Multimedia Tools and Applications (2024) 83:73627–73648
https://doi.org/10.1007/s11042-024-19284-8

1232: HUMAN-CENTRIC MULTIMEDIA ANALYSIS

Optimizing facial feature extraction and localization using YOLOv5: An empirical analysis of backbone architectures with data augmentation for precise facial region detection

Srishti Chanda[1] · Yachika N. Kumar[1] · Shrankhla Srivastava[1] · Ritu Rani[1] · Manu Shree[1] · A. K. Mohapatra[1]

Received: 24 May 2023 / Revised: 12 April 2024 / Accepted: 22 April 2024 /
Published online: 3 May 2024

Abstract
The task of object detection in computer vision revolves around the identification of objects within images or videos. A specific subtask within object detection is face detection, which focuses on detecting human faces. Within the realm of face detection, an important research area is facial feature detection, which has diverse applications ranging from facial recognition to emotion detection and facial expression analysis. The crucial step in facial feature detection is the identification and localization of key facial features such as the eyes, eyebrows, nose, mouth, and chin, which can also be called facial region detection. Face region detection can be done in two ways: landmark detection and Bounding box- based detection. Bounding boxes offer computational benefits such as increased speed and efficiency. They are preferable when the objective is to accurately detect and locate the presence of an object or face in an image or video frame. Although most of the existing algorithms for facial feature detection based on bounding box predictions typically treat the eyes as a single entity, our approach using YOLOv5 addresses the separation of left and right eye detection. In this research study, we conducted experiments using YOLOv5, which provides bounding box predictions. We used a subset of LFW (Labelled Faces in the Wild) Dataset which we augmented using GFP-GAN, Gaussian Noise, Image Sharpening, and CLAHE. We explored the effectiveness of different backbone architectures when applied to YOLOv5 for the task of facial region detection. We evaluated three popular backbone networks: EfficientNet-b0, GhostNet, and CSP-Darknet53. Our objective was to identify the most suitable backbone architecture that yields accurate detection of facial features, including the left eye, right eye, nose, and lips. Our experiments show that when GhostNet is used as a backbone in the YOLOv5 architecture, it produces superior results for the detection and classification of features as compared to the other backbones. We present a detailed evaluation of our findings, including discussions of the experimental results using different IOU thresholds and backbone combinations. Our proposed methodology and findings make valuable contributions to the field of facial feature extraction and provide meaningful insights into the potential and performance of YOLOv5 for detecting and localizing key facial elements.

Keywords Computer vision · Facial feature detection · Left eye · Right eye · Nose · Lips · Yolov5 · EfficientNet-B0 · GhostNet · IOU · Backbone

Extended author information available on the last page of the article

1 Introduction

Object detection is a computer vision task to detect objects in images or videos. Face detection is a variant of object detection where we detect human faces. It has various applications in the fields of security and surveillance. Face recognition, on the other hand, is a biometric technology that not only detects but analyzes and identifies the human face from an already existing database. We notice how face id helps us in unlocking our mobile phones. This is a classic example of face recognition that we encounter in our daily lives. It has various other applications like biometric surveillance and criminal identification in banks, retail stores, stadiums, and airports for security purposes. Facial region detection is an essential step in face recognition and the next step after face detection. It helps in localizing and extracting relevant facial features necessary for face identification/ verification. Figure 1 depicts the basic pipeline for face identification/verification [1]. When an image is fed in, it first goes through various preprocessing steps to extract its features. Once the features are extracted from every image, it is stored in the database. For matching, once again image goes through preprocessing, and then the process of face detection takes place. If the face is detected, its features are extracted again, and a similarity metric is run through. If the match is found, the process is completed else the image is sent to the database to get stored. Face Detection has progressed drastically over the last few decades. In the 1970s, the feature-based approach came into the picture [2].

It analyzed features like skin color and face geometry, using distance, angles, and area measurements and then using these features to classify the faces. Low-level analyses like edge detection and gray information [3] were performed.

A few years later came sequential feature-searching strategies, which were based on the relative positioning of individual facial features [4]. . These techniques assumed the availability of frontal faces of the same size. However, this was not true in reality due to the varied nature of facial appearance and environmental conditions. In the 1980s, techniques like support vector machines (SVMs), principal component analysis (PCA), and Linear Discriminant Analysis (LDA) were introduced. They were simple to use and achieved high accuracy [5]. However, with advancements in hardware, improved training algorithms, and the availability of large data sets, neural networks were introduced, they show promising results and achieve state-of-the-art accuracy.

