Nose-Lip and Background Region Segmentation for Precisely Predicting Face Orientation

Shet R. Prakash * and V. N. Manju

Department of Computer Science and Engineering, CMR Institute of Technology (VTU Research Center), Bangalore, Karnataka, India
Email: shet.reshma.p@gmail.com (S.R.P.); manju.vn@cmrit.ac.in (V.N.M.)
*Corresponding author

***Abstract*—The core necessity for the video analysis-based application is the prediction of human face orientation. Here, we propose a unique and novel solution for predicting face orientation through a combination of facial features like the nose and lip region, along with non-facial features like the background region of the facial image. The proposed solution has two stages. The first stage, namely the Nose-Lip Segmenter (NLSeg), generates a nose-lip mask (m2) through a custom-trained U-Net model. The U-Net model segments the nose and lip region very accurately with an Intersection Over Union (IOU) score of 87.56%. The second stage generates a background region mask (m1) through two different methods. In the first method, a background region mask (m1) is constructed through the k-means clustering and Hue-Saturation-Value (HSV) skin color model. In the second method, the background region mask (m1) is constructed through the media-pipe selfie segmentation model. The proposed algorithm blends the nose-lip mask (m2) and background region mask (m1) together to predict face orientation across left, right, and neutral directions. The efficiency of both approaches, namely (NLSeg+method 1) and (NLSeg+method 2), is studied and evaluated on six public datasets namely BIWI-Kinect, AFLW2000-3D, Pandora, Pointing04, Stirling/ESRC 3D face, and multi camera pedestrians video dataset obtained from École Polytechnique Fédérale de Lausanne (EPFL). The proposed work yields very accurate results in spite of challenges like diverse head pose angles, facial expressions, or improper segmentation. The accuracy is 98% when subjects are non-occluded and is above 90% in the case of partial occlusion.**

***Keywords*—head pose classification, Hue-Saturation-Value (HSV) color space, *K*-means clustering, media pipe, semantic segmentation, U-Net**

I. Introduction

Video analysis is the meticulous examination of the video frames to identify and understand the activities of the objects in the video sequence. Objects like vehicles, people, and animals compose video frames. Researchers analyze people as one of the preferred subjects in the context of video analysis-based applications. Nowadays, video analysis-based applications are used in various places, like monitoring employee activity in organization, attention analysis of students in classrooms, monitoring improper behaviour of people in public places for the prevention of crime, and many more. A little research has been done in the past to count the number of individuals in the video sequence or to determine the inappropriate behaviour of the person in the video sequence, and many more. The proposed research makes use of the segmentation model for estimating the face orientation of the person, which is very beneficial for video analysis-based applications. Image segmentation is a popular computer vision technique for the pixel-wise division of the image/picture into multiple regions based on color, threshold, etc. Three different kinds of image segmentation are semantic segmentation, instance segmentation, and panoptic segmentation [1, 2]. Semantic segmentation involves assigning a group of pixels to a single class or category based on color, shape, and texture. Hence, in semantic segmentation, there can be overlap between objects belonging to the same category or class. Semantic segmentation is suitable in conditions where you have a single instance of a particular object. Some of the widely used semantic segmentation models are Fully Convolutional Networks (FCNs) [3], U-shaped Convolutional Neural Networks (U-Nets) [4], DeepLab [5], and Pyramid Scene Parsing Network (PSPNet) [6]. In instance segmentation, different instances of a particular object are assigned to a single class or category, and each instance has its own instance ID. Instance segmentation is suitable in conditions where you have multiple instances of a single object. Some of the widely used instance segmentation models are Segment Anything Model (SAM) [7], Mask Region-based Convolutional Neural Network (Mask-RCNN) [8], You Only Look Once 8 (YOLOv8) [9], and You Only Look at Coefficients (YOLACT) [10]. In panoptic segmentation, each instance of a particular object has an instance ID as well as a class ID. Panoptic segmentation is a mixture of instance and semantic segmentation. Panoptic segmentation can be used for more complex segmentation-related problems like scene understanding. One of the models for panoptic segmentation is EfficientPS [11]. The most significant contribution of the research given is as follows.

Manuscript received July 22, 2025; revised October 11, 2025; accepted October 29, 2025; published March 26, 2026.

The algorithm is based on the novel idea that when the background region, nose region, and lip region are used in conjunction with each other, they become prominent evidence for estimating the face orientation. To our knowledge, this is among the earliest studies to use facial and non-facial features in conjunction. The first milestone is to segment the nose and lip region through a U-Net segmentation model and construct a nose-lip mask (m2). The second milestone is to segment the background region and construct the background region mask (m1). Both m1 and m2 are combined to estimate if the person is looking towards the left, right, or neutral direction. The block diagram of the proposed work is given below in Fig. 1. The Nose Lip Segmenter (NLSeg) generates a nose-lip mask (m2). Method 1 generates a background region mask (m1), making use of a Hue-Saturation-Value (HSV) [12]-based skin color model and *K*-means [13] clustering technique. Hence, method1 is unsupervised and rule-based. Method2 generates a background region mask (m1), making use of the MediaPipe [14] selfie segmentation model. Hence, method2 is supervised and based on deep learning. The efficiency of both approaches, namely (NLSeg + method1) and (NLSeg + method2), is tested on various public datasets. This paper is organized as follows: Section II reviews related work conducted by former researchers. Section III outlines the methodologies and techniques used in the proposed research work. Section IV presents an overview of the public datasets employed in this study. Section V discusses the results obtained from testing the proposed solution on various datasets. Section VI describes some of the failure cases and presents the comparative analysis of two different methods used for background region mask (m1) generation. Finally, Section VII concludes with a summary of the findings, future directions, and potential applications of the research.

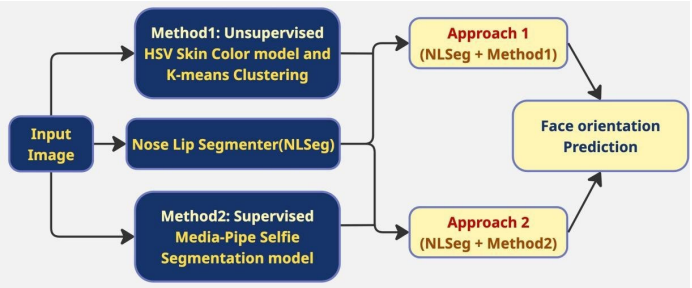


Fig. 1. Block diagram of the proposed work.

II. Related Work

The following section outlines the related work carried out by researchers in the past. Many researchers have extracted facial features for the purpose of solving challenging problems like head pose estimation or face, age, gender, and ethnicity recognition. Shah *et al*. [15] have performed a study to discover the most beneficial face landmark points for head pose estimation. Initially, the face is detected using a Deep Neural Network (DNN) [16] as well as Haar Cascade [17], and 68 facial landmarks are identified. Facial landmarks are acquired by a group of pre-trained regression trees. According to the study, it is reported that the right eye right corner point, left eye left corner point, mouth right corner point, mouth left corner point, chin, and nose tip are the foremost landmarks for estimating the orientation of the face across a wide range of pose angles. It is also reported that DNN-based face detection methods are very fast and accurate compared to Haar-based face detection.

Abate *et al*. [18] have proposed an unsupervised head pose estimation technique based on the quad tree data structure [19]. Here, the face detection is carried out through the Viola-Jones algorithm, followed by the localization of 68 facial landmarks [20]. The landmarks are represented using a quad-tree-based descriptor. The descriptors of the face derived from the quad-tree are compared against the reference pose gallery to quickly determine the Euler angles of the face. Also, the test is conducted on images of the BIWI-Kinect Head Pose Database (BIWI) and Annotated Facial Landmarks in the Wild (AFLW) datasets. Biswas *et al*. [21] have performed a study to classify the head movement of passengers and understand the relationship between the passenger and driver. It is found that it is essential to determine passengers' behavioral, physical, and psychological states for effectively designing the control structures of autonomous vehicles. The study was conducted on 56 subjects, and each subject performed the role of driver and passenger once. The head pose classification is done using the Visual Geometry Group 16 (VGG-16) Convolutional Neural Network (CNN) model with transfer learning [22].

Xia *et al*. [23] presented a technique for estimating head poses based on facial features. This study was conducted to improve the generalization ability of CNN [24–27]. First, the facial landmarks are recognized, and the facial region is delineated. The two basic components used to improve the accuracy of the estimation task are the task simplifier and the heat map generator. The task simplifier normalizes the input image to a canonical form by using an affine transformation and information about landmarks. The heatmap generator generates the heatmap, which allows the CNN to focus on the area surrounding the landmarks in the face and learn the features efficiently. The training is performed on the 300W-LP dataset, and the Euler angles are calculated [28]. Li *et al*. [29] proposed a technique for estimating head poses with a mask, called HGL, based on color texture analysis and line portrait. Here, the

processed hue channel image and the line portrait of the grayscale image are considered as input to the CNN for head pose classification. The head pose classes considered in this study are the front and side directions. The dataset available for this study is the MAFA dataset [30].

III. Proposed Work

The proposed research is an enhancement to our previous research, where only a background region mask was used for face orientation prediction without additional training [31]. In our previous research, we generated three different types of background region masks. For the given research work, we utilized only one type of background region mask (m1) along with a nose-lip region mask (m2) for accurate face pose classification. Fig. 2 depicts the series of steps followed for the research. The following steps outline the proposed work.

A. Face Detection

The process begins with face detection using a Deep Neural Network (DNN) integrated with the Single-Shot Detector (SSD) framework [32]. The framework has a pre-trained ResNet-10 as a backbone network [33]. SSD is very efficient since it has the capability to identify certain objects present in the image with a single pass through the network.

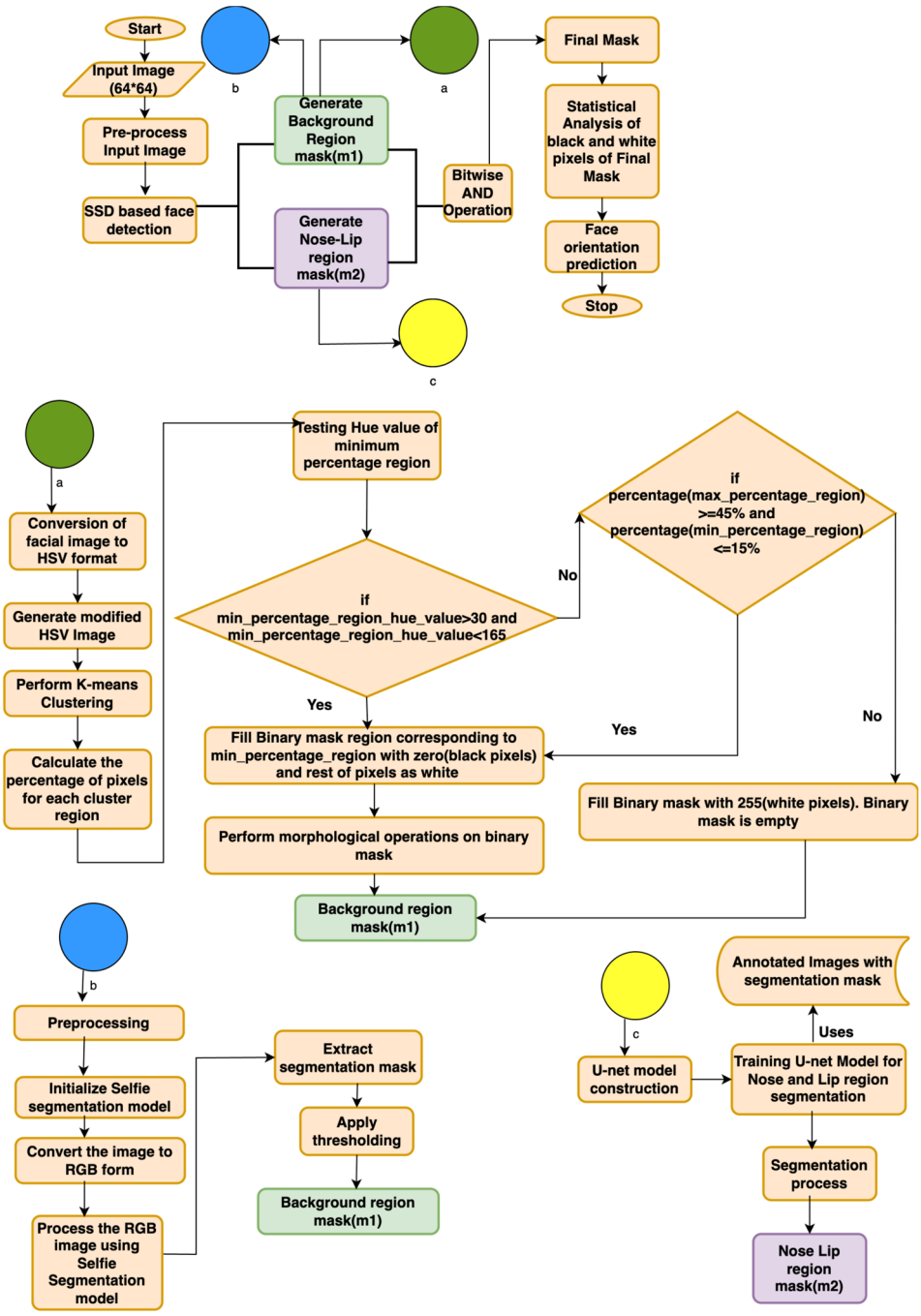


Fig. 2. Flowchart of proposed research work.

