Evolving Systems (2024) 15:185–201
https://doi.org/10.1007/s12530-023-09512-1

ORIGINAL PAPER

Attention-based hand semantic segmentation and gesture recognition using deep networks

Debajit Sarma[1] · H Pallab Jyoti Dutta[1] · Kuldeep Singh Yadav[2] · M.K. Bhuyan[1] · Rabul Hussain Laskar[2]

Received: 22 July 2022 / Accepted: 6 June 2023 / Published online: 3 July 2023

Abstract
The ability to discern the shape of hands can be a vital issue in improving the performance of hand gesture recognition for human–computer interaction. Segmentation itself is a very challenging problem having various constraints like illumination variations, complex background etc. The objective of the paper is to incorporate the perception of semantic segmentation into a classification problem and make use of the deep neural models to achieve improved results for both static and dynamic gestures. This paper utilizes the UNet architecture with attention-module to obtain the semantically segmented masks of the input images, which are then fed to a classifier for recognition. The concept of attention-mechanism adds to the improvement of segmentation accuracy. In this work, for static gestures, the top classifier layer of the VGG16 model is replaced with a classifier designed specifically for classifying the gestures at hand. For dynamic gestures, 3D-CNN (C3D) architecture is used as a classifier that can capture spatial as well as temporal information of a gesture video. The data augmentation process is used in preprocessing to generate a sufficient number of training images for the aforementioned CNN-based models. Significant and improved recognition has been achieved for both static and dynamic hand gesture databases through the inherent feature learning capability of CNN and refined segmentation.

Keywords Semantic segmentation · UNet · CBAM · VGG16 · C3D · Static and dynamic hand gestures · Human–computer interaction

1 Introduction

Accurate segmentation of the hand or the gesturing body part from the captured videos or images still remains a challenge in computer vision for many constraints like illumination variations, background complexity, occlusion and so on Chakraborty et al. (2017), Sarma and Bhuyan (2021). Illumination variations affect the accuracy of skin color segmentation methods. Poor illumination may change the chrominance properties of the skin colors, and the skin color will appear different from the original color. A major challenge in gesture recognition is the proper segmentation of skin-colored objects (e.g., hands, face) against a complex static/dynamic background. The accuracy of skin segmentation algorithms is limited because of objects in the background that are similar in color to human skin. Skin-colored objects present in the background also increase false positives (Sarma and Bhuyan 2018, 2022). All these factors make the detection of hand to be one of the vital stages in the gesture recognition system. The above-mentioned constraints need to be taken into care for detecting the hand across each frame in both static and dynamic hand gesture recognition, where segmentation generally becomes an unavoidable process.

Due to these constraints, researchers generally opt for other types of segmentation techniques like object detection by bounding box, semantic segmentation, or instance segmentation as shown in Fig. 1. In object detection through

✉ Debajit Sarma
s.debajit@iitg.ac.in

H Pallab Jyoti Dutta
h18@iitg.ac.in

Kuldeep Singh Yadav
kuldeeptheyadav@gmail.com

M.K. Bhuyan
mkb@iitg.ac.in

Rabul Hussain Laskar
rhlaskar@ece.nits.ac.in

[1] Department of EEE, IIT Guwahati, Guwahati, Assam 781039, India

[2] Department of ECE, NIT Silchar, Silchar, Assam 788010, India

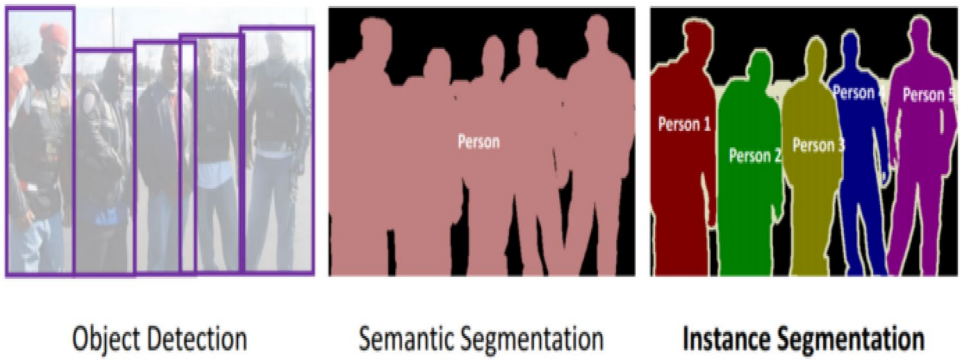

Fig. 1 Different techniques for the segmentation process

bounding boxes, people try to locate and classify multiple objects within an image/video, by drawing bounding boxes around them and then classifying what's in the box. One major disadvantage here is that we only get a bounding box covering the object, but we really don't get an idea regarding the shape of the object. Semantic segmentation is more informative where it classifies each and every pixel in the image and assigns an appropriate class level to the pixels and links the similar pixels into a group (class). Instance segmentation is a challenging task that requires the prediction of object instances and their per-pixel segmentation task. This makes it a hybrid of semantic segmentation and object detection. There is no hard and fast rule regarding their adoption and it all depends on the application where one preferred scheme can be applied using various approaches.

With the advancement in neural networks and computing devices, tasks like image classification, object recognition, and segmentation have been carried out with improved results and much efficiency. Convolutional Neural Networks (CNN) form the backbone of most of the modern-day deep learning models, which have achieved ground-breaking outcomes in regards to the above tasks. It has helped to achieve classification results close to the human level. Also, it is capable of localizing objects by assigning appropriate class labels to the pixels, which is the governing principle of semantic segmentation (Dutta et al. 2020).

Attention mechanism, which plays an important role in human perception, can effectively highlight useful information while suppressing the redundant one. Recently, attention mechanism has been receiving wide attention in a variety of tasks, such as natural language processing for machine translation, natural image classification, salient object detection, natural image segmentation, medical image segmentation and classification in medical image analysis fields, image captioning etc. There are many attempts that have embedded attention modules into deep neural network architecture for improving the performance of image segmentation, classification, and object detection in computer vision fields.

Meanwhile, UNet (Ronneberger et al. 2015) has achieved great success in the field of medical image segmentation, and it is also the mainstream of current segmentation methods. The network has a progressively narrowing structure, which tends to encode the input into a fine to the coarse manner, followed by a decoding structure that broadens progressively. However, during the process of downsampling, UNet constantly reduces the dimension of the image, which results in poor segmentation accuracy for the small-scale objects. Considering that attention mechanisms can enhance local feature expression, to solve the insufficient segmentation accuracy, researchers generally adopt attention mechanisms in the various segmentation processes. As a computing resource allocation scheme, the attention mechanism uses limited resource allocation to process more important information to solve the problem of information overload. Generally, the input of a neural network often contains a lot of redundant information and all the information is not needed to be focused on. So, one can pay attention only to something important to improve time and space utilization. Researchers have demonstrated that introducing an attention mechanism into UNet can enhance local feature expression and improve the performance of image segmentation.

It is already mentioned that hand segmentation is a challenging task due to constraints like illumination variations, background complexity, occlusion and so on. Background noise and varying lighting conditions also cause occlusions and clutter, which have to be considered. However, the most accurate approaches that try to mitigate such constraints tend to employ multiple modalities derived from input frames, such as optical flow or depth information (Kavyasree et al. 2020; Sarma et al. 2022). This practice limits real-time performance due to intense extra computational cost. In this paper, we avoid depth information or optical flow computation by proposing a hand gesture recognition method based on RGB frames combined with hand segmentation masks. Benitez-Garcia et al. (2021) found that the semantic segmentation is more than two times faster than the optical flow, making the semantic segmentation an alternative feasible option for real-time applications as well. Contextual information extraction is difficult with hand-crafted features in conventional

machine learning techniques. Attention-based methods have been proved to be effective ways to obtain important contextual information in different segmentation methods like semantic segmentation. This work is an extension of our previous work (Dutta et al. 2020) where segmentation with UNet was done without any attention mechanism and it was used for only static gestures. But it has disadvantages like inaccurate segmentation, specially in clutter backgrounds. Moreover, it was applicable to only static gestures. Therefore, in this work, we have proposed a deep learning attention-based segmentation framework as a solution to the above-mentioned issues. The rule-based algorithms of attention-based semantic segmentation for static gesture interpretation have successfully been transferred and extended into dynamic gesture recognition where C3D is used to automatically extract the robust temporal and spatial features to recognize the hand gestures. Here, we aim to explore the effectiveness of a recent attention module called Convolutional Block Attention Module (CBAM) (Woo et al. 2018) combined with UNet architecture for hand segmentation purposes.

For recognition of hand gestures, a lot of research has been carried out based on traditional machine learning approaches as well as deep neural networks. Here, an effective and more transparent solution for static and dynamic hand gesture recognition problems is delved into by bringing the perception of attention into semantic segmentation in a classification problem as shown in Fig. 2. The main contributions of our proposed model are as follows:

1. A deep supervised attention module to focus and guide the learning of information for segmentation in UNet structure.
2. We have proposed a novel approach for both static and dynamic hand gesture recognition where the attention technique is used to increase the segmentation performance on similar gestures.
3. We have demonstrated how the quality of segmented images impacts the performance of hand gesture classification through experiments on two databases and have proven our network has better results than state-of-the-art on a noisy dataset.