2 Facial region detection

Facial Region detection is the successive stage after face detection. It involves identifying and locating key facial features like eyes, nose, mouth, and chin. Facial features can be of different types: region [6, 7], key point (landmark) [8, 9], and contour [10, 11]. There are three types of facial region detection methods: (1) generic methods based on edges, lines, and curves; (2) feature template-based methods where each feature is characterized into a template; (3) structural matching methods like color segmentation-based and appearance-based that take into consideration geometrical constraints on the features [5]. Facial feature detection of eyes, nose, and lips is highly significant as they provide essential information about a person's age, gender, and emotions. For example, detecting eyes and their position relative to other facial features can be used to determine a person's gaze direction, which can be used to infer their level of engagement or interest in a particular task or activity. The detection of the nose can be used to estimate a person's age or ethnicity, while the detection of lips can

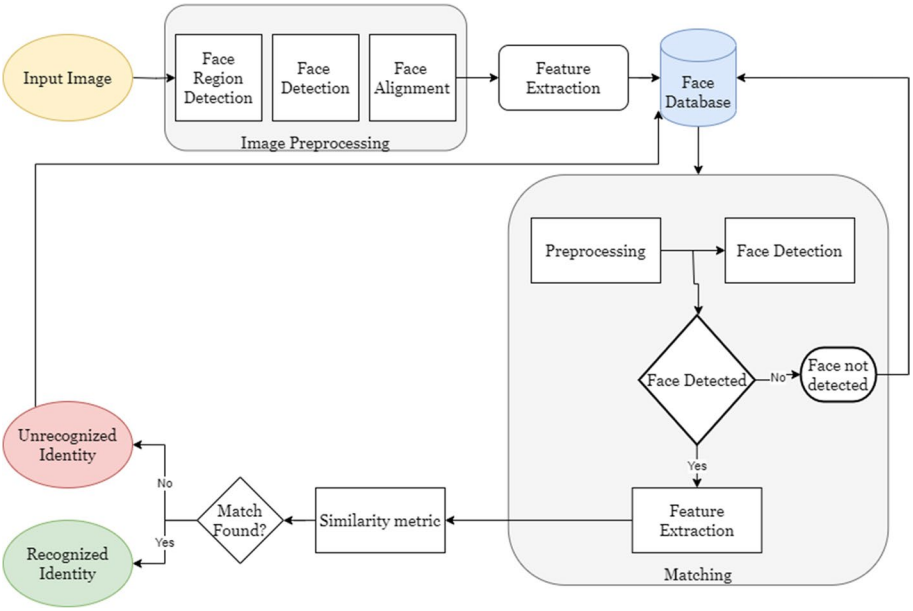

Fig. 1 Pipeline for face identification/verification

provide information about a person's emotional state or whether they are smiling. Feature detection can be done in two ways: feature point prediction or bounding box prediction.

2.1 Feature point prediction

Feature point prediction captures the location of the landmark points around facial components like eyes, nose, eyebrows, and lips. These are important for facial analysis tasks and extracting non-verbal messages like humans' identity, intent, and emotion [12]. Facial landmark detection is a popular facial feature extraction technique involving feature point prediction. Figure 2a depicts a basic diagram of feature point prediction.

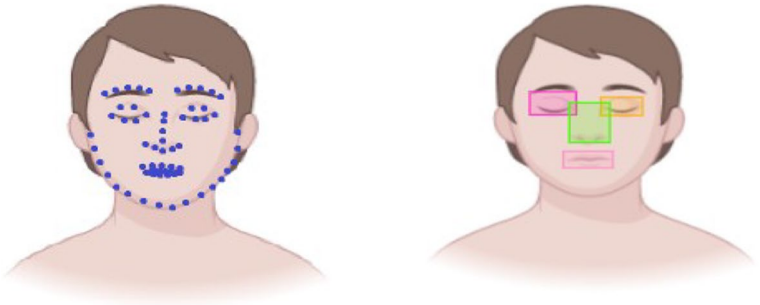

Fig. 2 **a** Feature point prediction **b** Bounding box prediction

2.2 Bounding box prediction

Bounding box prediction estimates the class and encloses the target object in a rectangular box within an image or video frame [13]. Face recognition can be either the face or facial features like eyes, nose, eyebrows, etc. Feature extraction techniques like Viola-Jones, R-CNN, and YoLo use bounding box predictions. Figure 2b depicts a basic diagram of bounding box prediction.