B. Construction of Background Region Mask (M1)

1) Method 1

We derive the background region mask (m1) by leveraging the HSV color model and the *K*-means clustering algorithm. *K*-means clustering is a well-liked unsupervised clustering algorithm. It splits the image into *K* regions based on color similarity, texture, and intensity of pixels.

(1) The identified face is resized and then provided as input. The resolution of the input image is 64×64 pixels.
(2) The value of *K* is chosen as 3 to enable segmentation of the face image into hair, background, and skin.
(3) The centers of the clusters are assigned randomly at the start.
(4) Next, we set the termination criteria. The segmentation process ends when either the epsilon value (i.e., 0.2) or the maximum iteration (i.e., 50) is reached.
(5) Each pixel is assigned to one of the *K* clusters depending on the nearest center.
(6) Next, the cluster centers for each cluster are revised, and the pixels are reassigned to the new cluster centers.
(7) Steps (5) and (6) are repeated according to the specified termination criteria.

The segmentation results are displayed in the figure given below. In Fig. 3(a), hair, skin, and background regions are clearly segmented. In certain images, the background is not segmented independently and is instead grouped with the skin region, as illustrated in Fig. 3(b). This happens when the background region has the same color as the skin color or if the image has inferior quality. Hence, *K*-means segmentation performs poorly when there are extreme illumination variations, occlusions, and extreme head poses. The background region mask is constructed very well in the case of correctly segmented images, as displayed in Fig. 3(a). In the case of a poorly segmented image, however, the mask for the background region is empty, as can be seen in Fig. 3(b).

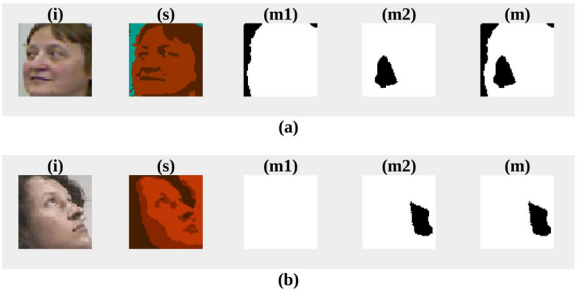


Fig. 3. Step by step segmentation process. (a) Original image, segmented image, background region mask (m1), nose-lip region mask (m2), final mask (m) of person 1. (b) Original image, segmented image, background region mask (m1), nose-lip region mask (m2), final mask (m) of person 2.

Although the background region mask is empty in some situations, the proposed work performs face pose classification efficiently. This is because face orientation is determined using both the background region and facial features like the nose and lips. The nose-lip region mask (m2) is produced by the process of semantic segmentation through a U-Net model, which is described in the next section. The HSV color space describes the color with respect to human eye perception. It has three vital components, specifically hue, saturation, and value. The hue component captures color identity, saturation expresses the color's purity or gray content, and value represents its brightness. By specifying particular values for hue, saturation, and value components, it is possible to perform color-based image segmentation. Here, we obtain the background region mask (m1) with the presumption that the background region is composed of non-skin-like pixels, and it also occupies the minimum area of the facial image. As shown in Fig. 2, initially, the input image is transformed into HSV format. The modified saturation channel is combined with the input image and processed using the *K*-means clustering algorithm to obtain the segmented image. This segmented image is then converted back to HSV format, where the hue values of the three regions are analyzed. The region with a hue value between 31 and 164 corresponds to a non-skin region, and a region with a hue value between 0 and 30 or 165 and 179 corresponds to a skin region. We came across the following cases during the generation of background region masks.

Case 1: Non-skin pixels in the background region, and the image is segmented properly. For example,

Region 1: HSV value = (15, 255, 38), percentage = 55.39%;

Region 2: HSV value = (0, 229, 106), percentage = 32.03%;

Region 3: HSV value = (108, 243, 177), percentage = 12.57%.

Here, region 3 has a huge value of 108, which means it is a non-skin region. Thus, region 3 is identified as the background region used in the creation of the background region mask.

Case 2: Skin-like pixels in the background region, and the image is segmented properly. For example,

Region 1: HSV value = (29, 255, 172), percentage = 9.32%;

Region 2: HSV value = (10, 255, 126), percentage = 48.80%;

Region 3: HSV value = (12, 248, 68), percentage = 41.87%.

Each of the three regions has hue values between 0 and 30, implying the presence of skin-like pixels in all of them. Accordingly, region 1, having the lowest percentage, is identified as the background region for generating the background mask. In this study, we used a threshold value of 15 for the region with the lowest percentage and 45 for the region with the highest percentage. The threshold was chosen after testing the approach on multiple face images.

Case 3: Improper segmentation. For example,

Region 1: HSV value = (14, 246, 57), percentage = 39.57%;

Region 2: HSV value = (10, 253, 106), percentage = 28.22%;

Region 3: HSV value = (9, 253, 146), percentage = 32.20%.

Each of the three regions has hue values between 0 and 30, implying the presence of skin-like pixels in all of them. With a percentage of 28.22%, region 2 is the lowest among the regions, but it is not identified as a background region since it exceeds the 15% threshold. Hence, the background region mask is unfilled or empty in this case.

2) Method 2

The media-pipe selfie segmentation model is used to generate the background region mask (m1). This segmentation model isolates the foreground region from the background, functioning as a one-class segmentation approach. It is an optimized model and runs very efficiently on CPU or GPU devices. The backbone network used in media-pipe selfie segmentation is the DeepLabV3+ architecture. DeepLabV3+ employs dilated convolution to effectively reduce computational overhead while enhancing the amount of information captured at each convolutional stage. To improve the segmentation results, multiple parallel dilated convolution layers with different dilation rates are used. It is a very fast and lightweight model for extracting the background region in real time. The procedure for generating mask m1 through media pipe models is illustrated in Fig. 2. The mask m1 is created through the following steps:

(1) Read the input image. The input image size is 64 pixels in width and 64 pixels in height.

(2) Import the required libraries. Initialize the selfie segmentation model.

(3) Transform the input image from BGR to RGB format.

(4) Process the RGB image with the selfie segmentation model to create a segmentation mask.

(5) Finally, we apply a threshold operation to the segmentation mask to create the mask for the background region (m1).

In Fig. 4, it is evident that the background region mask (m1) is precisely created for persons 1, 2, and 3. The media-pipe selfie segmentation provides clear results in different lighting environments. But extremely complicated background environments or very low lighting conditions can impact the performance of segmentation results.

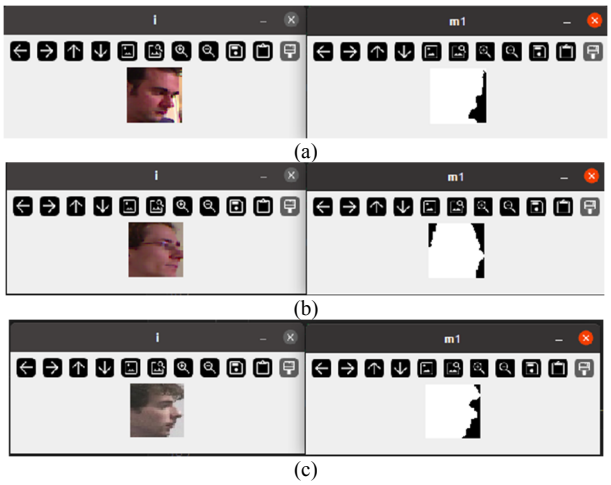


Fig. 4. Segmentation process using media-pipe selfie segmentation. (a) Background region mask (m1) of person 1. (b) Background region mask (m1) of person 2. (c) Background region mask (m1) of person 3.

C. Construction of Nose-Lip Region Mask (M2)

In the proposed research, a U-Net model-based Nose-Lip Segmenter (NLSeg) is utilized for performing nose and lip region semantic segmentation. The biomedical image segmentation problem was the motivating factor for the emergence of the U-Net. As presented in Fig. 5, it consists of a left-side contracting path and a right-side expansive path.

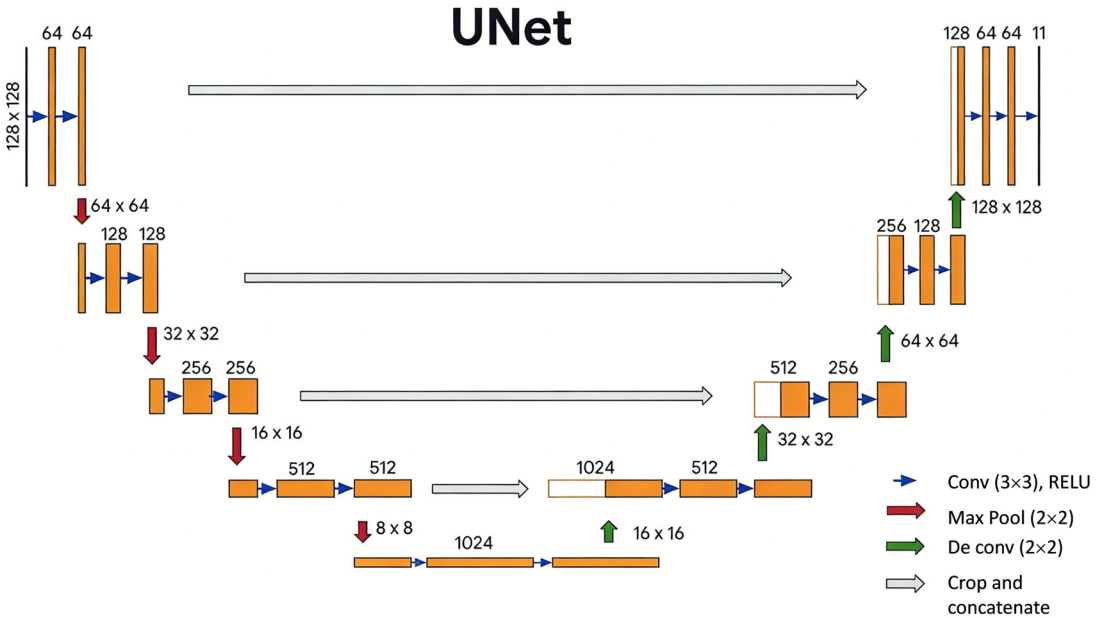


Fig. 5. U-Net model architecture.

The U-Net model performs very well even if the quantity of training data is less. The contracting path is like the traditional CNN. In each step of the contraction path, the following operations are applied sequentially, i.e., double convolution with a 3×3 kernel (unpadded), Rectified Linear Unit (ReLU), and a 2×2 max pooling operation with a step size of 2. At each step of the contracting path, the image size decreases, but the number of feature channels doubles. Hence, in the left-side path network, it learns and obtains rich information about the features. Along each step of the expansive path, the following operations are applied in sequence, i.e., image size is increased, and the number of feature channels is halved. In each step, the feature channel from the contracting path is concatenated, and a double convolution with a 3×3 kernel (unpadded) is applied, followed by ReLU. Hence, in an expansive path, the network learns about the spatial information of the features. In the last layer, a 1×1 convolution is applied for aligning the feature vector according to the total number of classes. Hence, the convolution operation is applied 23 times in the whole network.