2 Related works in literature

2.1 Semantic segmentation

There are several model variants based on Convolutional Neural Networks (FCNs) to enhance contextual aggregation in segmentation. Faster R-CNN (Ren et al. 2015), R-FCN (Dai et al. 2016) are used to exploit the region of each instance, and then predict the mask for each region. He et al. (2017) proposed Mask R-CNN that is built on the top of Faster R-CNN by adding an instance-level semantic segmentation branch. On the other hand, semantic segmentation, using CNN-based methods, was pioneered by Long et al. (2015) using Fully Convolutional Network (FCN). This work primarily defined a skip network that combined the information from the coarse upper layer of the deep neural architecture with the lower fine layer, which in turn helped

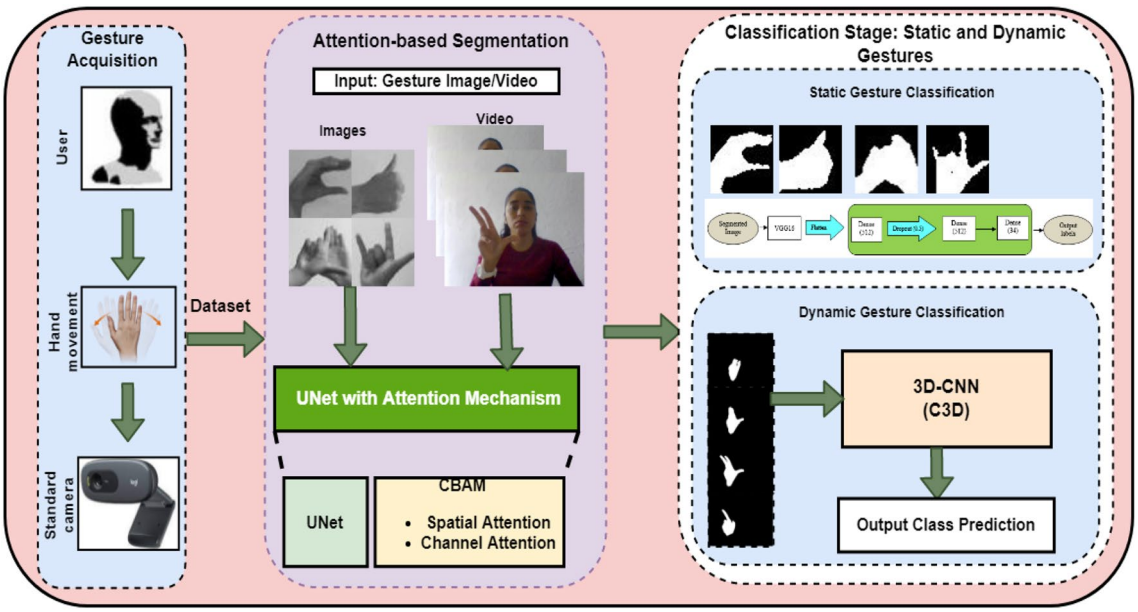

Fig. 2 Block diagram of our proposed hand gesture recognition framework

achieve meticulous segmentation results. In Ronneberger et al. (2015), Ronneberger et al. defined an architecture that apprehended contextual information through a gradually contracting path and localized the concerned objects via a symmetric expanding path. Due to its structure, it was called UNet and it is a classic work for medical image segmentation. The authors trained the network on a few biomedical images with the application of the data augmentation process and achieved state-of-the-art results. UNet++ (Zhou et al. 2018) re-design skip pathways that connect the encoder and decoder networks and adopt deep supervision on the basis of UNet to further improve the segmentation accuracy of the model. Huang et al. (2020) proposed a novel model UNet3+ that reconstructs the connections between the encoder and the decoder and internally. The role of the connections between decoders is to capture fine-grained details and coarse-grained semantics from the entire scale.

Apart from UNet, the Deeplab series is also one of the most popular CNN architectures in the field of semantic segmentation. Since 2014, v1 (Chen et al. 2014), v2 (Chen et al. 2017b), v3 (Chen et al. 2017a) and v3+ (Chen et al. 2018) series have been successively proposed. Deeplabv2 (Chen et al. 2017b) and Deeplabv3 (Chen et al. 2017a) adopt atrous spatial pyramid pooling (ASPP) to embed contextual information, which consists of parallel dilated convolutions with different dilated rates. Deeplabv3+ (Chen et al. 2018) is currently the latest neural network structure of the Deeplab series, which is mainly improved based on Deeplabv3. This network mainly borrows the traditional encoder-decoder architecture, expands a simple and effective module for recovering boundary information.

Semantic segmentation requires a significant number of annotated data at the pixel level, and this drawback is addressed in Souly et al. (2017). Souly et al. (2017) proposed a semi-supervised method of semantic segmentation, based on Generative Adversarial Network (GAN) (Goodfellow et al. 2014). Zhang et al. (2018) proposed a novel model named SegGAN, formed by fitting a pre-trained deep semantic segmentation model into a GAN. This composite network learned features, which reduced the loss between the original images and the generated ones, and eventually arrived at better segmentation masks. In order to make reasonable use of limited visual information processing resources, attention can be used to explain the alignment relationship between input and output data and explain what the model has learned. The reason why the attention mechanism is so

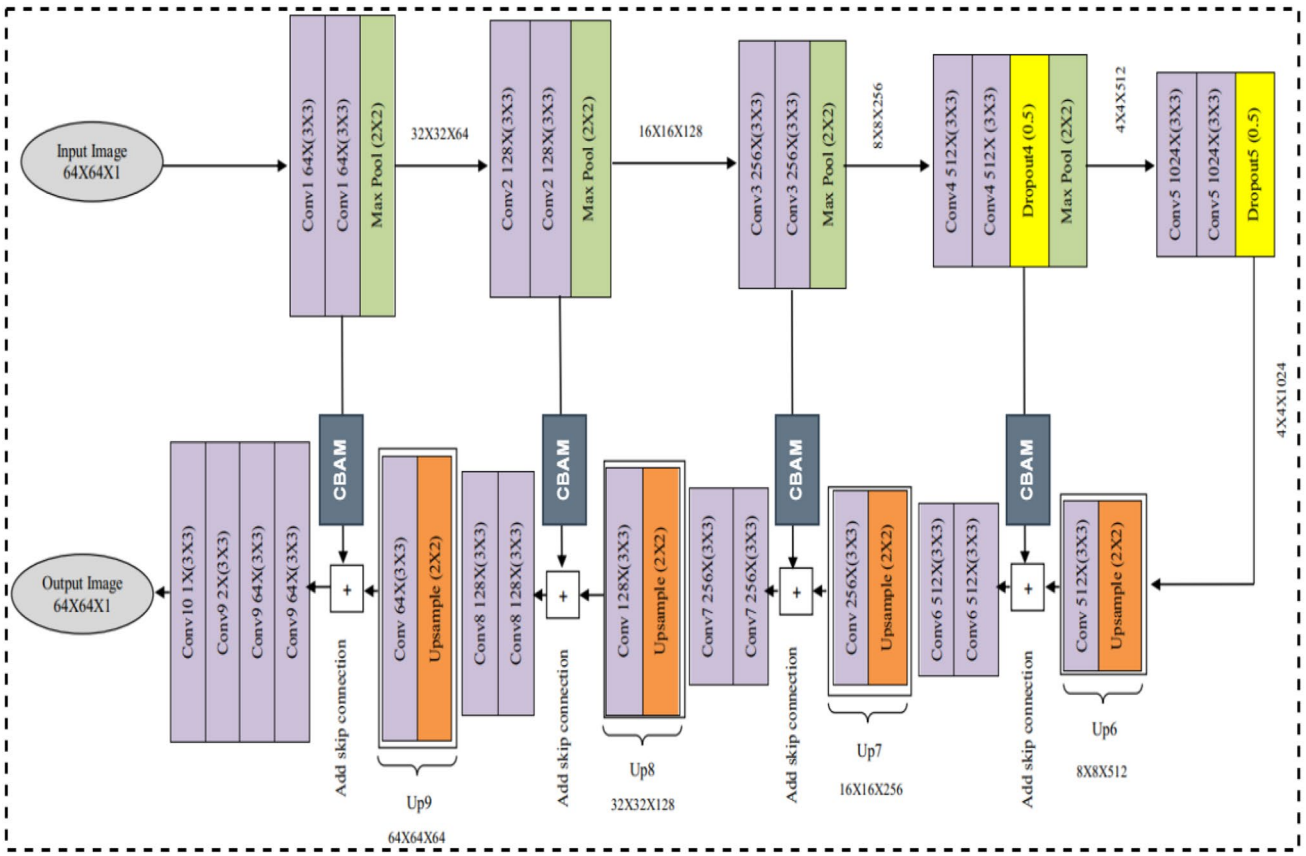

Fig. 3 UNet architecture used for semantic segmentation with attention mechanism

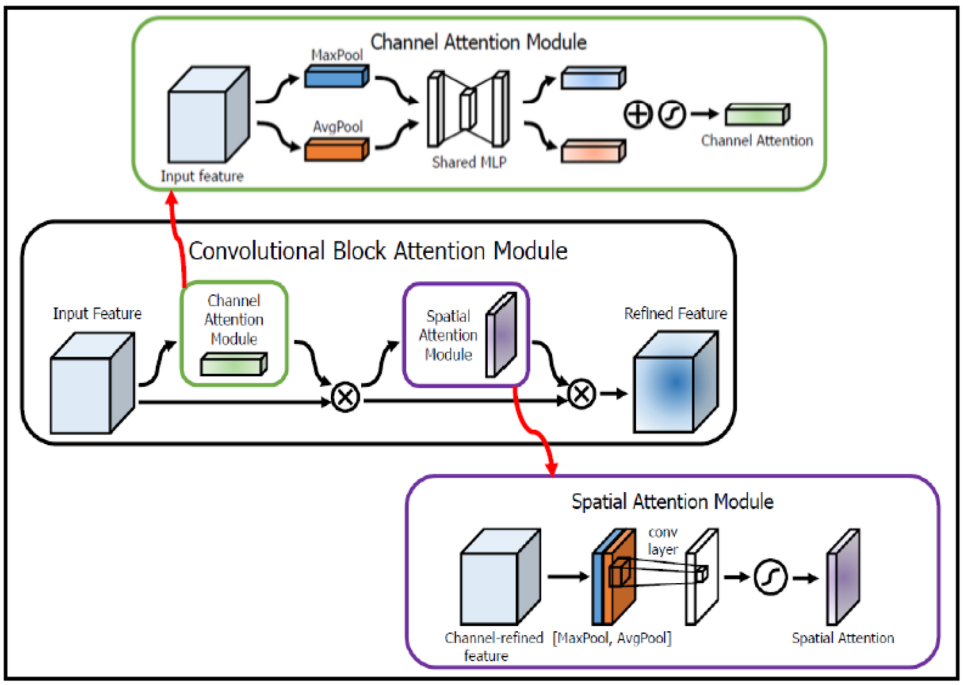

Fig. 4 CBAM architecture used for attention mechanism

popular is that the attention mechanism gives the network the ability to distinguish and focus.