2.2.1 Bounding box prediction over feature point prediction

Bounding box has proved to provide a more robust and efficient solution than feature point prediction in several cases [14, 15]. have found bounding box prediction to be better than feature point prediction in terms of speed and accuracy. It has also outperformed when there is scale, orientation, and pose variation. Choosing a method is dependent on various vital factors. Still, bounding box prediction should be used when dealing with applications of person counting, object detection, and face recognition, as it achieves more accuracy and robustness. In contrast, feature point prediction can be used for tasks like face alignment, emotion recognition, and expression analysis.

3 Literature review

Facial region detection has seen significant advancements in recent years, especially with the introduction of deep learning techniques. In the early 2000s, the Viola-Jones algorithm was introduced, which marked a significant breakthrough in the field, as it allowed for rapid and accurate face detection using Haar-like features and a boosted cascade of classifiers. However, this method was still limited in its ability to accurately detect and localize facial landmarks. Later in 2006, the HOG detector, a popular feature-based object detection algorithm, was introduced. In 2010, DPM was launched, unlike the HOG detector, which uses a single rectangular window to scan the image for objects, divides the object into multiple parts, and each part is modeled separately. Later in 2013, the Deep Convolutional Network Cascade for Facial Point Detection by Sun et al. [16] was introduced, proving to be a significant advancement in the field. This method used a cascade of convolutional neural networks to detect and localize facial landmarks accurately. Since then, many researchers have built upon this work, introducing new approaches such as cascaded pyramid networks, local-global context networks, and multi-task cascaded convolutional networks, which have achieved even greater accuracy in facial landmark detection.

More recently, there has been a shift towards end-to-end learning, where both face detection and landmark detection are performed jointly in a single network. This approach has shown promising results in [14, 17].

We have presented a Table 1 that summarizes research papers on facial feature detection, highlighting the face parts detected, the approach used, the dataset used, and the evaluation metric. The papers discussed in this table include various deep learning-based methods such as convolutional neural networks (CNNs), hierarchical frameworks, and local-global context networks. The evaluation metrics used vary across the papers and may include mean error rates, accuracy, and pixel error.

Table 1 Literature review

Name	Year	Methodology	Face parts detected	Dataset	Method used	Results
Wu et al. [12]	2018	Holistic Method, CLM, RegressionBased Method	Eyes, nose, mouth, eyebrows, jawline	LFPW, AFLW, HELEN	Feature Point Prediction	Mean error rates between 3% and 10%
Hou et al. [18]	2015	Cascaded CNN	Eyes, nose, mouth, eyebrows, jawline	300 W	Feature Point Prediction	4.91% error
Zhang et al. [19]	2018	Local-global context network	Eyes, nose, mouth, eyebrows, jawline	300 W	Feature Point Prediction	Normalized mean error of 2%
Zhang et al. [14]	2016	Multi-task cascaded convolutional networks	Eyes, nose, mouth	FDDB, WIDER FACE, AFLW	Bounding box, Feature Point Prediction	99% accuracy
Sun et al. [16]	2013	Convolutional network cascade	Eyes, nose, mouth	LFPW, BioID	Feature Point Prediction	Failure rate less than 3%
Deng et al. [20]	2017	Multi-view Hourglass Model	Eyes, nose, mouth, eyebrows, jawline	300 W, AFW, Helen, Menpo	Feature point prediction	Recall of 84.5%
Colaco et al. [21]	2022	The compound model scaling and multi-scale fully connected layers	Eyes, nose, mouth, eyebrows, jawline	300 W	Feature Point Prediction	Accuracy of 90%
Yang et al. [22]	2015	Deep convolutional network	Hair, eye, nose, mouth, beard	CelebFaces, AFLW	Bounding box	Recall rate of 90.99%
Feng et al. [23]	2018	Wingloss functionin CNN	Eyes, nose, mouth	AFLW	Bounding Box	Normalized mean error of 0.0147

4 Proposed approach

In this study, two subsets of the LFW Dataset were created with sizes of 300 and 3000 samples, respectively. The second subset was generated through image augmentation techniques. To address the presence of multiple faces in the background of the images, a face detection algorithm was employed to extract the single most prominent face from each image. Subsequently, the images were resized and augmented to create the second dataset.

In order to implement YOLO for object detection, the dataset needs to be annotated in the YOLO-specific format. The annotation process was done using open-source software, ensuring compatibility with the YOLO framework. Once the images and corresponding annotations were obtained, the training process was conducted using three backbone architectures: EfficientNet-b0, GhostNet, and CSP- DarkNet53 in YOLOv5.

The performance of these models was evaluated across various Intersection over Union (IOU) thresholds for both subsets of the dataset. Finally, the obtained results were analyzed.