Unlike the original work, which used unpadded convolution operations, this one uses padded convolution operations. Padding is used to ensure that the output image size matches the 64×64 input image size. The batch normalization procedure is used prior to max pooling and ReLU [34]. This process offers several benefits. It helps to boost the generalization capability of the model and avoid over-fitting problems. Batch normalization improves model performance by enabling the use of higher learning rates, which in turn reduces training time. First, the nose and lip regions of the human faces from the training dataset are annotated using the VGG annotation tool [35]. This tool is accessible at URL https://www.robots.ox.ac.uk/~vgg/software/via/.

The training and validation images are annotated as shown in Fig. 6(a). The nose and lip region are marked using the polygon tool, and a corresponding binary segmentation mask is created as shown in Figs. 6(b) and 6(c). Here, training and validation data are composed of images obtained from the BIWI, Pointing-04, and Pandora datasets. The training data contains 1383 training images along with their associated segmentation masks. Similarly, the validation data contains 153 validation images along with their associated segmentation masks. U-Net is employed in this work because it delivers precise pixel-level masks, which are essential for accurately identifying the nose-lip region of the face. Its ability to handle complex shapes makes it well-suited for segmenting intricate facial features. Furthermore, the encoder-decoder architecture with skip connections preserves fine details, enabling high-accuracy segmentation. Hence, U-Net is preferred over models like YOLO, Faster R-CNN, Mask R-CNN, SSD, and FCN-based architectures. While detection models are faster for locating objects, they cannot provide the detailed masks necessary for downstream analysis.

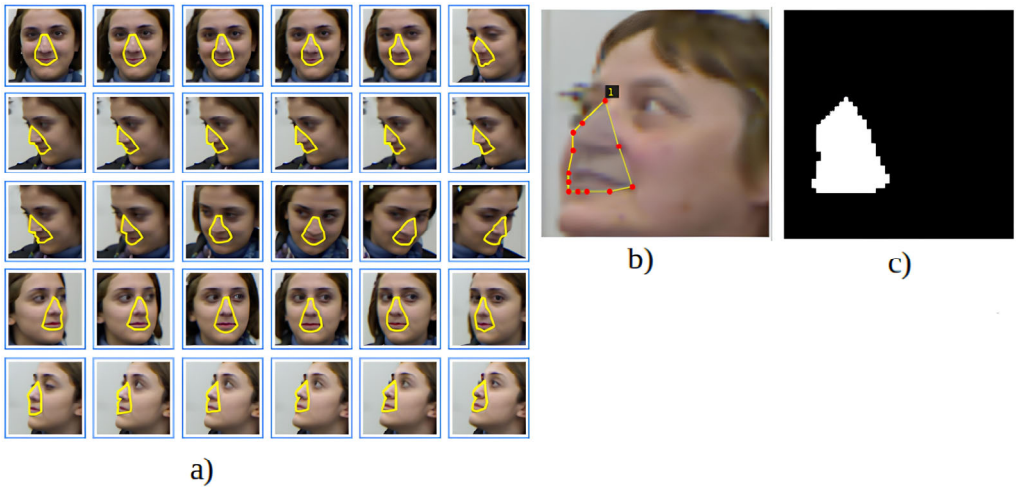


Fig. 6. The training and validation images. (a) Annotation process, (b) Nose-lip region marking, (c) Binary segmentation mask.

Each step in the U-Net model summary is highlighted with a different color in Fig. 7, which is seen below. The original 64×64 input image size is halved at each stage of the contracting path. The image size becomes 4×4 in the final layer of the contracting path. Along the expansive path, the image size is doubled at each step. Fig. 8 shows the Intersection Over Union (IOU) score, loss, and accuracy for the training and validation process.

The IOU is the salient metric used for assessing the performance of the object detection model or segmentation models. The IOU, in the case of image segmentation, is a measure of overlay between the predicted segmentation mask and the ground truth segmentation mask. A large overlap indicates better accuracy. The formula for IOU can be expressed as Eq. (1):

$$IOU = \frac{Area\ of\ Intersection}{Area\ of\ Union} \tag{1}$$

Performance is evaluated using various settings, as shown in Table I. With a fixed learning rate of 0.005, the batch size and the number of epochs are changed in each case. With an IOU score of 87.56% and a batch size of 8

and 50 epochs, case 5 yields the maximum accuracy of 96.40%.

As a result, numerous researchers have previously employed the U-Net model, which is highly popular. The numerous U-Net applications found in the literature are included in Table II, along with the accompanying mIOU scores.

Layer (type)	Output Shape	Param #
Conv2d-1	[-1, 64, 64, 64]	1,792
BatchNorm2d-2	[-1, 64, 64, 64]	128
ReLU-3	[-1, 64, 64, 64]	0
Conv2d-4	[-1, 64, 64, 64]	36,928
BatchNorm2d-5	[-1, 64, 64, 64]	128
ReLU-6	[-1, 64, 64, 64]	0
MaxPool2d-7	[-1, 64, 32, 32]	0
Conv2d-8	[-1, 128, 32, 32]	73,856
BatchNorm2d-9	[-1, 128, 32, 32]	256
ReLU-10	[-1, 128, 32, 32]	0
Conv2d-11	[-1, 128, 32, 32]	147,584
BatchNorm2d-12	[-1, 128, 32, 32]	256
ReLU-13	[-1, 128, 32, 32]	0
MaxPool2d-14	[-1, 128, 16, 16]	0
Conv2d-15	[-1, 256, 16, 16]	295,168
BatchNorm2d-16	[-1, 256, 16, 16]	512
ReLU-17	[-1, 256, 16, 16]	0
Conv2d-18	[-1, 256, 16, 16]	590,080
BatchNorm2d-19	[-1, 256, 16, 16]	512
ReLU-20	[-1, 256, 16, 16]	0
MaxPool2d-21	[-1, 256, 8, 8]	0
Conv2d-22	[-1, 512, 8, 8]	1,180,160
BatchNorm2d-23	[-1, 512, 8, 8]	1,024
ReLU-24	[-1, 512, 8, 8]	0
Conv2d-25	[-1, 512, 8, 8]	2,359,808
BatchNorm2d-26	[-1, 512, 8, 8]	1,024
ReLU-27	[-1, 512, 8, 8]	0
MaxPool2d-28	[-1, 512, 4, 4]	0
Conv2d-29	[-1, 1024, 4, 4]	4,719,616
BatchNorm2d-30	[-1, 1024, 4, 4]	2,048
ReLU-31	[-1, 1024, 4, 4]	0
Conv2d-32	[-1, 1024, 4, 4]	9,438,208
BatchNorm2d-33	[-1, 1024, 4, 4]	2,048
ReLU-34	[-1, 1024, 4, 4]	0
ConvTranspose2d-35	[-1, 512, 8, 8]	2,097,664
Conv2d-36	[-1, 512, 8, 8]	4,719,104
BatchNorm2d-37	[-1, 512, 8, 8]	1,024
ReLU-38	[-1, 512, 8, 8]	0
Conv2d-39	[-1, 512, 8, 8]	2,359,808
BatchNorm2d-40	[-1, 512, 8, 8]	1,024
ReLU-41	[-1, 512, 8, 8]	0
ConvTranspose2d-42	[-1, 256, 16, 16]	524,544
Conv2d-43	[-1, 256, 16, 16]	1,179,904
BatchNorm2d-44	[-1, 256, 16, 16]	512
ReLU-45	[-1, 256, 16, 16]	0
Conv2d-46	[-1, 256, 16, 16]	590,080
BatchNorm2d-47	[-1, 256, 16, 16]	512
ReLU-48	[-1, 256, 16, 16]	0
ConvTranspose2d-49	[-1, 128, 32, 32]	131,200
Conv2d-50	[-1, 128, 32, 32]	295,040
BatchNorm2d-51	[-1, 128, 32, 32]	256
ReLU-52	[-1, 128, 32, 32]	0
Conv2d-53	[-1, 128, 32, 32]	147,584
BatchNorm2d-54	[-1, 128, 32, 32]	256
ReLU-55	[-1, 128, 32, 32]	0
ConvTranspose2d-56	[-1, 64, 64, 64]	32,832
Conv2d-57	[-1, 64, 64, 64]	73,792
BatchNorm2d-58	[-1, 64, 64, 64]	128
ReLU-59	[-1, 64, 64, 64]	0
Conv2d-60	[-1, 64, 64, 64]	36,928
BatchNorm2d-61	[-1, 64, 64, 64]	128
ReLU-62	[-1, 64, 64, 64]	0
Conv2d-63	[-1, 2, 64, 64]	130

Total params: 31,043,586
Trainable params: 31,043,586
Non-trainable params: 0

Fig. 7. Model summary.

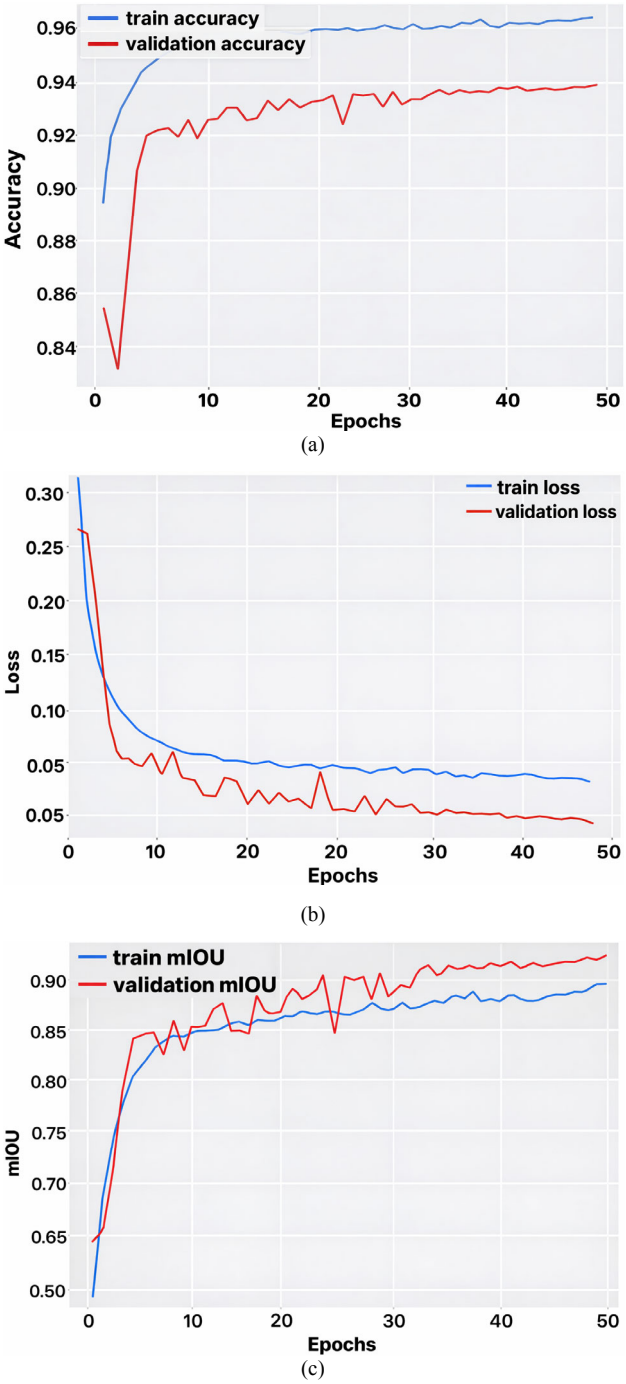


Fig. 8. Training and validation metrics. (a) Accuracy; (b) Loss; (c) mIOU.