2.2 Attention mechanism

Attention mechanisms are widely used in computer vision to extract better visual features. Attention not only tells where to focus, but it also improves the representation of interests. Our goal is to increase representation power by using an attention mechanism: focusing on important features and suppressing unnecessary ones.

Considering the **number of positions**, attention mechanisms are usually divided into soft attention and hard attention. The soft attention mechanism is easy to implement and it attends to arbitrary input locations using spatial transformer networks (Jaderberg et al. 2015). It produces a distribution over input locations, reweight features and feed as input. Soft attention focuses on the image channels and is a deterministic attention mechanism. Its advantage is that the derivative of the function can be differentiated. Thus, the gradient values can be back-propagated through the neural network. In contrast, hard attention emphasizes the salient areas of the image and is a random prediction process that focuses primarily on dynamic changes. It can't use gradient descent and need reinforcement learning.

According to the **type of architecture**, attention models can be implemented as encoder-decoder (Lea et al. 2017; Hu et al. 2018), transformer (Jaderberg et al. 2015) and memory networks (Li et al. 2020). An encoder-decoder-based attention model takes any input representation and reduces it to a single fixed length, a transformer network aims to capture global dependencies between input and output, and in the memory networks, facts that are more relevant to the query are filtered out.

Depending on the **type of focus**, there are two types of attention mechanism: spatial attention (Wang et al. 2017) and channel attention (Hu et al. 2018). The spatial attention mechanism makes the network pay more attention to the spatial position of the target, and the channel attention mechanism tends to focus on the size of the target (Fu et al. 2019).

With respect to **number of sequences**, attention can be of three types, namely distinctive, co-attention and self-attention. While in distinctive attention candidate and query states belong to two distinct input and output sequences, in self-attention (Vaswani et al. 2021) the candidate and query states belong to the same sequence. The self-attention mechanism just concerns single rather than multiple cross-modal semantic information, that is, query, key, and value are all obtained from the same semantic information in contrast to spatial transformer networks. Co-attention accepts multiple input sequences as input at the same time and jointly produces an output sequence.

There have been several attempts like (Lea et al. 2017; Hu et al. 2018; Wang et al. 2017) to incorporate attention processing to improve the performance of CNNs in large-scale classification tasks. Wang et al. (2017) proposed Residual Attention Network which uses an encoder-decoder style attention module. With the refining of the feature maps, the network not only performs well but is also robust to noisy inputs. Instead of directly computing the 3D attention

map, channel attention and spatial attention can be learned separately. Hu et al. (2018) introduced a compact module to exploit the inter-channel relationship in their Squeeze-and-Excitation (SE) module. They used global average-pooled features to compute channel-wise attention. However, these are suboptimal features in order to infer fine channel attention. They also missed the spatial attention, which plays an important role in deciding 'where' to focus as shown in Chen et al. (2017c). Li et al. (2018) and Yu et al. (2018) feed the features of deep layers with stronger semantics into SE-like attention block to provide high-level category information, which helped to precisely recover details in the upsampling stage of image segmentation.

2.3 Attention-based methods for hand gesture recognition

The first noted visual attention-based work to recognize hand postures in the complex background was given by Pisharady et al. (2013). The proposed method was simple without any deep architecture and utilized a Bayesian model of visual attention generating a saliency map to detect and identify the hand region. Feature-based visual attention was implemented using a combination of high-level (shape, texture) and low-level (color) image features. Using deep networks, a multi-channel method was proposed by Narayana et al. (2018) with spatial attention focused on the hands, and different channels were fused using a sparse network. Narasimhaswamy et al. (2019) extended MaskRCNN with a novel attention mechanism to incorporate contextual cues that captures non-local dependencies between features. Dhingra and Kunz (2019) proposed a stacked 3D attention-based residual network (Res3ATN) with convolution, residual and attention blocks in a sandwich manner layer after layer. The multiple attention blocks can generate different features at each attention block. D'Eusanio et al. (2020) used a transformer-based neural network for dynamic hand gesture recognition. Abdul et al. (2021) proposed an attention-model based on Inception CNN for extracting spatial features and Bi-LSTM (long short-term memory) for temporal feature extraction in Arabic sign language classification. Li et al. (2021) applied transformer-based self-attention mechanism to collect features from cropped input frames and combined through mutual-attention feature fusion to produce a joint RGB-D representation.

Most of these attention mechanisms, however, do not consider spatial locality. But locality is essential for hand detection in a scene. Furthermore, most of them are defined based on similarity instead of semantics, ignoring the contextual cues obtained by reasoning about the spatial relationships between semantically related entities. So, here, we have designed a method for hand segmentation in images as well as videos using attention-based semantic segmentation and subsequent recognition through deep networks.

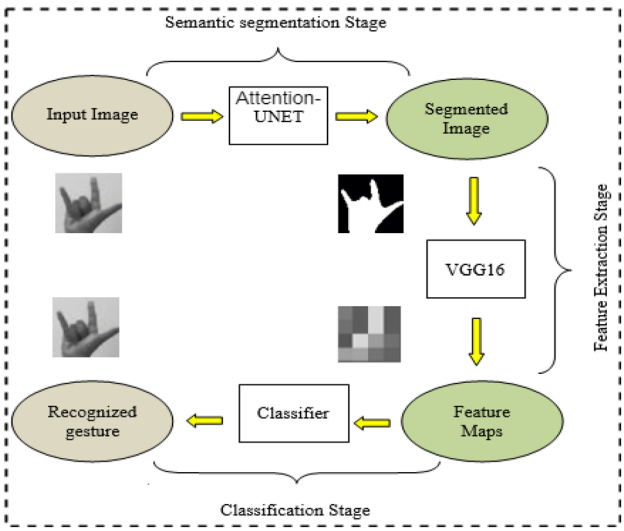

Fig. 5 Block diagram of the workflow for static hand gesture recognition

Fig. 6 Examples showing the challenges of HGR hand dataset: **a** clutter backgrounds, **b** different skin color, **c** weak illumination conditions

3 Methodology

This section describes the workflow of the proposed method. The proposal has two sections: attention-based semantic segmentation and classification. Here, we have semantically segmented the static (still images) or dynamic (video frames or image sequences) gestures, and segmented masks are subsequently fed to a classifier for recognition. The following subsections shed light on the models as well as the work process for semantic segmentation and classification independently.

3.1 Semantic segmentation

As already mentioned, the objective of semantic segmentation is to assign labels to each pixel and then link the similar pixels into a group. Here we employed UNet architecture for obtaining a segmented mask image, where the hand portion is segmented from the background.

3.1.1 UNet structure

The network has a progressively narrowing structure, which tends to encode the input into a fine to the coarse manner, followed by a decoding structure that broadens progressively. The decoder adds skip connections. This makes the segmented result more accurate block after block, as the finer details from the earlier layers of the encoding structure coalesce with the layers of the decoding path (up6 with dropout4, up7 with conv3, up8 with conv2 and up9 with conv1).

The architecture consists of convolutional blocks each on the encoding path as well as on the decoding path. Each block of the encoding path contains two convolutional layers with a receptive field of size (3×3) followed by a max-pooling layer with a (2×2) window. Zero padding is done for the convolutional operation to maintain the same spatial dimensions of the output feature map as the input. The layer succeeding the downsampling (max-pooling) layer contains twice the number of channels in the previous layer. Also, a dropout layer is included in the last and penultimate block of the encoder, which is basically to prevent the development of the co-dependencies amongst neurons.