This section discusses the methodology used for feature detection of the Left Eye, Right Eye, Nose, and Lips. Figure 3 represents the pipeline implemented.

4.1 Dataset

Several publicly available datasets have been used for training models for facial feature detection. One can also try building their own dataset for improved performance and accuracy of the model. We have used The Labelled Faces in the Wild (LFW) dataset [24] which is a benchmark dataset for face recognition containing a set of face images gathered from the web. The LFW dataset contains more than 13,000 images of faces from more than 5,700 people, with each face labeled with the person's name. The images are relatively unconstrained, with pose, lighting, and expression variations. The dataset has folders, where each folder corresponds to a person, and the number of images per person is variable in the range of 1-100 + per person. We then used a subset of this dataset, the subset consisted of 300 images.

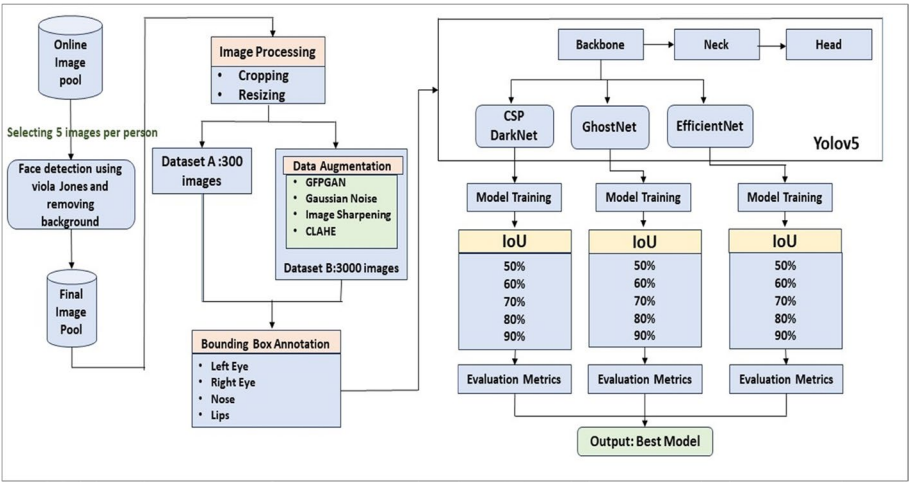

Fig. 3 Methodology

The Labeled Faces in the Wild (LFW) dataset offers several advantages that make it a valuable resource in face recognition research. Firstly, the dataset comprises of a large number of images, consisting of over 13,000 faces. This abundance of data provides ample samples for training and testing face recognition models. Secondly, the LFW dataset exhibits remarkable diversity in terms of image sources. It includes images sourced from news articles, celebrities, and ordinary individuals, resulting in a wide range of facial variations. Consequently, this diversity serves as a suitable benchmark for evaluating the performance of face recognition models across different scenarios. Moreover, the LFW dataset is meticulously labeled, with the subject's name serving as the image names. This labeling approach enhances its value for training face recognition and related problem models. Additionally, the LFW dataset incorporates significant variations in pose, expression, occlusion, background, and lighting conditions. This variability enables researchers to train models that can effectively handle and adapt to such changes.

Using the dataset also comes with certain limitations. One such limitation is the limited ethnic diversity present in the dataset, which exhibits bias towards certain faces. Consequently, the dataset's applicability to other ethnic groups becomes constrained, affecting its generalizability. Additionally, it is worth noting that a small proportion of images in the LFW dataset contain multiple faces. This poses a challenge when training models that specifically focus on single-face recognition tasks.

4.2 Image cropping and resizing

The dataset, when created, is never expected to be in proper size and form. Hence some preprocessing is always required. The first step we did in preprocessing was cropping faces from the image. The Viola-Jones technique [25] has become a popular solution for detecting faces in images. This technique becomes especially useful when working with datasets like LFW, which often include several images with multiple individuals in the background of a considered face image.

We used this technique to identify the most prominent face amongst all the other faces in the image based on the detected face area and cropped it out. Figure 4 shows the results of cropping the most prominent face using the Viola-Jones technique.

After cropping out faces with Viola-Jones, the resulting face images are likely to have varying sizes. To ensure proper compatibility and functionality as inputs for the considered models, it becomes necessary to resize all face images to a standardized dimension, in our case, we resized the images to 620×620 so that they are compatible with a wide range of models.