TABLE I. PERFORMANCE ASSESSMENT USING DIFFERENT PARAMETERS

Case	Batch Size	Learning Rate	Epochs	Training Accuracy	Validation Accuracy	Best Validation IOU
1	8	0.005	30	96.06%	93.30%	85.58%
2	16	0.005	30	95.37%	88.70%	80.74%
3	32	0.005	30	93.84%	81.15%	71.97%
4	64	0.005	30	90.68%	72.09%	60.00%
5	8	0.005	50	96.40%	93.80%	87.56%
6	16	0.005	100	95.87%	89.54%	83.77%

TABLE II. FORMER APPLICATIONS OF U-NET WITH MIOU SCORE

Reference	Year	Segmentation Details	mIOU Score
[36]	2020	In this case, U-Net-based segmentation is used to facilitate better urban management and planning. Ten distinct urban villages' worth of satellite photos are used for the segmentation process. Background, old home, old factory, iron roof building, and new construction are among the classes taken into consideration for the segmentation. This study shows the power of U-Net in delineating the individual buildings in the high-density urban village.	90%
[37]	2020	Here, the gangue and raw coal are separated using U-Net-based segmentation. Furthermore, 54 shortwave infrared (SWIR) raw coal images gathered from Datong Coalfield are used in the training phase. Gangue must be appropriately separated to lower transportation costs and pollution. The outcomes show that the U-Net model is very suitable for coal image segmentation across the industry.	86%
[38]	2024	To detect the thermal hotspot and facilitate effective solar panel installation and management, U-Net-based segmentation is used in this instance. Various classes considered for the segmentation are solar panels, thermal hotspots, and background components. The basic goal of the study is to detect the faults, like thermal hotspots, and increase the operational lifespan of solar panels.	72.46%
[39]	2024	Here, satellite images are used for aircraft identification and recognition using U-Net-based segmentation. The analysis and performance evaluation of the suggested model are carried out using the Dense Labeling Remote Sensing Dataset (DLRSD).	95.08%
Ours	2025	Proposed Method	87.56%

D. Face Pose Classification

The left, right, and neutral directions are the three classes taken into account while classifying face poses. A bit-wise and operation is applied on both the binary masks, i.e., the background region mask (m1) and the nose-lip region mask (m2), to generate the final segmentation mask (m). The percentage of black pixels is analyzed in the leftward and rightward parts of the final segmentation mask (m). Even the difference between the percentage of pixels in the leftward and rightward directions is computed. If the percentage of black pixels in the left part is greater than in the right part and the difference is more than 5%, the prediction is left. If the percentage of black pixels in the right part is greater than in the left part and the difference is greater than 5%, the prediction is right. If the above two conditions are not met, the prediction is neutral. The algorithm for the proposed work is shown in Table III.

TABLE III. ALGORITHM OF THE PROPOSED RESEARCH WORK

Algorithm 1: Proposed Face Orientation Classification System

Input: RGBImage
Output: PredictedOrientation ∈ {Left, Right, Neutral}
FaceImage ← FaceDetection(RGBImage)
Resized_FaceImage ← Resize(FaceImage)
Model ← UNet(num_classes)
Face_Tensor ← CreateTensor(Resized_FaceImage)
Output_Tensor ← GenerateLabel(Face_Tensor, Model)
Segmented_Image1 ← CreateSegmentedImage(Output_Tensor)
NoseLipMask ← CreateMask(Segmented_Image1)
choice ← SelectApproach() ▷ User chooses Approach-1 or Approach-2
if choice = 1 then
    BackgroundRegionMask ← Generate_BackgroundMask1(Resized_FaceImage)
else if choice = 2 then
    BackgroundRegionMask ← Generate_BackgroundMask2(RGBImage)
else
    print("Invalid Choice")
end if
BackgroundMask ← ApplyMorphologicalOperation(BackgroundRegionMask)
FinalMask ← BitwiseAND(NoseLipMask, BackgroundMask)
(LeftPart, RightPart) ← Divide(FinalMask)
P_left ← CountBlackPixels(LeftPart) / ((ImageWidth × ImageHeight) / 2) × 100
P_right ← CountBlackPixels(RightPart) / ((ImageWidth × ImageHeight) / 2) × 100
diff ← |P_left − P_right|
if P_left > P_right and diff > 5 then
    PredictedOrientation ← "Left"
else if P_right > P_left and diff > 5 then
    PredictedOrientation ← "Right"
else
    PredictedOrientation ← "Neutral"
end if
return PredictedOrientation

Algorithm 2: Background Region Mask Generation (Approach 1)

Input: Resized_FaceImage
Output: BackgroundRegionMask
procedure Generate_BackgroundMask1(Resized_FaceImage)
    HSV_Image ← ConvertToHSV(Resized_FaceImage)
    (*H*, *S*, *V*) ← SplitChannels(HSV_Image)

```
    for x ← 0 to n do
      for y ← 0 to m do
        S[x, y] ← 255
      end for
    end for
    Modified_HSV ← MergeChannels(H, S, V)
    RGB_Image ← ConvertToRGB(Modified_HSV)
    Segmented_Image ← KMeans(RGB_Image)
    HSV_Segmented ← ConvertToHSV(Segmented_Image)
    RegionStats ← CreateDictionary(HSV_Segmented)
    if 30 < RegionStats.min_percentage_region_pixelval < 165 then
      Case ← 1
    else if RegionStats.min_percentage_region_pixelval ≤ 30
        or RegionStats.min_percentage_region_pixelval ≥ 165 then
      Case ← 2
    end if
    for r ← 0 to n do
      for c ← 0 to m do
        if Case = 1 then
          if HSV_Segmented[r, c] = RegionStats.min_percentage_region_pixelval then
            BackgroundRegionMask[r, c] ← 0
          end if
        else if Case = 2 and
            RegionStats.max_percentage_region_percentage ≥ 45 and
            RegionStats.min_percentage_region_percentage ≤ 15 then
          if HSV_Segmented[r, c] = RegionStats.min_percentage_region_pixelval then
            BackgroundRegionMask[r, c] ← 0
          end if
        end if
      end for
    end for
    return BackgroundRegionMask
end procedure
```

Algorithm 3: Background Region Mask Generation (Approach 2)

```
Input: RGBImage
Output: BackgroundMask
procedure Generate_BackgroundMask2(RGBImage)
    selfie_seg ← mp.solutions.selfie_segmentation
    ss ← selfie_seg.SelfieSegmentation(model_selection)
    results ← ss.process(RGBImage)
    seg_mask ← results.segmentation_mask
    binary_mask ← Threshold(seg_mask)
    binary_mask_int ← ConvertToUint8(binary_mask)
    binary_mask_3channel ← ConvertTo3Channel(binary_mask_int)
    return binary_mask_3channel
end procedure
```

IV. Datasets Used

Here, we have used images from the BIWI-Kinect [40], Pointing04 [41], and Pandora [42] datasets for training. The images of AFLW2000-3D [43], BIWI-Kinect, Pointing04, Pandora, and Stirling/ESRC [44] 3D face datasets are used in the testing phase.

A. BIWI-Kinect

Many researchers use the well-known BIWI-Kinect head pose dataset to solve the head pose estimation problem. This dataset contains more than 15,000 images of 20 people with diverse head pose angles. A Kinect sensor that was positioned around a meter away from the 20 participants was used to collect the data. Each image has a _pose.txt file that contains the ground truth value for the head rotation, which is represented by a 3×3 rotation matrix, and the head positioning in 3D.

B. Pointing04

The Pointing04 dataset contains 2790 images of 15 people with extensive head pose angles. The pan and tilt angle varies between −90 and +90 degrees. The age range of people varies between 20 and 40 years. Among 15 people, seven people wear glasses and five people have facial hair. The ground truth is in the form of the tilt and pan angle for each image.

C. Pandora

The Pandora dataset is a large dataset and is often used by researchers to estimate head and shoulder pose. It includes images of 20 subjects who performed wide head movements ranging from +70/−70 degree roll, +100/−100 degree pitch, and +125/−125 degree yaw. Also, the subjects wear glasses, sunglasses, caps, etc. Ground truth information is available for head pose as well as shoulder pose.

D. AFLW2000-3D

The AFLW2000-3D dataset is chiefly used for assessing the performance or behavior of facial landmark detection models. This dataset consists of around 2000 images with ground truth information on facial landmarks.

E. Stirling/ESRC 3D Face

The main goal of creating this dataset was to address problems related to face perception and recognition. 45 men and 54 women participated in the data collection phase. Expressions like anger, fear, surprise, unhappiness, disgust, happiness, and neutrality are also considered when recording the data. The data is recorded in indoor as well as outdoor environments. The images are captured in the presence of an array of four cameras set at 0, 22.5, 45, and 90 degrees to the main source of lighting. Additional dataset details are available at this URL: https://pics.stir.ac.uk/ESRC/index.htm.

F. "EPFL" Data Set: Multi-camera Pedestrian Videos

Multi-camera pedestrian dataset was collected by the Computer Vision Laboratory (CVLab) at EPFL and contains several video sequences captured simultaneously from multiple static cameras observing the same scene from different viewpoints [45]. The dataset contains videoes on indoor and outdoor scenarios, such as laboratory, campus, terrace, passageway, and basketball sequences. The videos are recorded at 25 frames per second and feature 3–4 camera views per scene.

V. Experimental Results

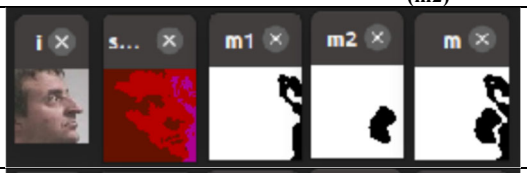
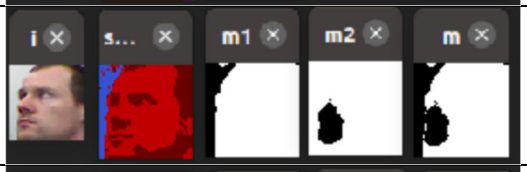
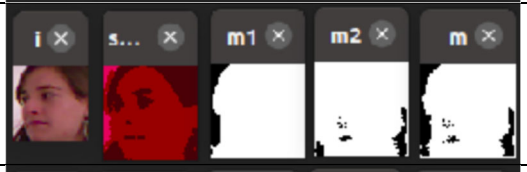
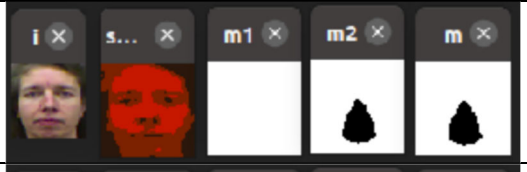
This section illustrates the results obtained after testing the proposed work against different benchmark datasets and different test conditions. The experiment is conducted using the images of the BIWI-Kinect, Pointing04, AFLW2000-3D, Pandora, Stirling/ESRC 3D face datasets, and EPFL Multi-camera pedestrian's dataset. The testing and training images are completely different. Both the qualitative and quantitative results obtained are presented below.

A. Qualitative Results

1) Test scenario 1

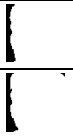
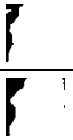
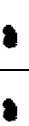
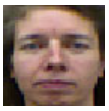
In this scenario, the experiment is conducted on some images collected randomly from the BIWI-Kinect and Pointing04 datasets. Here, the performance of the first approach (NLSeg+method 1) is evaluated and studied. Table IV shows the results obtained for some images.

TABLE IV. Results Obtained on Images of BIWI-Kinect, Pointing04 Dataset Using First Approach NLSeg+Method1

No.	Original Image (i)	Segmented Image (s)	Background Region Mask (m1)	Nose-Lip Region Mask (m2)	Final Mask (m)	Segmentation Status	Percentage of Black Pixels in Left Region	Percentage of Black Pixels in Right Region	Prediction
1						Background region and nose-lip region is segmented.	32.66%	0	Left
2						Background region and nose-lip region is segmented.	0.14%	46.09%	Right
3						Background region and nose-lip region is segmented.	8.10%	18.45%	Right
4						Background region and nose-lip region is segmented.	34.52%	0	Left
5						Only nose-lip region is segmented.	12.74%	8.25%	Neutral
6						Only nose-lip region is segmented.	18.99%	0	Left

As shown in Table IV, in images 1, 2, 3, and 4, the background region mask (m1) is non-empty, and even the nose-lip region is segmented very well. But in images 5 and 6, mask m1 is empty because the segmentation is improper. Yet the proposed work classifies the face direction accurately for images 5 and 6 using the nose-lip region mask (m2). Image 5 is predicted to have a neutral label because even though the percentage of black pixels in the left region is higher than in the right region, the difference in percentage is less than the threshold value of 5. The prediction is left if the percentage of black pixels in the left part of the final mask is greater than in the right part of the final mask and vice versa.

