, A., Dada, K. V., Mohamed, N. A., and Arshad, H. (2018). State-of-the-art in artificial neural network applications: A survey. *Heliyon*, 4(11):e00938.

Bharati, P. and Pramanik, A. (2020). Deep learning techniques—r-cnn to mask r-cnn: a survey. In *Computational Intelligence in Pattern Recognition: Proceedings of CIPR 2019*, pages 657–668.

Caldarola, E. G., Picariello, A., and Rinaldi, A. M. (2015). Big graph-based data visualization experiences: The wordnet case study. In *IC3K 2015 - Proceedings of the 7th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management*, page 104 – 115.

Cook, D., Feuz, K. D., and Krishnan, N. C. (2013). Transfer learning for activity recognition: A survey. *Knowledge and information systems*, 36:537–556.

Dai, J., Li, Y., He, K., and Sun, J. (2016). R-fcn: Object detection via region-based fully convolutional networks. In *Advances in neural information processing systems*, volume 29.

Girshick, R. (2015). Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448.

Girshick, R., Donahue, J., Darrell, T., and Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587.

He, K., Gkioxari, G., Dollár, P., and Girshick, R. (2017). Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969.

Khan, R. Z. and Ibraheem, N. A. (2012). Hand gesture recognition: a literature review. *International journal of artificial Intelligence & Applications*, 3(4):161.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., ..., and Zitnick, C. L. (2014). Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer International Publishing.

Loshchilov, I. and Hutter, F. (2016). Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*.

Madani, K., Rinaldi, A. M., Russo, C., and Tommasino, C. (2023). A combined approach for improving humanoid robots autonomous cognitive capabilities. *Knowledge and Information Systems*, 65(8):3197–3221.

Mitra, S. and Acharya, T. (2007). Gesture recognition: A survey. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 37(3):311–324.

Muscetti, M., Rinaldi, A. M., Russo, C., and Tommasino, C. (2022). Multimedia ontology population through semantic analysis and hierarchical deep features extraction techniques. *Knowledge and Information Systems*, 64(5):1283–1303.

Park, U. and Jain, A. K. (2007). 3d model-based face recognition in video. In *Advances in Biometrics: International Conference, ICB 2007, Seoul, Korea, August 27-29, 2007. Proceedings*, pages 1085–1094. Springer Berlin Heidelberg.

Rastgoo, R., Kiani, K., and Escalera, S. (2021). Sign language recognition: A deep survey. *Expert Systems with Applications*, 164:113794.

Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788.

Ren, S., He, K., Girshick, R., and Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, volume 28.

Rinaldi, A. M. and Russo, C. (2020). A content based image retrieval approach based on multiple multimedia features descriptors in e-health environment. In *2020 IEEE International Symposium on Medical Measurements and Applications (MeMeA)*, pages 1–6. IEEE.

Rinaldi, A. M., Russo, C., and Tommasino, C. (2020). A knowledge-driven multimedia retrieval system based on semantics and deep features. *Future Internet*, 12(11):183.

Rinaldi, A. M., Russo, C., and Tommasino, C. (2021). Visual query posing in multimedia web document retrieval. In *2021 IEEE 15th International Conference on Semantic Computing (ICSC)*, pages 415–420. IEEE.

Ruder, S. (2016). An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*.

Russo, C., Madani, K., and Rinaldi, A. M. (2020). An unsupervised approach for knowledge construction applied to personal robots. *IEEE Transactions on Cognitive and Developmental Systems*, 13(1):6–15.

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., and Chen, L.-C. (2018). Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4510–4520.

Suarez, J. and Murphy, R. R. (2012). Hand gesture recognition with depth images: A review. In *2012 IEEE RO-MAN: the 21st IEEE international symposium on robot and human interactive communication*, pages 411–417. IEEE.

Tzutalin (2015). Labelimg. https://github.com/tzutalin/labelImg.

Wadhawan, A. and Kumar, P. (2021). Sign language recognition systems: A decade systematic literature review. *Archives of Computational Methods in Engineering*, 28:785–813.

Wang, C.-Y., Bochkovskiy, A., and Liao, H.-Y. M. (2022). Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. *arXiv preprint harXiv:2207.02696*.