4.3 Image enhancement

Accurately labeling features in an image, particularly the small and intricate ones like eyes on the LFW dataset, poses a significant challenge due to its low-quality and variable posed images. We enhanced the images using GFPGAN (Generative Face Prior GAN) [26], a face super-resolution algorithm, to overcome this and ensure more accurate and reliable image labeling. GFP-GAN attempts to provide fully restored facial details and enhanced colors in a single pass by introducing a Generative Facial Prior (GFP) into the face restoration process.

This is because a pretrained face GAN's extensive and varied priors enable it to achieve a balance between fidelity and realness. GFP-GAN achieves cutting-edge outcomes compared to the other state-of- art methods by fusing the strength of the generative facial prior with the usability of facial restoration through the use of spatial feature transform layers.

Fig. 4 Viola Jones face detection

Figure 5 shows the comparison of the original and enhanced images in the GFP GAN enhancement. The first row shows the original images (a)-(d), and the second row shows the enhanced images (e)-(h).

The definition of an acceptable metric for assessing GAN's effectiveness is still an open issue with vast scope for improvement. Despite the abundance of GAN models, these models are only usually reviewed qualitatively using manual inspection to check the visual quality of the produced pictures. The manual inspection is laborious, vulnerable to error, and perhaps deceptive. Researchers today concentrate on quantitative assessment of GAN models combined with qualitative measures, each of which has advantages and disadvantages [27]. The most popular quantitative assessment metrics for GANs are Inception Score (IS) [28], Fréchet Inception Distance (FID) [29], Peak Signal to Noise Ratio (PSNR) [30], and Structural Similarity (SSIM) [31]. Table 2 shows the values of different evaluation metrics applied to the images produced by GFP-GAN.

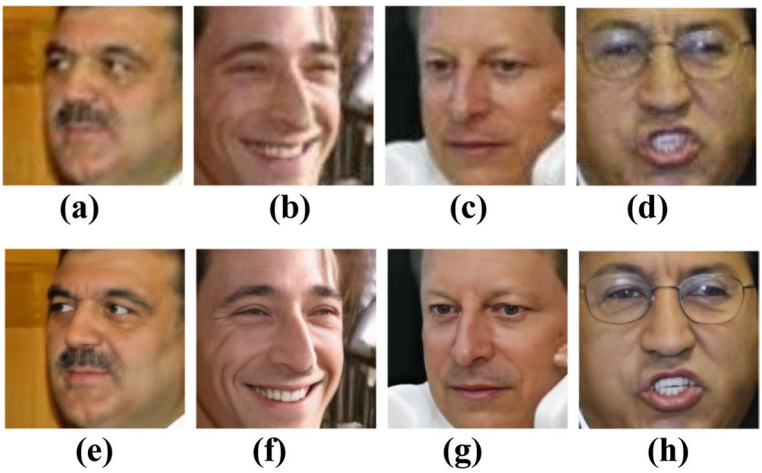

Fig. 5 Original images (**a**)–(**d**), enhanced images (**e**)–(**h**) using GFP GAN enhancement

4.4 Image augmentation

Due to the restricted amount of data that could be annotated, the annotated photos were augmented such that the same labels could be applied to many images. The image augmentation techniques employed ensured that the placement of the objects in the photos was preserved. The augmentation methods that were applied to the LFW data subset of 600 images to enlarge the dataset without labeling any more data. The augmented dataset had 3000 images. Two datasets were considered in this research which are referred to as Dataset A and Dataset B in the following sections, both these datasets are a subset of LFW. The details for the dataset composition of each dataset have been provided in Table 3. The table specifies the total number of images in a dataset along with the total number of unique individuals whose images have been used and the total number of images corresponding to each individual considered.

The augmentation techniques which were used apart from GFP-GAN were Gaussian Noise, Image Sharpening, and CLAHE. Hence in the augmented dataset, each image from the LFW dataset had four augmented versions, which were augmented using GFP-GAN, Gaussian Noise, Image sharpening, and CLAHE.

The details for GAN enhancement have been provided in the above section and Table 4 provides the PSNR, MSE and SSIM values for the remaining augmentation methods. When considering representation, SSIM values are normalized, while MSE and PSNR values are not.

Though MSE and PSNR have clear physical meanings and are convenient for mathematical optimization, they can sometimes be ineffective for assessing visual quality. SSIM, on the other hand, provides perceptual and saliency-based error, making it a more accurate measure from a semantic perspective. Simply put, while MSE and PSNR give absolute error, SSIM takes into account human perception.