TABLE V. RESULTS OBTAINED ON IMAGES OF BIWI-KINECT, POINTING04 DATASET WITH BOTH APPROACHES (NLSEG+METHOD1 AND NLSEG+METHOD2)

No.	Dataset	Subject No	Approach	Original Image	Background Region Mask (m1)	Nose Lip Region Mask (m2)	Final Mask(m) m=m1+m2	Final Prediction	Direction
1	BIWI	1	1						Right
			2						Right
2	Pointing04	6	1						Right
			2						Right
3	Pointing04	10	1						Left
			2						Left
4	BIWI	5	1						Neutral
			2						Neutral
5	BIWI	19	1						Left
			2						Left
6	BIWI	8	1						Left
			2						Left
7	Pointing04	8	1						Right
			2						Right
8	BIWI	20	1						Left
			2						Left
9	Pointing04	14	1						Right
			2						Right
10	Pointing04	1	1						Right
			2						Right

2) Test scenario 2

Here some images randomly selected from the BIWI-Kinect and Pointing04 datasets are used for the experiment. We have tested the performance of both approaches, i.e., (NLSeg+method 1) and (NLSeg+method 2). Regarding aspects like accuracy and computing efficiency, it is found that the second approach (NLseg+method 2) performs better than the first approach (NLseg+method 1). The outcomes of both approaches are displayed in Table V.

As shown in the table, the background region mask (m1) obtained through the second approach more clearly represents the background information compared to the first approach for all the images. Only for images 4 and 5, mask m1 generated through approach 1 is empty, and the mask m1 generated through approach 2 is non-empty. The predictions for all the images are the same if approaches 1 and 2 are considered. It was observed that the second approach is computationally faster compared to the first approach.

3) Test scenario 3

The effectiveness of the proposed research work is tested against the images of the AFLW2000-3D dataset. The outcomes are displayed in Figs. 9(a) and 10(b). Here, the images have a broad range of changes across factors like illumination, pose, and expression. The proposed research can efficiently classify the face pose in spite of these challenges.

(a)

(b)

Fig. 9. Results obtained on AFLW2000-3D dataset with predicted left, right and neutral labels. (a) Results obtained with approach 1 (NLSeg+method 1). (b) Results obtained with approach 2 (NLSeg+method 2).

4) Test scenario 4

In this case, images from the Pandora dataset are used for the experiment. These images have been organized into various folders and carefully tagged as neutral, left, and right. It included 1200 left-labeled images, 1200 right-labeled images, and 100 neutral-labeled images. Hence, the classes are imbalanced. There are fewer samples belonging to the neutral class. The confusion matrix is displayed in Fig. 10. There is a 2% misclassification rate and 98% accuracy for both approaches.

Training Set

TARGET / OUTPUT	Left	Right	Neutral	SUM
Left	1194 47.76%	17 0.68%	7 0.28%	1218 98.03% 1.97%
Right	1 0.04%	1173 46.92%	8 0.32%	1182 99.24% 0.76%
Neutral	5 0.20%	10 0.40%	85 3.40%	100 85.00% 15.00%
SUM	1200 99.50% 0.50%	1200 97.75% 2.25%	100 85.00% 15.00%	2452 / 2500 98.08% 1.92%

(a)

Training Set

TARGET / OUTPUT	Left	Right	Neutral	SUM
Left	1187 47.48%	6 0.24%	6 0.24%	1199 99.00% 1.00%
Right	1 0.04%	1171 46.84%	7 0.28%	1179 99.32% 0.68%
Neutral	12 0.48%	23 0.92%	87 3.48%	122 71.31% 28.69%
SUM	1200 98.92% 1.08%	1200 97.58% 2.42%	100 87.00% 13.00%	2445 / 2500 97.80% 2.20%

(b)

Fig. 10. Confusion matrix for images of Pandora dataset. (a) Matrix obtained with approach 1 (NLSeg+method 1). (b) Matrix obtained with approach 2 (NLSeg+method 2).

Each class's precision and recall values are determined, as shown in Table VI. The Macro-F1-Score and Weighted-F1-Score of approach 1 are 0.94 and 0.98, and for approach 2, they are 0.92 and 0.98. Precision, in the case of multi-class classification, is the quantification of a model's capability to correctly classify class labels, for instance. Recall for multi-class classification refers to the percentage of correctly classified instances in relation to the total number of instances in that particular class. The macro-F1-Score is the average of F1-Scores procured across all the classes. The weighted F1 value is the average of the F1 values of all classes, taking into account the weight/support of the individual classes.

The formulas for accuracy, precision, F1-Score, and recall are given below in Eqs. (2)–(5).

$$Accuracy = \frac{Correct\ Predictions}{All\ Prediction} \quad (2)$$

$$Precision_{Left} = \frac{TP_{Left}}{TP_{Left} + FP_{Left}} \quad (3)$$

$$Recall_{Left} = \frac{TP_{Left}}{TP_{Left} + FN_{Left}} \quad (4)$$

$$F1 - Score_{Left} = \frac{2 \times Precision_{Left} \times Recall_{Left}}{Precision_{Left} + Recall_{Left}} \quad (5)$$

TABLE VI. PRECISION, RECALL AND F1-SCORE OF ALL THREE CLASSES WHEN CLASSES ARE IMBALANCED

Class name	No. of samples	Approach	Precision	Recall	F1-Score
Left	1200	NLSeg+method 1	0.98	0.99	0.99
	1200	NLSeg+method 2	0.99	0.99	0.99
Right	1200	NLSeg+method 1	0.99	0.98	0.98
	1200	NLSeg+method 2	0.99	0.98	0.98
Neutral	100	NLSeg+method 1	0.85	0.85	0.85
	100	NLSeg+method 2	0.71	0.87	0.78

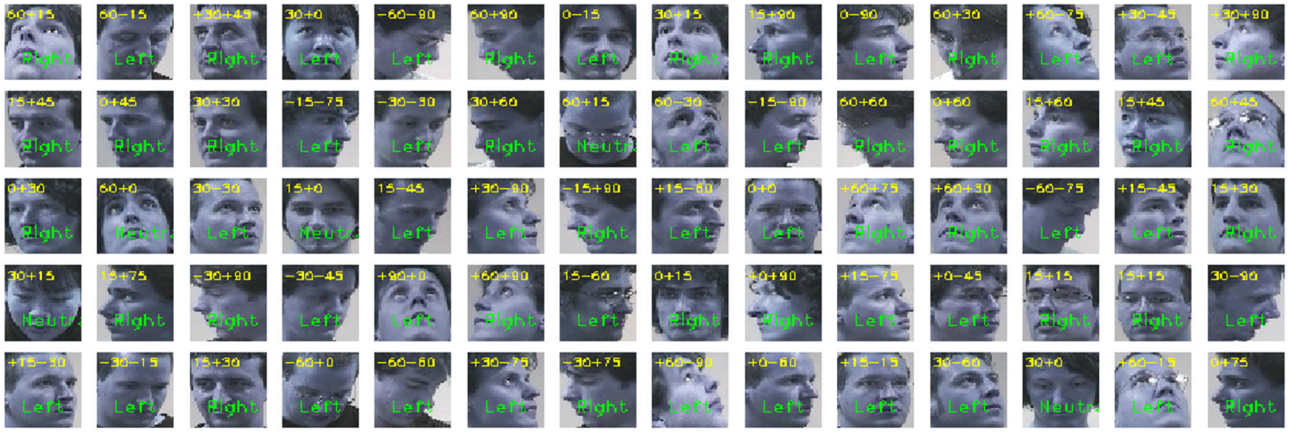
5) Test scenario 5

The Pandora dataset contains the data of 20 subjects who have worn different garments or accessories during data collection, like caps, sunglasses, prescription glasses, hoodies, scarves, etc. Here, we tested images in which the face was covered by several items or garments. The accuracy attained in each situation is shown in Table VII below. In this case, the test is conducted on 20 images selected manually for each subject. When subjects wear accessories like hoodies and caps, the accuracy is 100%. This is because hoodies and hats do not cover the nose and lip area. The accuracy is slightly reduced when subjects are wearing the mask, sunglasses, or prescription glasses because either the lip or nose region is hidden or occluded to some extent.

TABLE VII. CLASSIFICATION ACCURACY WHEN SUBJECTS HAVE WORN DIFFERENT GARMENTS OR ITEM

Subject	Item worn	No of images considered	Approach	No of correct prediction	No of incorrect prediction	Accuracy
1	cap	20	NLSeg+method 1	20	0	100%
			NLSeg+method 2	20	0	100%
2	sunglasses1	20	NLSeg+method 1	20	0	100%
			NLSeg+method 2	19	1	95%
3	hoodies	20	NLSeg+method 1	20	0	100%
			NLSeg+method 2	20	0	100%
4	sunglasses2	20	NLSeg+method 1	19	1	95%
			NLSeg+method 2	20	0	100%
5	mask and cap	20	NLSeg+method 1	18	2	90%
			NLSeg+method 2	16	4	90%
6	mask	20	NLSeg+method 1	18	2	90%
			NLSeg+method 2	19	1	95%
7	prescription glasses	20	NLSeg+method 1	20	0	100%
			NLSeg+method 2	19	1	95%

(a)



(b)

Fig. 11. Outcome on images of Pointing04 datasets. (a) Results obtained with approach 1 (NLSeg+method 1). (b) Results obtained with approach 2 (NLSeg+method 2).

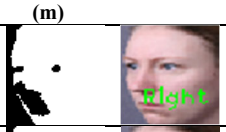
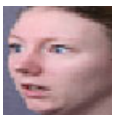
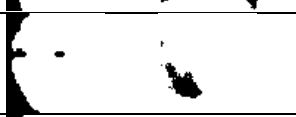
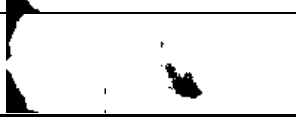
6) *Test scenario 6*

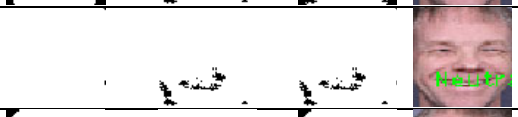
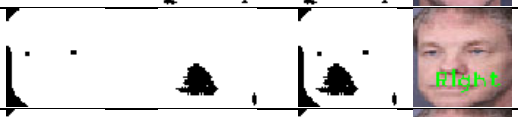
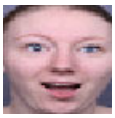
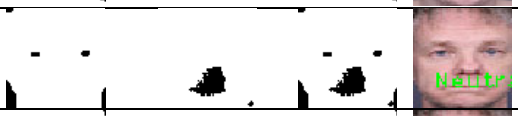
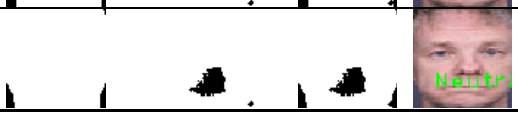
In this scenario, the images extracted from the Pointing04 dataset are utilized for the purpose of testing. Fig. 11 shows the results obtained for both approaches. A total of 65 images belonging to different participants are considered for the study. For each of the 65 images, the predicted label is displayed together with the tilt and pan angle of the ground truth. The yellow label indicates the tilt and pan angle values. The green label indicates the predicted direction for head pose (left, right, or neutral). The head's orientation on the vertical axis is called "tilt", while its orientation on the horizontal axis is called "pan". It is found that the predicted label is "neutral" when the pan angle falls between +0 and +15 degrees regardless of the tilt angle.