In the decoder, the max-pooling layer is replaced by the upsampling layer of window size (2×2). After each

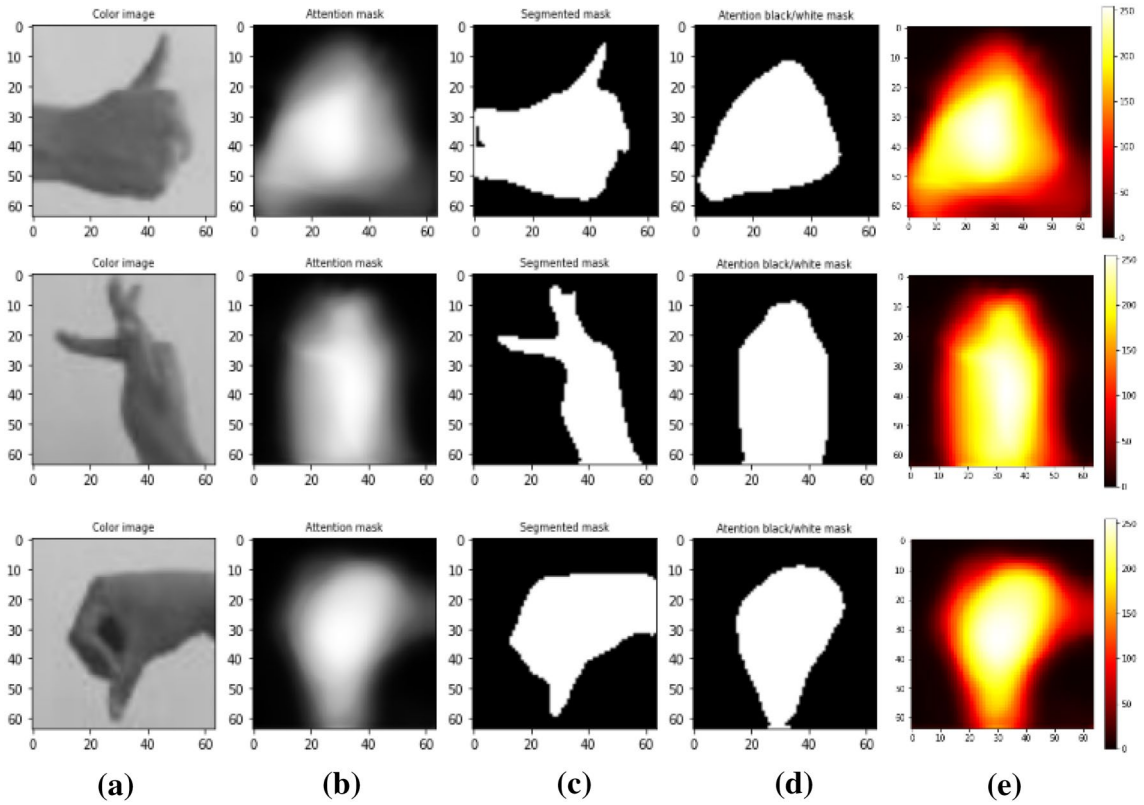

Fig. 7 Semantic segmentation results showing attention masks for the Brazilian dataset: **a** shows the gesture images, **b** shows the attention masks, **c** shows the segmented masks, **d** shows the black/white attention masks and **e** shows the heat-map of the attention masks

upsampling layer, there is a convolutional layer to match the dimension of the feature map of the layer on the encoding path, which is concatenated with this layer. This convolutional layer reduces the feature channel to half of the number of channels in the previous layer. This is basically for the skip connection. The activation function for each convolutional layer is 'ReLU' except the last layer, which is 'Sigmoid'.

3.1.2 Re-designed skip path with attention module

In the original UNet, the skip-connected feature maps of the decoder are received straight from the encoder. But, here, we have made some modifications in skip connection by inserting an attention unit called Convolutional Block Attention Module (CBAM) between encoder and decoder. Dataflow passes through the chain of convolutional layers using CBAM in the skip-connections. With the insertion of the attention module, the semantic distance between the encoder and the decoder maps is likely to decrease. The architecture for the modified network is shown in Fig. 3.

3.1.3 Convolutional Block Attention Module (CBAM)

Usually, the attention mechanism is placed after the convolutional layer; then, the features to which the attention network pays attention and the features extracted by the neural network are input into the next convolutional layer. So, an attention network can be understood as a weighting operation that operates on different feature regions. Convolutional Block Attention Module (CBAM) (Woo et al. 2018) emphasize meaningful features along those two principal dimensions: channel and spatial axes. To achieve this, we sequentially apply channel and spatial attention modules, so that each of the branches can learn 'what' and 'where' to attend in the channel and spatial axes respectively. Given an input image, two attention modules, channel and spatial compute complementary attention, focusing on 'what' and 'where' respectively. The channel attention block is proposed to integrate the interaction among the inter-channel feature maps. It is employed to enhance the vital information of a feature map of the object i.e. hand. The spatial attention mechanism focuses on the local regions of a feature map. Thus, this module is employed to preserve the location of the hand information (ROI) in the feature maps. Considering this, two modules can be placed in a parallel or sequential manner. We found in the literature that the sequential arrangement (as shown in Fig. 4) gives a better result than a parallel arrangement Woo et al. (2018).

Given an intermediate feature map $F \in \Re^{C\times H\times W}$ as input, CBAM sequentially infers a 1D channel attention map $M_c \in \Re^{C\times 1\times 1}$ and a 2D spatial attention map $M_s \in \Re^{1\times H\times W}$ as illustrated in Fig. 4. The overall attention process can be summarized as:

$$F' = M_c(F) \otimes F, \tag{1}$$

$$F'' = M_s\left(F'\right) \otimes F' \tag{2}$$

where $\otimes$ denotes element-wise multiplication. During multiplication, the attention values are broadcasted (copied) accordingly: channel attention values are broadcasted along the spatial dimension, and vice versa. F'' is the final refined output. The zoomed section of the channel and the spatial portion in Fig. 4 depicts the computation process of each attention map. The following describes the details of each attention module.

1. **Channel attention module:** A channel attention map is exploiting the inter-channel relationship of features. The spatial information of a feature map is aggregated by using both average-pooling and max-pooling operations, generating two different spatial context descriptors: F^c_{avg} and F^c_{max}, which denote average-pooled features and max-pooled features respectively. Both descriptors are then forwarded to a shared network to produce our channel attention map $M_c \in \Re^{C\times 1\times 1}$. The shared network is composed of a multi-layer perceptron (MLP) with one hidden layer. To reduce parameter overhead, the hidden activation size is set to $\Re^{C/r\times 1\times 1}$, where r is the reduction ratio. After the shared network is applied to each descriptor, we merge the output feature vectors using element-wise summation. In short, the channel attention is computed as:

$$M_c(F) = \sigma(MLP(AvgPool(F)) + MLP(MaxPool(F))) \tag{3}$$

$$= \sigma\left(W_1\left(W_0\left(F^c_{avg}\right)\right) + W_1\left(W_0\left(F^c_{max}\right)\right)\right) \tag{4}$$

where σ denotes the sigmoid function, $W_0 \in \Re^{C/r\times C}$, and $W_1 \in \Re^{C\times C/r}$. Note that the MLP weights, W_0 and W_1, are shared for both inputs and the ReLU activation function is followed by W_0.

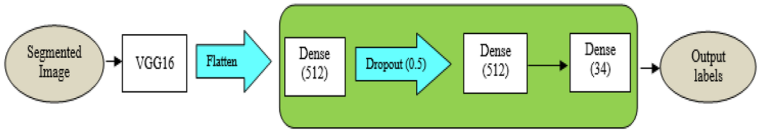

Fig. 8 Block diagram for the classification process for static gestures

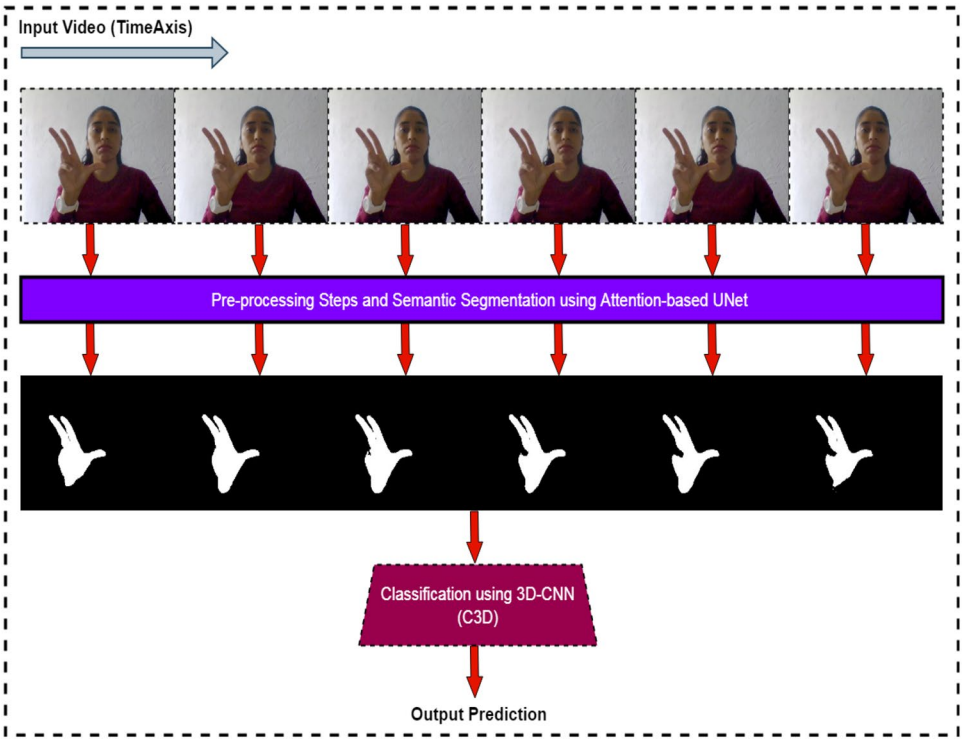

Fig. 9 The workflow for dynamic hand gesture recognition

Fig. 10 Some examples showing the challenges of IPN hand dataset: **a** clutter backgrounds, **b** natural interaction with objects, **c** weak illumination conditions

2. **Spatial attention module:** The spatial attention map is generated by utilizing the inter-spatial relationship of features. On the concatenated feature descriptor, we apply a convolution layer to generate a spatial attention map $M_s(F)^{H\times W}$ which encodes where to emphasize or suppress. For this, we aggregate channel information of a feature map by using two pooling operations, generating two 2D maps: $F^s_{avg} \in \Re^{1\times H\times W}$ and $F^s_{max} \in \Re^{1\times H\times W}$. Each denotes average-pooled features and max-pooled features across the channel. Those are then concatenated and convolved by a standard convolution layer, producing our 2D spatial attention map. In short, the spatial attention is computed as:

$$M_s(F) = \sigma\left(f^{7\times 7}\left[(AvgPool(F); (MaxPool(F)\right]\right) \quad (5)$$

$$= \sigma\left(f^{7\times 7}\left[F^s_{avg}; F^s_{max}\right]\right) \quad (6)$$

where σ denotes the sigmoid function and $f^{7\times 7}$ represents a convolution operation with the filter size of 7×7.