Table 2 GFP-GAN evaluation metrics

Metrics	FID	IS	PSNR	SSIM
Results	53.33	6.2	34.90	0.68

Table 3 Dataset composition

Dataset	Size	Number of individuals	Number of images per person	
			Original	Augmented
Dataset A	300	60	5	0
Dataset B	3000	120	5	20

Table 4 Image augmentation evaluation

Method	PSNR	SSIM (%)	MSE
Gaussian Noise	12.28	54.6	3.86
Sharpening	61.01	99.98	0.18
CLAHE	19.14	83.5	9.71

4.5 Feature labelling

Feature labelling is a vital process that involves the assignment of informative and distinctive labels or designations to various features or variables within a given dataset. In the case of a dataset consisting of diverse facial variations, this process assumes paramount importance. The YOLOv5 model accepts image data as input along with the labels which are in YOLO-specified format [32] that requires bounding boxes to be present around specific facial features, which include the left eye, right eye, nose, and lips in our case. Extensive research was conducted to identify an appropriate dataset with suitable labels for this purpose, but none was found to be publicly available that met our specific requirements. Therefore, we resorted to manually annotating each bounding box within the Labeled Faces in the Wild (LFW) dataset images. An opensource image annotation tool was utilized for data labeling, and the resulting labels were subsequently downloaded for further use. A sample of the labelled data annotated using the tool is provided in Fig. 6.

4.6 Model training

The device used has the following specifications: OS, Linux, GPU, and Tesla T4. The training is carried out in Google Colab. All trainings were conducted in the same environment with the parameter specifications described below in Table 5. Each model was trained for 10,000 epochs, and early stopping was implemented during the training process to avoid overfitting and ensure the model's generalization ability.

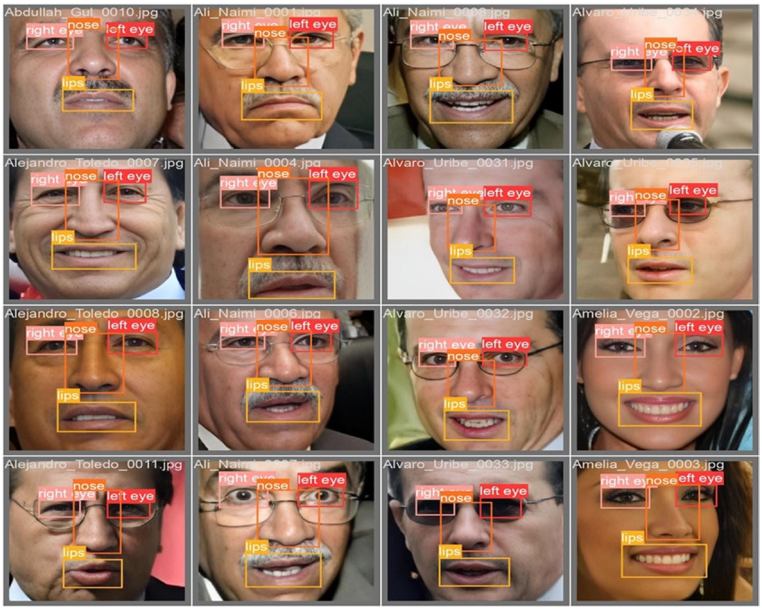

Fig. 6 Labelled data

Table 5 Parameter configuration

Batch size	64
Max Batches	10,000
Subdivision	16
Width x Height	416×416
Learning Rate	0.001
Early Stopping Patience	100

4.7 YoLov5 architecture

The network architecture of YOLOv5 [33] consists of 3 parts: the Backbone, the Neck, and the Head. The Backbone is used for feature extraction, the Neck is used for feature fusion, and Head is to make dense predictions.

YOLOv5 uses the CSPDark-net53 architecture layer as the backbone, PANet [34] with an additional SPP [35] as the neck, and YOLO [36] as the detecting head. Figure 7 represents the high-level architecture of YOLO, depicting the three parts.

1. **Backbone**: Pre-trained networks, like DarkNet53, EfficientNet, and CSP- DarkNet53, are commonly employed as backbones in image processing tasks to extract high-quality feature representations. These backbones help reduce the spatial resolution of the input image while increasing the resolution of its feature channels. This enables the network to capture rich and discriminative information for subsequent tasks such as object detection, segmentation, or classification.
2. **Neck**: The model neck enhances the extracted features from the Backbone by reprocessing and rationalizing them, enabling better utilization for subsequent tasks. Extracting feature pyramids helps the model generalize to objects of different sizes and scales. This optimization improves the model's accuracy and robustness.
3. **Head**: In a classification network, the Backbone focuses on extracting features but lacks localization capabilities. The model head is responsible for detecting object locations and classes using the feature maps from the Backbone. It applies anchor boxes on the