7) *Test scenario 7*

In this particular scenario, the test images are taken from the Stirling/ESRC 3D face dataset. The dataset has images of multiple subjects, each showing different expressions like anger, disgust, happiness, unhappiness, surprise, fear, and neutrality. Here we have studied and evaluated the performance of both approach 1 (NLSeg+method 1) and approach 2 (NLSeg+method 2). The outcomes are displayed in Table VIII, which is provided below. Here, we looked at how different facial expressions and lighting affected the ability to classify face poses. Despite these difficulties, it is found that the proposed research project performs exceptionally well.

TABLE VIII. RESULTS OBTAINED ON IMAGES OF STIRLING/ESRC 3D FACE DATASET WITH APPROACH 1 AND APPROACH 2

Filename	Input image	Expression	Camera position	Approach	Background region mask (m1)	Nose lip mask (m2)	Final mask (m)	Prediction	Remarks
F1002_06_L4_V4L_A.png		Angry	3–Positioned 45 degrees from subject	NLSeg + method1					Looking towards her right direction
				NLSeg + method2					Looking towards her right direction
M1000_07_L9_V9L_D.png		Disgust	4–Positioned 90 degree from subject	NLSeg + method1					Looking towards his right direction
				NLSeg + method2					Looking towards his right direction
F1002_08_L4_V4L_F.png		Fear	3–Positioned 45 degrees from subject	NLSeg + method1					Looking towards her right direction
				NLSeg + method2					Looking towards her right direction

M1000_05_L0_V0S_H.png		Happy	1–Positioned 0 degrees from subject	NLSeg + method1		Looking in neutral direction
				NLSeg + method2		Looking in neutral direction
M1000_04_L2_V2L_N.png		Neutral	2–Positioned 22.5 degrees from subject	NLSeg + method1		Looking towards his right direction
				NLSeg + method2		Looking towards his right direction
F1002_10_L0_V0S_S.png		Surprise	1–Positioned 0 degrees from subject	NLSeg + method1		Looking in neutral direction
				NLSeg + method2		Looking in neutral direction
M1000_13_L0_V0S_U.png		Unhappy	1–Positioned 0 degrees from subject	NLSeg + method1		Looking in neutral direction
				NLSeg + method2		Looking in neutral direction

8) Test scenario 8

The performance of NLSeg+method 2 was evaluated on video sequences from the "EPFL" multi-camera pedestrian videos dataset provided by cvLab, specifically from the laboratory and terrace environments. The videos were processed at 5 Frames Per Second (FPS), as the primary goal was to extract face orientation information rather than perform full video summarization, scene understanding, or activity recognition. A total of 12 videos were tested, with frame counts ranging from 3915 to 5010. The laboratory sequences feature four to six individuals entering and walking around the room over approximately 2 min 30 s, recorded at 25 FPS using the MPEG-4 codec. The terrace sequences include seven people walking for about 3 min 30 s, recorded at 25 FPS with the Indeo 5 codec. The method achieves an average latency per frame of 23.25–45.80 ms and a throughput of 21–39 FPS, exceeding the typical real-time processing threshold of 30 FPS. This ensures smooth, real-time operation suitable for applications such as surveillance and pedestrian tracking. Despite variations in camera views, the average FPS and latency remain consistently high, demonstrating the method's robustness across different scenes and angles. Each video contains multiple frames, and the system maintains consistent performance throughout, indicating reliability over extended sequences. The results obtained for the video sequence are given in Table IX. The predictions obtained on some sample video frames are illustrated in Fig. 12.

Fig. 12. Results obtained on sample frames of EPFL multi camera pedestrians video dataset.

TABLE IX. PERFORMANCE ON VIDEO DATASET EPFL-MULTI-CAMERA PEDESTRIAN

Video file	Total number of frames	Average Latency per frame	Average FPS	Throughput (frame/s)
4p-c0.avi	3915	42.43 ms	23.56	23
4p-c1.avi	3915	45.80 ms	21.83	21
4p-c2.avi	3915	44.46 ms	22.49	22
4p-c3.avi	3915	43.24 ms	23.12	23
terrace1-c0.avi	5010	30.01 ms	33.31	33
terrace1-c1.avi	5010	23.25 ms	43.00	43
terrace1-c2.avi	5010	25.75 ms	38.83	38
terrace1-c3.avi	5010	30.72 ms	32.54	32
terrace2-c0.avi	4480	30.75 ms	32.51	32
terrace2-c1.avi	4480	26.84 ms	37.25	27
terrace2-c2.avi	4480	25.42 ms	39.33	39
terrace2-c3.avi	4480	30.69 ms	32.57	32

9) Ablation study

This experiment aims to evaluate the contribution of each component to face orientation prediction in our second approach, NLSeg+method2. The study was carried out on 900 images from the Pandora dataset, categorized into three classes: left, right, and neutral. Two strategies were compared. In the first, both the U-Net-based nose-lip segmenter and the Media-pipe-based background region extractor were combined to predict face orientation. In the second, only the U-Net-based nose-lip segmenter was used. Results show that the second strategy led to a performance drop of 2.67% for the left class and 6% for the right class and neutral class. The first strategy is advantageous compared to the second, leading to improved accuracy in face orientation prediction. By combining local facial features (nose and lips) with global cues from the background, the system can better handle challenging cases such as subtle head tilts or partial occlusions. This multimodal integration in the first strategy reduces ambiguity in classification and ensures higher robustness and reliability, as reflected in the better performance scores compared to the second strategy. The outcome of the experiment is depicted in Table X given below.

TABLE X. ACCURACY OBTAINED ON IMAGES OF PANDORA DATASET WITH BOTH STRATEGIES

Class	Total Images Tested	Accuracy with Strategy 1 (UNet + Mediapipe)	Accuracy with Strategy 2 (Only UNet)
Left	300	99%	96.33%
Right	300	97.66%	91.66%
Neutral	300	78%	72%

B. Quantitative Results

This section shows the quantitative performance of the proposed research work on five benchmark datasets, namely AFLW2000-3D, BIWI-Kinect, Pandora, Pointing04, and the Stirling/ESRC 3D face dataset. Here, we studied and evaluated the performance of both approaches (NLSeg+method 1 and NLSeg+method 2). The evaluation metrics like accuracy, misclassification rate, macro-F1, weighted F1, precision, recall, and F1-Score are considered for the study. Around 120 images are used for the study. The images are manually selected and labeled into three classes, namely left, right, and neutral. Additionally, class balance is maintained by assigning 40 images to each class. As shown in Fig. 13, both approaches (NLSeg+method 1 and NLSeg+method 2) have shown the highest accuracy for the BIWI-Kinect dataset. The highest improvement is seen for the Stirling/ESRC 3D face dataset (+6.67%). Also, approach 1 outperforms approach 2 for all datasets except Pointing04. However, both approaches have shown lower accuracy for the AFLW2000-3D dataset, possibly due to varying illumination, pose, and complex background in images. The second approach, NLSeg+method 2, has the highest accuracy of 97.5% on the BIWI-Kinect dataset compared to the first approach, NLSeg+method 1. The misclassification rate is another important metric used to measure the number of incorrect predictions. It is the opposite of accuracy. As shown in Fig. 14, both the approaches (NLSeg+method 1 and NLSeg+method 2) have shown the highest misclassification rate for the AFLW2000-3D dataset and the lowest for the BIWI-Kinect dataset. The second approach, NLSeg+method 2, outperforms the first approach for all the datasets except Pointing04. The Stirling/ESRC 3D face dataset displays the greatest improvement, at −6.67%. Another important metric that treats each class equally without any bias is Macro-F1. The performance of multi-class classification tasks with imbalanced datasets is typically measured using Macro-F1. The formula for Macro-F1 is given in Eq. (6).

$$Macro-F1=\frac{1}{N}\sum_{i=1}^{N}\left(F1-score_i\right) \tag{6}$$

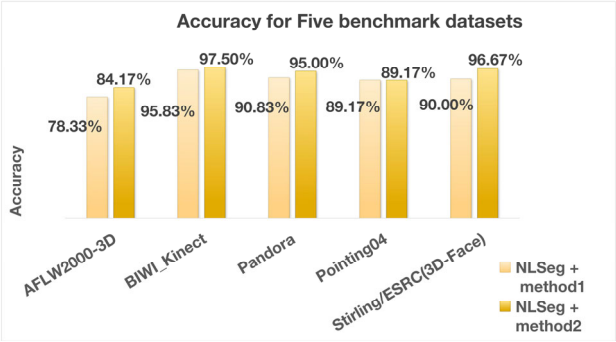


Fig. 13. Accuracy obtained for five benchmark datasets with both approaches (NLSeg+method 1 and NLSeg+method 2).

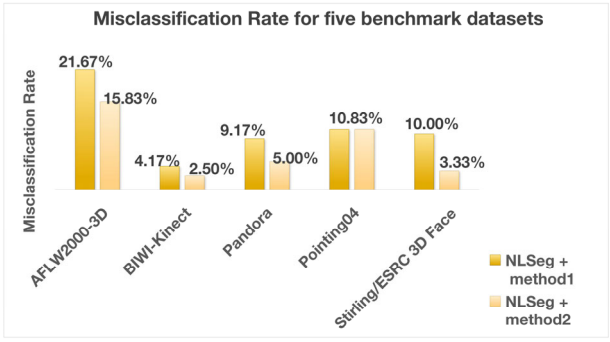


Fig. 14. Misclassification rate for five benchmark datasets with both approaches (NLSeg+method 1 and NLSeg+method 2).

As shown in Fig. 15, both approaches (NLSeg+method 1 and NLSeg+method 2) exhibit the highest Macro-F1-Score for the BIWI-Kinect dataset and the lowest score for the AFLW2000-3D dataset. The second approach outperforms the first approach for all datasets except the Pointing04 dataset. The second approach, NLSeg+method 2, has produced higher Macro-F1-Scores of 0.9748 for the BIWI-Kinect dataset and 0.9663 for the Stirling/ESRC 3D face dataset. However, for both approaches, the macro-F1-Score is above 0.8 for all datasets, which is above average for classification problems. Weighted F1 is another well-known evaluation metric used for classification problems in the presence of imbalanced datasets. Weighted F1 is computed by taking the class-wise average of F1-Scores weighted by the number of true instances belonging to each class. The formula for weighted F1 is indicated below in Eq. (7).

$$Weighted - F1 = \sum_{i=1}^{N} \frac{n_i}{N} F1_i \quad (7)$$

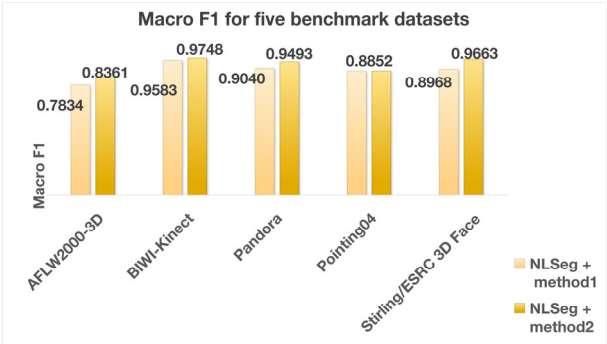


Fig. 15. Macro-F1 for five benchmark datasets with both approaches (NLSeg+method 1 and NLSeg+method 2).

The weighted-F1 value lies between 0 and 1. The value closer to 1 indicates good performance, and the value closer to 0 indicates poor performance. As shown in Fig. 16, both approaches (NLSeg+method 1 and NLSeg+method 2) exhibit the highest score for the BIWI-Kinect dataset and a lower score for the AFLW2000-3D dataset. The highest weighted F1-Scores of 0.9752 are observed for the BIWI-Kinect dataset with the second approach.