3.2 Static hand gesture recognition

For static gestures, above mentioned attention-based UNet architecture is employed to obtain the segmented masks from the still images. Then these segmented images are fed to a neural network for feature extraction and finally extracted features are fed to a classifier for recognizing the gestures. The workflow is shown in Fig. 5. Though this model is simple, experimental results have demonstrated that

it is able to achieve state-of-the-art (SOTA) results. The following sections also throws light into the dataset used and the importance of data augmentation process.

3.2.1 Datasets used

The dataset used for static hand gesture recognition is the Brazilian Sign Language (Libras) dataset (Bastos et al. 2015). The official sign language of Brazil is called Libras. It consists of a total of 9600 images, evenly distributed among 40 classes. The different classes represent letters of Libras alphabets (22 classes), numbers (6 classes), and a few words (12 classes). Though there are 40 classes in the original Brazilian Sign Language database, the experiments were carried out for 34 compatible classes. Each class contains the segmented masks depicting the skin region of the gesture. Each image in the dataset has a resolution of $50 \times 50 \times 3$, and the images are captured considering small variations in the illumination as well as the hand posture and size. The background is kept uniform, without any cluttering objects. Since this dataset is comparatively simple, hence the segmentation has also been tried on another dataset namely Hand Gesture Recognition (HGR) dataset (Kawulok et al. 2014). It has different clutter background, varying illumination etc (shown in Fig. 6) and our method has shown satisfactory performance on both the databases.

3.2.2 Data augmentation

Data augmentation plays a crucial role in deep learning approaches, as the number of data samples required in deep learning techniques is very high. Data augmentation generates more training data out of the few training samples available, generally employing affine transformation to the samples. Thus, the model is exposed to every possible aspect of the distribution of the data samples and helps it generalize the new data. For data augmentation, several transforms are used like rotation up to 20°, width shift (up to 0.2 range), height shift (up to 0.2 range), sheer (up to 0.2 range), zoom mode (up to 0.2), fill mode on nearest data, etc. This newly generated data also contributes to the required robustness against the variation of scale, translation and rotation. The training sample size for the classifier network is increased to 106,800 images after data augmentation.

3.2.3 Generation of segmented masks

Till now, the architecture of the model has been discussed, and in this section, the process of generation of the segmented masks is highlighted. Speaking of the procedure, the original images are passed through the attention-based UNet architecture, and the output images are obtained through the final convolutional layer of the decoder part of the trained structure. After a sufficient number of images are generated through the data augmentation process, the images are fed to the UNet model for training. For the training process, the input images are arranged into two sections, one containing the original images, and the other containing the segmented masks (included in the dataset in grayscale). During the training process, the architecture learns to focus on the ROI portion and when the test images are fed, it can segment out the hand portion. The segmented image has its advantage, as

Table 1 Comparison of the segmentation performance measures for the Brazilian Sign Language (Libras) dataset (Bastos et al. 2015)

	Bastos et al. (2015)	Proposed method (without attention module)	Proposed method (with attention module)
Jaccard (IoU)	0.76	0.89	**0.98**
PSNR	15.11	15.62	**17.32**

Bold values mean the best results obtained by our method

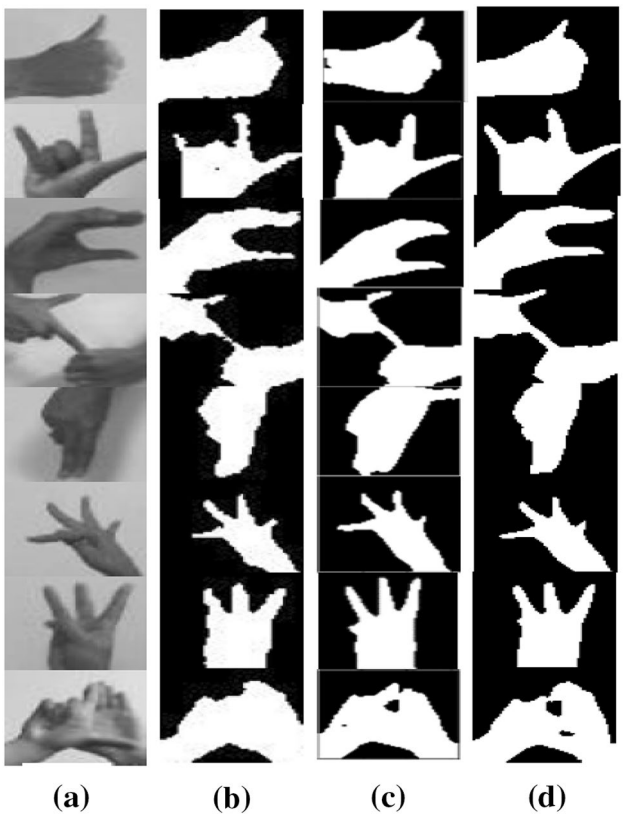

Fig. 11 Comparison among semantic segmentation outputs for static gestures: **a** shows the gesture images, **b** shows the segmented masks obtained by Bastos et al. (2015), **c** shows the segmented masks obtained through UNet without attention mechanism and **d** shows the segmented masks obtained through attention-based UNet

it has only two regions (i.e., the hand and the background), and it is free from variations in the intensity values within the same region (shown in Fig. 7). The shading and the depth information are also not present, which may increase the complexity and in turn, the time to process the images in the next stage i.e. classification.

Since this dataset consists of two regions—the hand (i.e., the foreground) and the background, it is, in fact, a sort of binary segmentation problem, assigning a certain range of intensity values to the foreground and the rest to the background. During the training process, the parameters are optimized using the Adam Optimization method (Kingma and Ba 2014), where the learning rate is kept at 0.0001 and the hyperparameters β_1 and β_2 are kept at 0.9 and 0.999 respectively. The network is trained for twenty-five epochs with the loss function being the binary cross-entropy, which is given as:

$$Loss(y, \hat{y}) = -\frac{1}{N}\sum_{i=1}^{N}\left[ylog\hat{y}_i + (1-y)log\left(1-\hat{y}_i\right)\right]\Bigg)$$

where, N is the number of samples and $\hat{y}$ is the predicted value for a true value y.

3.2.4 Classification

The classification problem in this work is a multi-class problem with a limited amount of data. Hence, we opted for a pre-trained network, i.e., VGG16 (Simonyan and Zisserman 2014), trained on ImageNet dataset. The final images obtained after segmentation have two regions—the hand and the background. Since the network is trained on RGB images, so these segmented images are converted into 3-channel images using the pseudo coloring method. It is a minor pre-processing step before feeding them into the classification stage, which would recognize the different gestures. The objective of using the pre-trained network is to exploit the spatial hierarchy of features learned by the network, which can be considered as generic and reusable representation of data. Then the classifier on top of the VGG16 model is replaced by our classifier to learn the specific features of the classes of used database. This classifier consists of a dense layer containing 512 neurons and ReLU as the activation function, which is followed by a dropout layer to fight the situation of overfitting. The final layer of the classifier is a dense layer, with 34 neurons giving us the respective class labels with softmax function being used for activation. The block diagram of the classifier is shown in Fig. 8. The softmax activation function returns the probability score of the different classes where the largest value gives the class predicted. It is defined as:

$f : \mathfrak{R}^N \rightarrow \mathfrak{R}^N$, such that $f(x_i) = \frac{e^{x_i}}{\sum_{j=1}^{N} e^{x_j}}$, for $i = 1, \ldots, N$ and $\forall x_i \in \mathfrak{R}^N$

Similar to the semantic segmentation step, the weights are optimized similarly, but the performance measure is being changed from binary to categorical cross-entropy, and it is given as:

$$Loss(y, \hat{y}) = -\sum_{j=1}^{M}\sum_{i=1}^{N} y_{ij} log\hat{y}_{ij} \tag{7}$$

where, N is the number of samples, M is the number of classes and $\hat{y}$ is the predicted value for a true value y.

Fig. 12 Semantic segmentation outputs for HGR dataset: 1st column shows the gesture images, 2*nd* column shows the segmented masks obtained by Kawulok et al. (2014), 3*rd* column shows the segmented masks obtained through attention-based UNet (our method)

3.3 Dynamic hand gesture recognition

This attention-based model is modified and extended for dynamic hand gesture recognition as well. A dynamic hand

Table 2 Comparison of Accuracy Performance (%) for the Brazilian Sign Language (Libras) dataset (Bastos et al. 2015)

	Without pre-segmentation	With pre-segmentation (without attention module)	With pre-segmentation (with attention module)
Bastos et al. (2015)	–	97.14%	–
Our method	93.28%	98.97%	**99.50%**

Bold values mean the best results obtained by our method

Table 3 Table showing the comparison between Bastos et al. and the proposed method for static gestures

Class	Bastos et al. (2015)	Proposed method
1	100	100
2	95.83	100
4	99.16	100
5	100	100
7	96.67	100
Adult	95.83	100
America	100	100
Plane	100	100
C	90	100
House	100	100
D	96.67	100
E	98.33	100
G	99.16	100
Gas	100	100
I	89.16	100
Identity	98.33	96
Together	98.33	100
L	95	94
Lei	99.16	100
N	99.16	100
O	96.67	96
P	100	100
Word	100	100
Stone	99.16	100
Little	90.83	100
Q	98.83	100
R	96.67	100
T	100	100
U	90.83	100
V	95.83	100
Verb	95	97
W	95	100
X	98.33	100
Y	95	100
Average	97.14	99.50

gesture video can be considered as a sequence of several static hand gestures. This sequence contains enough information to be used for dynamic hand gesture recognition. So, in this section, we first get the segmented masks for the sequential images after some pre-processing steps. We have chosen a 3D-CNN (C3D) network as a classifier that can capture spatial as well as temporal information of a video. The workflow is shown in Fig. 9.