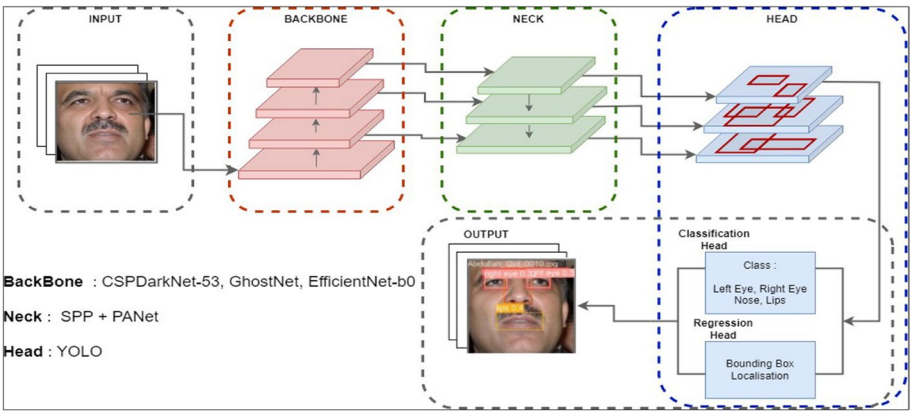

Fig. 7 YOLO architecture

feature maps and generates the final output, including classes, objectness scores, and bounding boxes.

The following backbones have been experimented with in this work.

4.7.1 CSP Dark Net-53

CSPDarknet-53 [37] is a variant of the Darknet architecture, popularly known for its use in the YOLO (You Only Look Once) object detection system. CSPDarknet-53 improves the efficiency and performance of Darknet. The network begins with a series of convolutional layers for initial feature extraction. It incorporates the concept of the "CSP" (Cross Stage Partial) module, which splits the feature maps into two branches. One branch goes through additional convolutional layers, while the other branch remains unchanged.

The two branches are then concatenated, allowing the model to learn various and rich features at different scales. Figure 8 represents the high-level architecture of CSP DarkNet-53, where each block represents the CSP module. The model has 53 convolutional layers and a global average pooling layer at the end. It uses the Mish activation function. CSPDarknet-53 is designed for object detection and image classification tasks that require high accuracy and computational efficiency. It is the default backbone for YOLOv5.

4.7.2 GhostNet

GhostNet [38] is a lightweight deep neural network designed for efficient image classification. It aims to reduce model size and computational complexity while maintaining high accuracy. The input image is first passed through convolutional layers and then a sequence of ghost modules. Ghost Module is a cheap version of the inverted residual block used in MobileNet, to reduce computational cost. The network also incorporates squeeze-and-excitation (SE) blocks, which dynamically recalibrate channel-wise feature responses to enhance discriminative power. After several repetitions of the ghost modules and SE blocks, global average pooling is applied to obtain a fixed-size feature vector. Figure 9 represents a high-level architecture of GhostNet. GhostNet has 54 convolutional layers and uses Ghost BN + ReLU activation function. GhostNet is designed for mobile and embedded devices, achieving high accuracy with fewer parameters than other models.