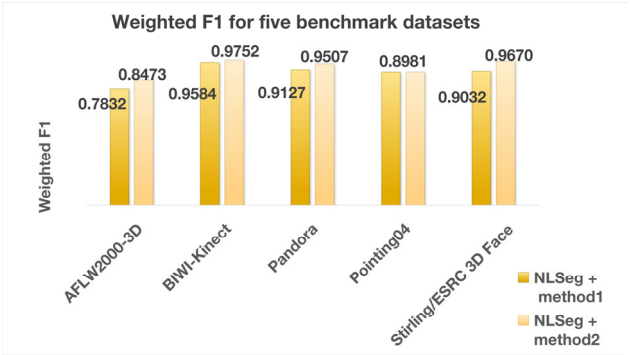


Fig. 16. Weighted-F1 for five benchmark datasets with both approaches (NLSeg+method 1 and NLSeg+method 2).

Precision is one way to measure the quality of positive predictions. The formula for calculating the precision is provided in Eq. (3). Fig. 17 exhibits the precision score obtained for three classes, i.e., left, right, and neutral. For the class "left", the second approach outperforms the first approach for all datasets except Pandora and Pointing04. For the class "right", the second approach outperforms the first approach for all datasets except BIWI-Kinect and Pointing04. For class "neutral", the second approach outperforms the first approach for all datasets except for AFLW2000-3D. The precision value is high or often perfect (near 1) for left and right classes across both approaches (NLSeg+method 1 and NLSeg+method 2). The precision score is lower for the neutral class compared to other classes across both approaches (NLSeg+method 1 and NLSeg+method 2).

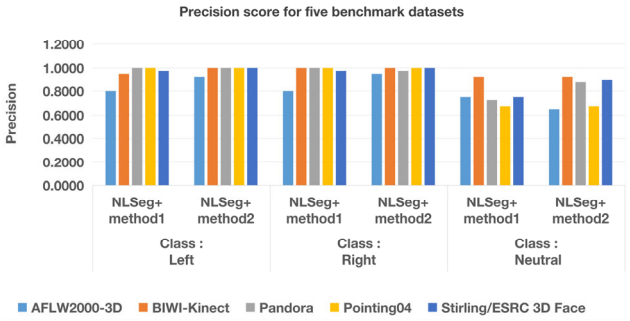


Fig. 17. Precision score for five benchmark datasets for all classes across both approaches (NLSeg+method1 and NLSeg+method2).

Recall, also called the True Positive Rate (TPR), indicates the amount of actual positives that the model was able to identify. The formula for calculating the recall is provided in Eq. (4). Fig. 18 shown below, exhibits the recall score obtained for three classes, i.e., left, right, and neutral. In the case of class "left" and "right", the second approach outperforms the first approach for all datasets except for the Pointing04 dataset. In the case of class "neutral", the recall score is 1 (100%) for the dataset Pointing04 and BIWI-Kinect across both approaches. The recall score is higher for the neutral class compared to other classes across both approaches (NLSeg+method 1 and NLSeg+method 2). The F1-Score is another prominent evaluation metric used in machine learning. It is used to evaluate the performance of the classification model by considering precision as well as recall. The F1-

Score is obtained by taking the harmonic mean of precision and recall.

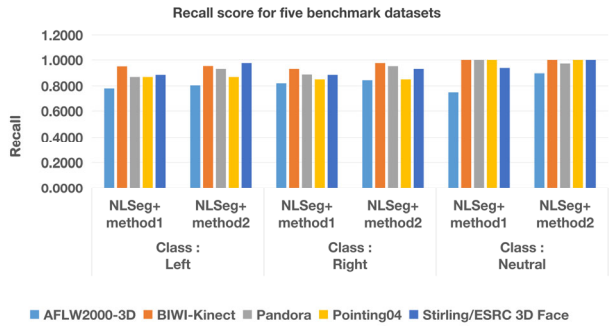


Fig. 18. Recall score for five benchmark datasets for all classes across both approaches (NLSeg+method1 and NLSeg+method2).

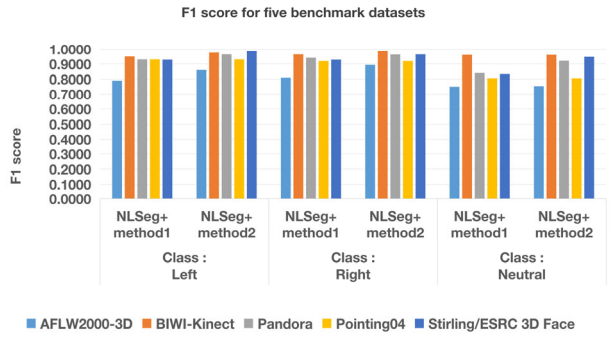


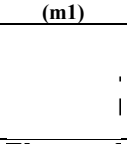
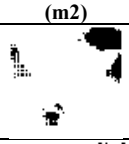
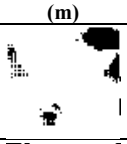
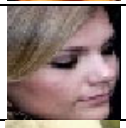
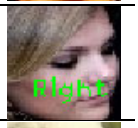
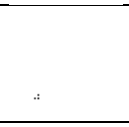
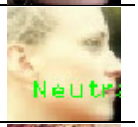
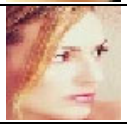
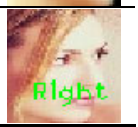
Fig. 19. F1-Score for five benchmark datasets for all classes across both approaches (NLSeg+method1 and NLSeg+method2).

The formula for calculating the F1-Score is provided in Eq. (5). Fig. 19 shown below exhibits the F1-Score obtained for three classes, i.e., left, right, and neutral. The highest F1-Score is obtained for the BIWI-Kinect dataset, indicating excellent classification performance across all classes and approaches. For all the classes, the second approach outperforms the first approach except for the Pointing04 dataset. The AFLW dataset shows comparatively lower F1-Score in the neutral class (around 0.75), suggesting potential difficulty in correctly identifying neutral class samples in this dataset. The Pandora dataset also shows a relatively low score (0.8406) for the neutral class using approach 1 (NLSeg+method 1).

VI. Empirical Evaluation: Comparative and Error Analysis

The error analysis has been performed on the images of the AFLW2000-3D dataset, has been shown in Table XI. For images 1 and 2, the segmentation model exhibited failure cases primarily due to two factors. First, the Media-pipe based selfie segmentation model was unable to accurately delineate the background region in mask 1. Second, the nose-lip segmenter incorrectly identified the nose-lip region, misclassifying other facial areas with similar visual characteristics. This misclassification likely arose from feature similarity and local texture ambiguities, resulting in perceptual illusions during segmentation. In image 3, the hair region was erroneously segmented as part of the background by the MediaPipe selfie segmentation model. However, the nose-lip region was correctly localized. In image 4, both the background mask and the nose-lip mask were inaccurately generated, leading to incorrect predictions in subsequent stages of processing. Similarly, in image 5, the MediaPipe model misinterpreted the hair region as background, while the segmentation of the nose-lip region was also imprecise, further contributing to erroneous output predictions. The failure cases observed in the provided images can be attributed to limitations in both the MediaPipe-based selfie segmentation model and the nose-lip region segmenter.

TABLE XI. Failure Cases of AFLW2000-3D Dataset

No	Input image	Background region mask (m1)	Nose-lip region mask (m2)	Final mask (m)	Predicted class	Actual class	Output
1					Left	Right	
2					Left	Right	
3					Right	Left	
4					Neutral	Left	
5					Right	Left	

The MediaPipe segmentation model relies on a lightweight encoder-decoder architecture optimized for real-time human segmentation. However, it often struggles to differentiate between hair regions and background when there is low contrast, strong lighting, or similar color tones between hair and the background. This occurred because the boundary textures and color gradients between the hair and the background were not sufficiently distinct for the model's pretrained feature extractor. The nose-lip segmenter failed primarily due to feature ambiguity and occlusion effects. The model incorrectly localized the nose-lip area, mistaking visually similar regions (e.g., ear or forehead highlights) as part of the target mask. This issue can also arise when illumination variations alter local feature intensities, causing confusion in models trained on other datasets.

As presented in the proposed research work, two different approaches for generating the background region mask m1 are tested against different datasets. Table XII given below shows the comparison between the two approaches with respect to different characteristics. This research is compared with a handful of similar works done by researchers in the past. The comparison is highlighted in Table XIII. The comparison is based on factors like facial features, methods, and face pose classes.

TABLE XII. COMPARATIVE ANALYSIS OF BOTH APPROACHES AGAINST KEY CHARACTERISTICS

Characteristics	Approach 1: NLSeg+method 1	Approach 2: NLSeg+method 2
Methodology	Using K-means and HSV based skin color model.	Using media-pipe based selfie segmentation.
Supervised or Unsupervised	Unsupervised	Supervised
Training data	Since it is unsupervised it constructs the background region mask m1 through color based segmentation without additional training phase.	Supervised and trained on large dataset focused on people in different environments (indoor, outdoor, different lighting, etc.).
Input Image resolution	64×64	256×256
Computational speed	Slightly slower than Media-pipe selfie segmentation.	Faster
Semantic Segmentation	Yes. Able to segment the hair,skin and background region.	Not able to perform full semantic segmentation of hair, skin and clothes etc. Only focused on extracting background region from foreground(person) region.
Real Time	Yes. Can be used in real time.	Lightweight and optimized for real time applications.

TABLE XIII. COMPARISON OF PROPOSED RESEARCH WORK

Reference	Facial features	Method used	Categories for face pose classification
[46]	Haar like features	Here, the traditional Haar-Cascade classifier is improved for facial alignment detection over a wide range of head position angles. The study is conducted with data from 10 subjects between the ages of 20 and 24. The efficiency of the research has been tested in controlled and uncontrolled environments. In a controlled environment, the subjects are unable to move their heads freely. The subjects are wearing spectacles and caps. The proposed research work provides accurate results in both controlled and uncontrolled environments.	Front, Right, Left
[47]	68 facial landmarks	Have proposed an efficient method for recognizing the most important facial gestures based on 68 facial features. In the first stage, 68 facial features are recognized, and head pose is estimated using a neural network. The head pose probability is calculated for five classes, namely 0 (angles between −30 and +30), −60 (angles less than −60), +60 (angles greater than +60), −30 (angles between −60 and −30), and +30 (angles between +30 and +60). Zone-wise patches are created for the mouth, eye, and eyebrow regions. At the second level, the estimated head pose and zone patch data are used to determine the probability of the gesture. The gestures considered for the mouth are smile, frown, kiss, and neutral. The gestures considered for the eye and eyebrow region are frown, raise, and neutral.	Head pose front, Head pose right profile, Head pose right-front, Head pose left profile, Head pose left-front
[21]	Facial features	Have proposed a method for estimating head posture to monitor the relationship between passengers and drivers. It has been shown that passengers look straight ahead most of the time and are not oriented sideways. The main objective of the research is to analyze the behavior and physical and mental state of passengers to develop intelligent autonomous vehicles. The data is collected by Multi-Aspect Real-World Integrated Neuro-imaging (MARIN) with Commercial-off-The-Shelf (COTS) technology to record Electroencephalograms (EEGs), Electrocardiograms (ECGs), Electrodermal Activity (EDA), Photoplethysmograms (PPGs), and skin temperature while driving. The classifier used for the classification of head posture is VGG16.	Left, Straight, Right
Ours	Nose, lip and background region	Proposed work	Left, Right, Neutral

VII. Conclusion and Future Scope

The research is very robust since it predicts the facial orientation even when the skin, hair, and background region are incorrectly segmented by the K-means algorithm or the face has a broad range of head pose angles. The efficiency of this research work is tested in both controlled and uncontrolled environments. In an uncontrolled environment, the faces are occluded due to the presence of spectacles, sunglasses, scarves, caps, etc. Also, in an uncontrolled environment, the images have complex backgrounds, varied lighting, and facial expressions. Despite these challenges, the proposed system has an accuracy of over 90 percent in an uncontrolled environment. It has an accuracy of 98 percent with a misclassification rate of 2 percent in a controlled environment. From this study, it can be concluded that the nose, lip region, and background of the facial image are very robust features for head pose estimation when both approaches are considered. The U-Net segmentation model utilized here is very fast and accurate with a high IOU score. Both approaches used for generating mask m1 are equally effective. However, approach 2 is computationally faster than approach 1. This is because approach 2 employs the MediaPipe-based selfie segmentation model, which is a pre-trained model specifically optimized for human segmentation. In contrast, K-means is an unsupervised and generic segmentation method. The MediaPipe model demonstrates high accuracy in detecting and segmenting humans, even under challenging conditions such as poor lighting, varied poses, or complex backgrounds. On the other hand, K-means often struggles to correctly separate the background when its colors are similar to skin tone or clothing. Moreover, the MediaPipe segmentation model is highly optimized for real-time performance on CPUs, GPUs, and mobile devices.