3.3.1 Dataset

The dataset used for dynamic hand gesture recognition is the IPN hand dataset (Benitez-Garcia et al. 2021). It is a recent dataset with 13 static and dynamic gesture classes for interaction with touchless screens. It contains 4218 gesture instances and 800,491 frames from 50 subjects in 28 diverse scenes. The recordings have different clutter backgrounds with varying illumination in both static/dynamic scenes as shown in Fig. 10. All videos were recorded using a normal PC or laptop in the resolution of 640×480 at 30 fps. The recorded gestures can be used to control the pointer and actions focused on the interaction with touchless screens. Apart from the RGB frames, real-time optical flow and hand segmentation results are also available with the database.

3.3.2 Pre-processing

To effectively train the CNNs, we have adopted a few preprocessing steps as performed by Sharma and Kumar (2021). To reduce the chances of CNNs being trained on noisy data some filtering operations are also performed. It is also noticeable that preprocessing is only done with the training data to reduce the elements resulting in degraded performance and it is a prior expense of time. The various pre-processing stages are outlined below

1. Each gesture video is first converted into several frames, then each frame is processed individually. Each frame is resized to 512×384 and normalized to [0, 1] to reduce the computation.
2. The unwanted noise and spots in the frame are removed using median filtering.
3. The illumination variations in the frame are canceled out using Histogram equalization.
4. The hand portions in the frames are then segmented out using previously explained attention-based UNet structure.
5. The segmented frames are then combined again to form the video sequence for training the 3D-CNNs.

3.3.3 Segmentation

Static gesture is represented by single still image and dynamic gesture is nothing but a sequence of images. The segmentation method used for dynamic gestures is exactly the same as for the static gesture. The only difference with dynamic gesture is that the semantic segmentation mask is obtained for each streaming frame. Finally, all the segmented frames are combined into a single video to be processed by the next classification stage.

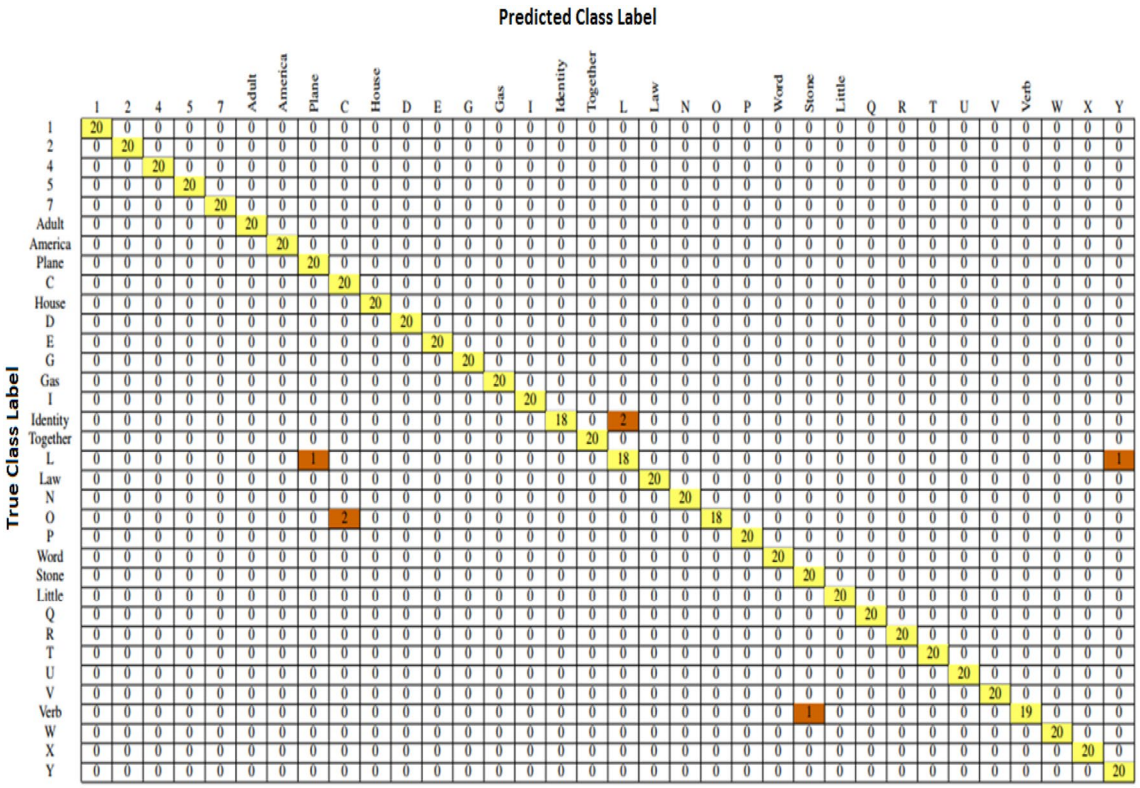

Fig. 13 Confusion matrix for static gestures

3.3.4 Classification

The original C3D (Tran et al. 2015) was designed for RGB videos. The number of parameters of the networks depends on the resolution of input frames. The original C3D was trained on the large-scale dataset Sport1M (Karpathy et al. 2014), which consists of 1.1M videos downloaded from YouTube consisting of 487 sports classes. 2D-CNN is extended to a 3D-CNN by incorporating the temporal dimension of a video sequence. In 2D-CNNs, the dimension of each feature map is $c \times h \times w$, where c represents the number of filters in the convolutional (conv) layer, h and w represents the height and width of the feature map. In 3D-CNNs, the dimension of each feature map is $c \times l \times h \times w$, where additional parameter l represents the number of frames. This network extracts the features which are compact and generic while being discriminative. As we worked on two smaller databases, a slightly different architecture with 5 conv layers is employed which has a smaller number of parameters compared to the original C3D (Tran et al. 2015) with 8 conv layers. The proposed network has 5 space-time conv layers with 64, 128, 256, 256, 256 kernels. Each conv layer is followed by a rectified linear unit (ReLU) and a space-time max-pooling layer. All 3D convolution kernels are of size $3 \times 3 \times 3$, that gives the best performance (Tran et al. 2015) with stride $1 \times 1 \times 1$. Max pooling kernels are of size $2 \times 2 \times 2$ except for the first, where it is $2 \times 2 \times 1$ and stride is $2 \times 2 \times 1$. The conv layers are followed by two dense layers with 2048 and 1024 neurons and ReLU as the activation function. To avoid over-fitting while learning, there is a dropout in each dense layers. The parameter of dropout is set to 0.4, which means the layer randomly excludes 40% of neurons. The final dense layer of the classifier has 13 neurons giving us the respective class labels where softmax function is used for activation.

4 Experimental results

This section gives an idea of the work performed and analysis of the results obtained. The entire experiment was done in a python environment, taking the help of the NVIDIA Tesla K80 GPU in Google Colab, especially, for classification.

4.1 Results for static gestures

The experimentation is carried out on the Brazilian Sign Language dataset given by Bastos et al. (2015) for static hand gesture recognition. The results for this section is given in two subsections namely for segmentation and classification.

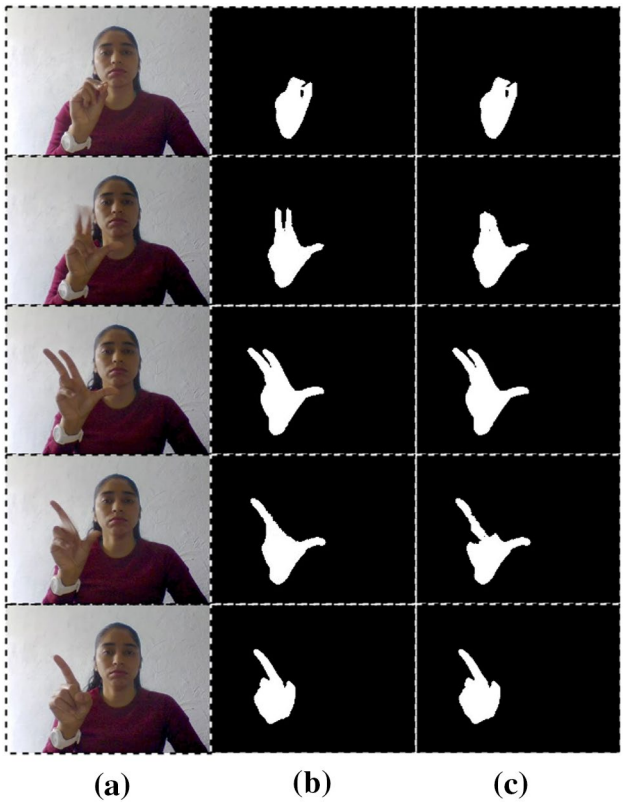

Fig. 14 Semantic segmentation output for a few dynamic gesture frames from IPN hand dataset: **a** shows the gesture images, **b** ground truths, and **c** shows the corresponding segmented masks obtained by our method

Table 4 Comparison of segmentation performance measures for IPN hand dataset

	Proposed method (without attention module)	Proposed method (with attention module)
Jaccard (IoU)	0.86	**0.93**
PSNR	13.60	**19.52**

Bold values mean the best results obtained by our method

4.1.1 Results from the segmentation stage

The first step of the proposed method is to find the segmented masks of the dataset samples. The input image is first resized into $64 \times 64 \times 1$ image, and then, it is passed through the UNet architecture along with the corresponding segmented mask provided by the dataset. The training process was executed for 25 epochs in a regular PC of 8 GB RAM and 3.5 GHz processor speed. It took about 413 s, i.e., 103 ms/image to segment 4000 test images with an accuracy of 96.33 %. The comparison of the segmented masks provided by Bastos et al. (2015) and the generated segmented masks are shown in Fig. 11. From the figure, it is evident that, the segmentation results obtained in this work is better than the ones obtained by Bastos et al. (2015). It may be subjective, hence, in order to support the results obtained, mean Jaccard Similarity Index and the mean PSNR values are calculated.

Jaccard Similarity Index, also known as Intersection over Union (IoU), computes the similarities between the elements of two sets. It ranges in the interval [0,1], with 0 referring the sets to be disjoint and 1 signifying the exact match. Mean Jaccard Similarity Index is defined as:

$$J = \frac{1}{M}\sum_{j=1}^{M}\frac{\sum_i min\left(I_i, G_i\right)}{\sum_i max\left(I_i, G_i\right)} = \frac{I \cap G}{I \cup G}$$

where M is the number of classes and I and G are the vectorized segmented mask and ground truth respectively. It measures the overlap between two bounding boxes I and G as the ratio of the total covered area.

Another measure used is PSNR, the ratio of peak signal power to noise power. The mean PSNR measure is given as:

$$PSNR = \frac{1}{M}\sum_{j=1}^{M} 10log\left(peak^2/MSE(f, gt)\right)$$

$$MSE = \frac{1}{N_1 N_2}\sum_{x=1}^{N_1}\sum_{y=1}^{N_2}(f(x,y) - gt(x,y))^2$$

where, MSE stands for Mean Square Error and f and gt represents the image and the ground truth respectively. N_1 and N_2 are the number of rows and columns of the image.

Table 1 gives a comparison between the segmentation result of this work and the skin segmentation using multilayer perceptron adopted by Bastos et al. (2015). The table shows a slightly better result for the PSNR measure, but the proposed work has achieved a significant improvement in the Jaccard Similarity performance measure which justifies the betterment in the subjective results shown in Fig. 11.

The segmentation outputs of HGR dataset is shown in Fig. 12. Kawulok et al. (2014) is a color-based skin segmentation method. But the accuracy of skin detection methods is severely affected constraints like the presence of skin-like

Table 5 Table showing the individual class accuracy for IPN hand dataset

Class	B0A	B0B	G01	G02	G03	G04	G05	G06	G07	G08	G09	G10	G11
Accuracy (%)	89.21	86.23	90.20	87.30	88.72	87.52	86.80	87.50	88.23	88.50	87.60	87.91	87.75

	B0A	B0B	G01	G02	G03	G04	G05	G06	G07	G08	G09	G10	G011
B0A	62	0	0	0	0	0	0	0	0	0	0	2	0
B0B	0	59	0	3	0	0	0	0	0	0	2	0	0
G01	0	0	61	0	0	0	0	0	0	3	0	0	0
G02	0	2	0	60	0	0	0	0	0	0	2	0	0
G03	0	0	0	0	62	0	0	0	1	0	0	1	0
G04	0	0	0	0	0	64	0	0	0	0	0	0	0
G05	0	0	0	0	0	0	61	1	0	0	0	2	0
G06	0	0	0	0	0	0	2	60	0	0	0	2	0
G07	0	0	0	0	1	0	0	0	63	0	0	0	0
G08	1	0	2	0	0	0	0	0	0	61	0	0	0
G09	0	0	0	2	0	0	0	0	0	0	62	0	0
G10	0	0	0	0	0	0	0	0	0	0	0	62	2
G11	0	0	0	0	0	0	0	0	0	0	0	1	63

Fig. 15 Confusion matrix for dynamic gestures

colors in the background, illumination variation, background complexity, and occlusion etc. Whereas UNET structure is based on CNN in encoder-decoder form. So, UNET has provided better results compared to skin-based segmentation. The proposed CBAM-based attention mechanism has given the ability to focus and guide the learning of information for segmentation in UNet structure. Hence, CBAM-based UNET gives quite satisfactory performance which is seen in our experimentation.

4.1.2 Results from the classification stage

After segmentation, the images are passed through the classification stage. In Bastos et al. (2015), the feature vector comprising of two shape descriptors—Histogram of Oriented Gradients (HOG) and Zernike Invariant Moments (ZIM), are used for training and testing a two-stage Multi-Layer Perceptron (MLP) classifier. This method produced a high recognition rate. Since the proposed method also uses the same dataset, the work in Bastos et al. (2015) is used for comparing the classification results. The original gesture images could also have been passed through the classifier without going through the segmentation stage. But, through experimentation, it was found that with the inclusion of the attention-based semantic segmentation step the accuracy of classification has increased from 93.28 to 99.50% (Table 2). This also helped achieve much better results compared to the prior work by Bastos et al.

The classifier training was done on 106800 images for 20 epochs and the model with the best accuracy was saved. It took about 280 s for each epoch in the NVIDIA Tesla K80 GPU in Google Colab. For testing, 200 images were considered from 34 classes each in a 10-fold cross-validation pattern. Table 3 shows the comparison of the results by Bastos et al. and the proposed method.

Figure 13 gives the detailed confusion matrix of the testing phase. The yellow colored cells show the true positives while the brown-colored cells represent the misclassified samples.

4.2 Results for dynamic gestures

The experimentation is carried out on the IPN hand dataset Benitez-Garcia et al. (2021) for dynamic hand gesture recognition. The results for this section are given in two subsections namely for segmentation and classification.

4.2.1 Results from the segmentation stage

The continuous video sequences are segmented into isolated gesture samples based on the beginning and ending frames by manual annotation. Since we classify on a 3D-CNN model, first, the semantic segmentation masks are evaluated for each streaming frame. The subjective comparison of the ground truth segmented mask provided by Benitez-Garcia et al. (2021) and the generated segmented masks are shown in Fig. 14. It is seen that when the frame is shaky, then segmentation is not so accurate. For quantitative analysis, the mean Jaccard Similarity Index and the mean PSNR values are given in Table 4.

4.2.2 Results from the classification stage

The task of the classifier is to predict class labels for each gesture sample as shown in Fig. 9. We use classification accuracy, which is the percent of correctly labeled examples as an evaluation metric for this classification task. Table 5 shows the individual class accuracy using the proposed method.

The confusion matrix of the testing phase for IPN dataset is shown in Fig. 15. From the 324 instances of each class in the dataset, 64 instances i.e. almost 20% are used for testing.

Table 6 Comparison of performance measures (% accuracy) for Isolated IPN Gestures

Method	Modality	Accuracy (%)
C3D (Benitez-Garcia et al. 2021)	RGB	77.75
ResNeXt-101 (Benitez-Garcia et al. 2021)	RGB-Flow	86.32
ResNeXt-101 (Benitez-Garcia et al. 2021)	RGB-Seg	84.77
ResNet-50 (Benitez-Garcia et al. 2021)	RGB-Flow	74.65
ResNet-50 (Benitez-Garcia et al. 2021)	RGB-Seg	75.11
Our Method	Segmented Masks	87.95

The yellow colored cells show the true positives while the brown-colored cells represent the misclassified samples.

We have also compared our results in Table 6 with Benitez-Garcia et al. (2021) where authors have used either only RGB frames as a single input or RGB frames with segmented masks (RGB-Seg) or RGB frames with optical flow (RGB-Flow) as multi-modal inputs. From Table 6, it is clear that our method has achieved SOTA performance. This is due to the effective attention-based segmentation process which has led to a better classification result.

5 Conclusion

Motivated by the success of the attention-based methods, and considering it from the view of focus and region-wise representations, we have embedded an attention-based module in semantic segmentation to capture global contexts from the perspective of space and channel for better feature representations. CNN proves to be a magnificent tool for classification, whose benefit can also be exploited for image segmentation tasks. Hence, two of the CNN-based deep models—UNet and VGG16 are employed in this paper concerning semantic segmentation and classification respectively to achieve state-of-the-art results for static hand gesture recognition problem. An attention-based UNet model is used for segmenting the gesture images in the pre-processing stage, which is basically an encoding-decoding structure. It adds fine information to the coarse layers, and thus, helps in improving the segmentation results. Moreover, benefiting from the attention mechanism, UNet can be used more efficiently and effectively than other segmentation methods. The hierarchical pattern learned by the UNet projects accurate visualization of the problem at hand. Speaking of the classification stage, a pre-trained VGG16 network is used to extract the features of the segmented images, and the extracted feature maps are passed through the designed classifier. This same process has also been extended for dynamic hand gesture recognition as well. Here, in place of 2D-CNN, a 3D-CNN is used as a classifier since it can capture more subtle spatio-temporal features. Comprehensive empirical results verify that our proposed model is better than state-of-the-art. As a future direction, the proposed method can be modified for implementation in real-time applications. Besides, for further evaluation, our model can be extended to other image segmentation tasks like brain tumor segmentation in the medical domain.

Funding No funding was received for this work.

Data availability The databases used in the experiments are - the Brazilian Sign Language dataset (http://sites.ecomp.uefs.br/lasic/projetos/libras-dataset) (Bastos et al. 2015), HGR dataset (https://sun.aei.polsl.pl/mkawulok/gestures/) (Kawulok et al. 2014) and IPN hand dataset (https://gibranbenitez.github.io/IPN_Hand/) (Benitez-Garcia et al. 2021) and these are publicly available databases.

Declarations

Conflict of interest The authors declare that they have no Conflict of interest.