The effectiveness and robustness of NLSeg+method 1 and NLSeg+method 2 have been evaluated across six different datasets, each featuring varying illumination conditions. Notably, the AFLW2000-3D dataset includes significant variations in head pose and lighting, yet the proposed methods maintain strong performance across all conditions. These evaluations demonstrate that the proposed approach is robust to changes in illumination and delivers consistently reliable performance across diverse real-world lighting environments.

The experimental work in this study was conducted on an Acer Swift 3 SF314-52-50FX laptop operating in a Linux environment. The system is equipped with an Intel Core i5 (7th Generation) processor and 8 GB of DDR4 RAM, providing adequate computational resources for executing the proposed algorithms. It features an Intel integrated HD Graphics 620 GPU, which supports efficient graphical processing for image-based tasks. On this configuration, the proposed method achieves a throughput of 21–39 FPS, thereby exceeding the typical real-time processing threshold of 30 FPS. This demonstrates the efficiency of the approach even on a mid-range CPU-based system, and the performance is expected to improve significantly when executed on a dedicated GPU platform.

The primary objective of this paper is to perform discrete head pose classification into three categories—left, right, and neutral—rather than continuous head pose estimation. While previous studies such as Hopenet, FSA-Net, and Transformer-based models have primarily focused on regressing Euler angles for continuous head pose estimation, our work focuses on discrete class-based orientation recognition, which is more relevant for applications such as human-computer interaction and attention analysis, where categorical orientation is sufficient. Furthermore, the proposed framework is designed to be extendable, with the potential to incorporate Euler angle estimation and finer-grained discrete classes (e.g., slightly left, extreme left, slightly right, extreme right) in future work. Thus, while our current focus differs from Hopenet or FSA-Net in its objective, our approach complements these models and serves as a foundational step toward a unified system capable of both discrete and continuous pose estimation. The research work can be employed in some scenarios as given below.

Intention estimation of pedestrians: An autonomous vehicle should be capable enough to analyze the intention of pedestrians on the road for safe and comfortable driving. This intention can be determined by predicting face orientation. Once the face orientation is estimated, it is possible to find if a person is heading in a left or right direction.

Human-robot interaction: Robots should have the potential to identify a person's gaze direction and realize the scene or context of the environment.

Attention analysis: The proposed work can be utilized in large supermarkets to analyze the customer's attention towards various product sections like frozen foods, groceries, home supplies, etc. It can be used for student attention monitoring in virtual or physical classrooms. Hence, the research work given can be adapted and used in various intelligent video surveillance-based applications in the future.

Conflict of Interest

The authors declare no conflict of interest.

Author Contribution

Shet Reshma Prakash: Study conception and design, data collection, data annotation, implementation, analysis and interpretation of results and manuscript preparation. V. N. Manju: Proofreading, verification and interpretation of results, reviewing, supervision and editing. All authors had approved the final version.

References

[1] O. Elharrouss, S. Al-Maadeed, N. Subramanian *et al*., "Panoptic segmentation: A review," arXiv Print, arXiv:2111.10250, 2021. doi: 10.48550/arXiv.2111.10250

[2] A. Chaudhary and V. Bhattacharjee, "An efficient method for brain tumor detection and categorization using MRI images by K-means

clustering & DWT," *International Journal of Information Technology*, vol. 12, no. 1, pp. 141–148, 2020.
[3] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3431–3440.
[4] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Proc. Medical Image Computing and Computer-Assisted Intervention*, 2015, pp. 234–241.
[5] L. C. Chen, G. Papandreou, I. Kokkinos *et al*., "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 4, pp. 834–848, 2017.
[6] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2881–2890.
[7] A. Kirillov, E. Mintun, N. Ravi *et al*., "Segment anything," in *Proc. IEEE/CVF International Conference on Computer Vision*, 2023, pp. 4015–4026.
[8] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proc. IEEE International Conference on Computer Vision*, 2017, pp. 2961–2969.
[9] J. Redmon, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 779–788.
[10] D. Bolya, C. Zhou, F. Xiao, and Y. J. Lee, "Yolact: Real-time instance segmentation," in *Proc. IEEE/CVF International Conference on Computer Vision*, 2019, pp. 9157–9166.
[11] R. Mohan and A. Valada, "Efficientps: Efficient panoptic segmentation," *International Journal of Computer Vision*, vol. 129, no. 5, pp. 1551–1579, 2021.
[12] N. A. Ibraheem, M. M. Hasan, R. Z. Khan, and P. K. Mishra, "Understanding color models: A review," *ARPN Journal of Science and Technology*, vol. 2, no. 3, pp. 265–275, 2012.
[13] N. Dhanachandra, K. Manglem, and Y. J. Chanu, "Image segmentation using K-means clustering algorithm and subtractive clustering algorithm," *Procedia Computer Science*, vol. 54, pp. 764–771, 2015.
[14] C. Lugaresi, J. Tang, H. Nash *et al.*, "Mediapipe: A framework for building perception pipelines," arXiv Print, arXiv:1906.08172, 2019. doi: 10.48550/arXiv.1906.08172
[15] H. M. Shah, A. Dinesh, and T. S. Sharmila, "Analysis of facial landmark features to determine the best subset for finding face orientation," in *Proc. 2019 International Conference on Computational Intelligence in Data Science (ICCIDS)*, 2019, pp. 1–4.
[16] M. H. Yang, D. J. Kriegman, and N. Ahuja, "Detecting faces in images: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 1, pp. 34–58, 2002.
[17] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proc. 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 1, 2001.
[18] A. F. Abate, P. Barra, C. Bisogni, M. Nappi, and S. Ricciardi, "Near real-time three axis head pose estimation without training," *IEEE Access*, vol. 7, pp. 64256–64265, 2019.
[19] H. Samet, "The quadtree and related hierarchical data structures," *ACM Computing Surveys (CSUR)*, vol. 16, no. 2, pp. 187–260, 1984.
[20] A. Saeed and A. Al-Hamadi, "Boosted human head pose estimation using kinect camera," in *Proc. 2015 IEEE International Conference on Image Processing (ICIP)*, 2015, pp. 1752–1756.
[21] S. Biswas, D. Chambers, W. D. Hairston, and S. Bhattacharya, "Head pose classification for passenger with CNN," *Transportation Engineering*, vol. 11, 100157, 2023.
[22] T. Kaur and T. K. Gandhi, "Deep convolutional neural networks with transfer learning for automated brain image classification," *Machine Vision and Applications*, vol. 31, no. 3, p. 20, 2020.
[23] J. Xia, L. Cao, G. Zhang, and J. Liao, "Head pose estimation in the wild assisted by facial landmarks based on convolutional neural networks," *IEEE ACCESS*, vol. 7, pp. 48470–48483, 2019.
[24] K. O'Shea and R. Nash, "An introduction to convolutional neural networks," arXiv Print, arXiv:1511.08458, 2015. doi: 10.48550/arXiv.1511.08458
[25] S. K. Upadhyay and A. Kumar, "A novel approach for rice plant diseases classification with deep convolutional neural network," *International Journal of Information Technology*, vol. 14, no. 1, pp. 185–199, 2022.
[26] R. Singh and B. B. Agarwal, "An automated brain tumor classification in MR images using an enhanced convolutional neural network," *International Journal of Information Technology*, vol. 15, no. 2, pp. 665–674, 2023.
[27] S. R. Prakash and P. N. Singh, "Object detection through region proposal based techniques," *Materials Today: Proceedings*, vol. 46, pp. 3997–4002, 2021.
[28] X. Zhu, Z. Lei, X. Liu *et al*., "Face alignment across large poses: A 3d solution," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 146–155.
[29] S. Li, X. Ning, L. Yu *et al*., "Multi-angle head pose classification when wearing the mask for face recognition under the COVID-19 coronavirus epidemic," in *Proc. 2020 International Conference on High Performance Big Data and Intelligent Systems (HPBD&IS)*, 2020, pp. 1–5.
[30] S. Ge, J. Li, Q. Ye, and Z. Luo, "Detecting masked faces in the wild with lle-cnns," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2682–2690.
[31] S. R. Prakash and P. N. Singh, "Background region based face orientation prediction through HSV skin color model and K-means clustering," *International Journal of Information Technology*, vol. 15, no. 3, pp. 1275–1288, 2023.
[32] X. Hu and B. Huang, "Face detection based on SSD and CamShift," in *Proc. 2020 IEEE 9th Joint International Information Technology and Artificial Intelligence Conference (ITAIC)*, 2020, vol. 9, pp. 2324–2328.
[33] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.
[34] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," arXiv Print, arXiv:1502.03167, 2015. doi: org/10.48550/arXiv.1502.03167
[35] A. Dutta and A. Zisserman, "The VIA annotation software for images, audio and video," in *Proc. 27th ACM International Conference on Multimedia*, 2019, pp. 2276–2279. https://www.robots.ox.ac.uk/~vgg/software/via/
[36] Z. Pan, J. Xu Y. Guo *et al*., "Deep learning segmentation and classification for urban village using a worldview satellite image based on U-Net," *Remote Sensing*, vol. 12, no. 10, p. 1574, 2020.
[37] R. Gao, Z. Sun, W. Li *et al.*, "Automatic coal and gangue segmentation using u-net based fully convolutional networks," *Energies*, vol. 13, no. 4, p. 829, 2020.
[38] M. Z. A. Hamid, K. Daud, Z. H. C. Soh *et al*., "Enhanced solar panel segmentation and hotspot recognition using U-Net: A multiclass semantic segmentation approach," *Journal of Electrical and Electronic Systems Research (JEESR)*, vol. 26, no. 1, pp. 27–33, 2025.
[39] F. Shaar, A. Yılmaz, A. E. Topcu, and Y. I. Alzoubi, "Remote sensing image segmentation for aircraft recognition using u-net as deep learning architecture," *Applied Sciences*, vol. 14, no. 6, p. 2639, 2024.
[40] G. Fanelli, M. Dantone, J. Gall *et al*., "Random forests for real time 3d face analysis," *International Journal of Computer Vision*, vol. 101, no. 3, pp. 437–458, 2013.
[41] N. Gourier, "Estimating face orientation from robust detection of salient facial features," in *Proc. Pointing 2004, ICPR, International Workshop on Visual Observation of Deictic Gestures*, 2004.
[42] G. Borghi, M. Venturelli, R. Vezzani, and R. Cucchiara, "Poseidon: Face-from-depth for driver pose estimation," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 4661–4670.
[43] X. Zhu, Z. Lei, X. Liu *et al*., "Face alignment across large poses: A 3d solution," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 146–155.
[44] Stirling-ESRC 3D Face Database. [Online]. Available: https://pics.stir.ac.uk/ESRC/
[45] F. Fleuret, J. Berclaz, R. Lengagne, and P. Fua, "Multicamera people tracking with a probabilistic occupancy map," *IEEE*

Transactions on Pattern Analysis and Machine Intelligence, vol. 30, no. 2, pp. 267–282, 2008.

[46] M. S. A. Rosli, R. Yahya, R. Jailani, N. K. Zakaria, and H. Supriyono, "Real-time head pose estimation using haar cascade classifier for visual attention application," *SSRG International Journal of Electronics and Communication Engineering*, vol. 11, no. 3, pp. 130–140, 2024. https://doi.org/10.14445/23488549/IJECE-V11I3P114

[47] J. Goenetxea-Imaz, L. Unzueta-Irurtia, U. Elordi-Hidalgo, O. Otaegui-Madurga, and F. Dornaika, "Efficient multi-task based facial landmark and gesture detection in monocular images," in *Proc. 16th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications*, 2021, vol. 5, pp. 680–687.