Human and animal rights In this research work, there is no involvement of human participants and/or animals in any part of the experimentation.

References

Abdul W, Alsulaiman M, Amin SU, Faisal M, Muhammad G, Albogamy FR, Bencherif MA, Ghaleb H (2021) Intelligent real-time Arabic sign language classification using attention-based inception and bilstm. Comput Electric Eng 95:107395

Bastos IL, Angelo MF, Loula AC (2015) Recognition of static gestures applied to Brazilian sign language (libras). In: 2015 28th SIBGRAPI Conference on Graphics, Patterns and Images, pp. 305–312. IEEE

Benitez-Garcia G, Olivares-Mercado J, Sanchez-Perez G, Yanai K (2021) Ipn hand: a video dataset and benchmark for real-time continuous hand gesture recognition. In: 2020 25th International Conference on pattern recognition (ICPR), pp 4340–4347. IEEE

Chakraborty BK, Sarma D, Bhuyan M, MacDorman KF (2017) Review of constraints on vision-based gesture recognition for human-computer interaction. IET Comput Vis 12(1):3–15

Chen L-C, Papandreou G, Kokkinos I, Murphy K, Yuille AL (2014) Semantic image segmentation with deep convolutional nets and fully connected crfs. arXiv preprint arXiv:1412.7062

Chen L-C, Papandreou G, Schroff F, Adam H (2017a) Rethinking atrous convolution for semantic image segmentation. arXiv preprint arXiv:1706.05587

Chen L-C, Papandreou G, Kokkinos I, Murphy K, Yuille AL (2017b) Deeplab: semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. IEEE Trans Pattern Anal Mach Intell 40(4):834–848

Chen L, Zhang H, Xiao J, Nie L, Shao J, Liu W, Chua T-S (2017c) Sca-cnn: spatial and channel-wise attention in convolutional networks for image captioning. In: Proceedings of the IEEE Conference on computer vision and pattern recognition, pp 5659–5667

Chen L-C, Zhu Y, Papandreou G, Schroff F, Adam H (2018) Encoder-decoder with atrous separable convolution for semantic image segmentation. In: Proceedings of the European Conference on computer vision (ECCV), pp 801–818

D'Eusanio A, Simoni A, Pini S, Borghi G, Vezzani R, Cucchiara R (2020) A transformer-based network for dynamic hand gesture recognition. In: 2020 International Conference on 3D Vision (3DV), pp. 623–632. IEEE

Dhingra N, Kunz, A (2019) Res3atn-deep 3d residual attention network for hand gesture recognition in videos. In: 2019 International Conference on 3D vision (3DV), pp 491–501. IEEE

Dutta HPJ, Sarma D, Bhuyan MK, Laskar RH (2020) Semantic segmentation based hand gesture recognition using deep neural networks. In: 2020 National Conference on Communications (NCC), pp 1–6, 2020. IEEE

Fu J, Liu J, Tian H, Li Y, Bao Y, Fang Z, Lu H (2019) Dual attention network for scene segmentation. In: Proceedings of the IEEE/

CVF Conference on computer vision and pattern recognition, pp 3146–3154
Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville A, Bengio Y (2014) Generative adversarial nets. In: Advances in neural information processing systems, pp 2672–2680
He K, Gkioxari G, Dollár P, Girshick R (2017) Mask r-cnn. In: Proceedings of the IEEE International Conference on computer vision, pp 2961–2969, 2017
Huang H, Lin L, Tong R, Hu H, Zhang Q, Iwamoto Y, Han X, Chen Y-W, Wu J (2020) Unet 3+: A full-scale connected unet for medical image segmentation. In: ICASSP 2020-2020 IEEE International Conference on acoustics, speech and signal processing (ICASSP), pp 1055–1059. IEEE
Hu J, Shen L, Sun G (2018) Squeeze-and-excitation networks. In: Proceedings of the IEEE Conference on computer vision and pattern recognition, pp 7132–7141
Jaderberg M, Simonyan K, Zisserman A et al (2015) Spatial transformer networks. Adv Neural Inf Process Syst 28:2017–2025
Karpathy A, Toderici G, Shetty S, Leung T, Sukthankar R, Fei-Fei L (2014) Large-scale video classification with convolutional neural networks. Proceedings of the IEEE Conference on computer vision and pattern recognition, pp 1725–1732
Kavyasree V, Sarma D, Gupta P, Bhuyan M (2020) Deep network-based hand gesture recognition using optical flow guided trajectory images. In: 2020 IEEE Applied Signal Processing Conference (ASPCON), pp 252–256. IEEE
Kawulok M, Kawulok J, Nalepa J, Smolka B (2014) Self-adaptive algorithm for segmenting skin regions. EURASIP J Adv Signal Process 2014:1–22
Kingma DP, Ba J (2014) Adam: a method for stochastic optimization. arXiv preprint arXiv:1412.6980
Lea C, Flynn MD, Vidal R, Reiter A, Hager GD (2017) Temporal convolutional networks for action segmentation and detection. In: Proceedings of the IEEE Conference on computer vision and pattern recognition, pp 156–165, 2017
Li H, Xiong P, An J, Wang L (2018) Pyramid attention network for semantic segmentation. arXiv preprint arXiv:1805.10180
Li C, Tan Y, Chen W, Luo X, He Y, Gao Y, Li F (2020) Anu-net: attention-based nested u-net to exploit full resolution features for medical image segmentation. Comput Graph 90:11–20
Li X, Hou Y, Wang P, Gao Z, Xu M, Li W (2021). Trear: Transformer-based rgb-d egocentric action recognition. IEEE Transactions on Cognitive and Developmental Systems, 14(1),246–252.
Long J, Shelhamer E, Darrell T. Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE Conference on computer vision and pattern recognition, pp 3431–3440, 2015
Narasimhaswamy S, Wei Z, Wang Y, Zhang J, Hoai M (2019) Contextual attention for hand detection in the wild. In: Proceedings of the IEEE/CVF International Conference on computer vision, pp 9567–9576
Narayana P, Beveridge R, Draper BA (2018) Gesture recognition: focus on the hands. In: Proceedings of the IEEE Conference on computer vision and pattern recognition, pp 5235–5244
Pisharady PK, Vadakkepat P, Loh AP (2013) Attention based detection and recognition of hand postures against complex backgrounds. Int J Comput Vis 101(3):403–419
Ren S, He K, Girshick R, Sun J (2015) Faster r-cnn: towards real-time object detection with region proposal networks. Adv Neural Inf Process Syst 28:91–99
R-FCN, D. A. I. J. (2016) Object detection via region-based fully convolutional networks. In Proceedings of IEEE International Conference on Computer Vision. Piscataway: IEEE Press, pp 1–9
Ronneberger O, Fischer P, Brox T (2015) U-net: convolutional networks for biomedical image segmentation. In: International Conference on medical image computing and computer-assisted intervention, pp 234–241, 2015. Springer
Sarma D, Bhuyan MK (2018) Hand gesture recognition using deep network through trajectory-to-contour based images. In: Proceedings of the IEEE India Council International Conference (INDICON), 2018
Sarma D, Bhuyan M (2021) Methods, databases and recent advancement of vision-based hand gesture recognition for hci systems: a review. SN Comput Sci 2(6):1–40
Sarma D, Bhuyan M (2022) Hand detection by two-level segmentation with double-tracking and gesture recognition using deep-features. Sens Imaging 23(1):1–29
Sarma D, Kavyasree V, Bhuyan M (2022) Two-stream fusion model using 3d-cnn and 2d-cnn via video-frames and optical flow motion templates for hand gesture recognition. Innov Syst Softw Eng pp 1–14
Sharma S, Kumar K (2021) Asl-3dcnn: American sign language recognition technique using 3-d convolutional neural networks. Multimed Tools Appl 80(17):26319–26331
Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556
Souly N, Spampinato C, Shah M (2017) Semi supervised semantic segmentation using generative adversarial network. In: Proceedings of the IEEE International Conference on computer vision, pp 5688–5696, 2017
Tran D, Bourdev L, Fergus R, Torresani L, Paluri M (2015) Learning spatiotemporal features with 3d convolutional networks. In: Proceedings of the IEEE International Conference on computer vision, pp 4489–4497
Vaswani A, Ramachandran P, Srinivas A, Parmar N, Hechtman B, Shlens J (2021) Scaling local self-attention for parameter efficient visual backbones. In: Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition, pp 12894–12904
Wang F, Jiang M, Qian C, Yang S, Li C, Zhang H, Wang X, Tang X (2017) Residual attention network for image classification. In: Proceedings of the IEEE Conference on computer vision and pattern recognition, pp 3156–3164
Woo S, Park J, Lee J-Y, Kweon IS (2018) Cbam: convolutional block attention module. In: Proceedings of the European Conference on computer vision (ECCV), pp 3–19, 2018
Yu C, Wang J, Peng C, Gao C, Yu G, Sang N (2018) Learning a discriminative feature network for semantic segmentation. In: Proceedings of the IEEE Conference on computer vision and pattern recognition, pp 1857–1866
Zhang X, Zhu X, Zhang N, Li P, Wang L et al (2018) Seggan: Semantic segmentation with generative adversarial network. In: 2018 IEEE Fourth International Conference on multimedia big data (BigMM), pp 1–5, 2018. IEEE
Zhou Z, Rahman Siddiquee MM, Tajbakhsh N, Liang J (2018) Unet++: a nested u-net architecture for medical image segmentation. In Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support: 4th International Workshop, DLMIA 2018, and 8th International Workshop, ML-CDS 2018, Held in Conjunction with MICCAI 2018, Granada, Spain, September 20, 2018, Proceedings 4 (pp. 3–11). Springer International Publishing.