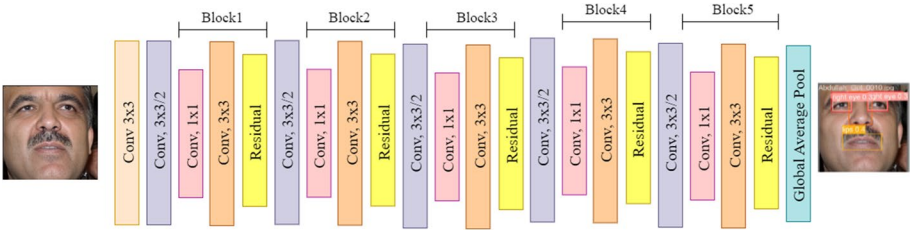

Fig. 8 CSP DarkNet-53 architecture

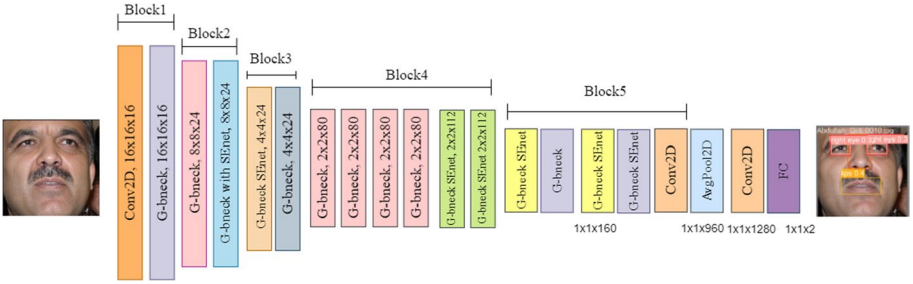

Fig. 9 GhostNet architecture

4.7.3 EfficientNet B0

EfficientNet-B0 [39] is a baseline model in a family of models developed to achieve state-of-the-art accuracy with improved efficiency in terms of parameters and computations. It uses a compound scaling method to scale the network in a balanced way. The architecture consists of efficient blocks that use a combination of depth-wise and point-wise convolutions. Figure 10 represents a high-level architecture of EfficientNet B0. The model has 20 convolutional layers and a global average pooling layer at the end. It uses the Swish activation function. EfficientNet-B0 is optimized for mobile and embedded devices with limited computational resources.

Table 6 provides the comparison between the number of layers, parameters, and some other key features of GhostNet, CSPDarkNet-53, EfficientNet B0.

4.8 Feature extraction

Following the completion of model training, we utilized its learned weights to identify the precise locations of the left eye, right eye, nose, and lips within a given input image. Subsequently, these identified coordinates were leveraged to crop the input image into sub-images corresponding to each facial feature. These sub-images were then segregated and saved within separate folders according to the corresponding facial feature. Specifically, we employed a test image dataset consisting of 600 images for dataset B and 60 for dataset A, previously unseen images to assess the model's performance. Four distinct folders were generated, one for each facial feature category, and saved the cropped-out result for each image

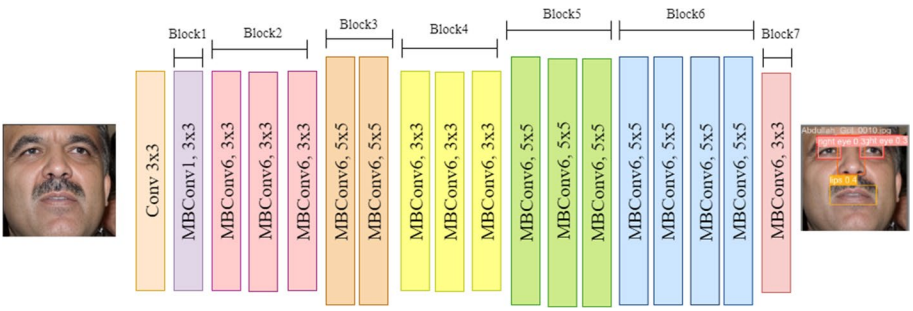

Fig. 10 EfficientNet B0 architecture

Table 6 CSPDarkNet-53, GhostNet, EfficientNEt-b0 comparative analysis

Model	Layers	Parameters	FLOPs	Input size	Key features
CSPDarkNet- 53	110	7.4 M	17.4B	640×640	CSPDarknet53 backbone, detection head with 3 YOLO layers
GhostNet	55	5.7 M	142 M	224×224	Squeeze- and- excitation module, feature fusion block
EfficientNet-B0	20	5.3 M	0.4B	224×224	EfficientNet architecture, compound scaling

was within the corresponding folder. Additionally, within each folder, the cropped sub-images of each individual's facial features were saved with the name of the original image.

5 Results

The evaluation process involved training two datasets, Dataset A and Dataset B, using YOLOV5 with three distinct backbones (CSPDarkNet-53, EfficientNet, and GhostNet). Each backbone was then evaluated using five different IOU threshold [40] values (50%, 60%, 70%, 80%, and 90%). The evaluation metrics used were Precision, Recall, and Mean Average Precision(mAP), F1-Score. The results obtained from training the models on both datasets are presented from Tables 7, 8 and 9 (Fig. 11).

From the results, it was observed that for Dataset A, GhostNet exhibited the most consistent performance across all IOU thresholds considered, displaying minimal variations in Recall and mAP scores. EfficientNet provided the second-most consistent performance, followed by CSPDarkNet-53. In terms of Precision, CSPDarkNet delivered the most consistent results across all IOU thresholds, while EfficientNet and GhostNet followed closely behind. GhostNet achieved the best performance for Recall and AP at the lowest as well as the highest IOU thresholds, followed by EfficientNet. In contrast, the performance of CSPDarkNet varied for Precision at the highest (0.9) and the lowest (0.5) IOU thresholds considered. In the case of Dataset B, it was observed that GhostNet provided the most consistent results across all IOU thresholds for all evaluation metrics, followed by EfficientNet and CSPDarkNet. When considering the lowest IOU thresholds, CSPDarkNet exhibited the best performance across all evaluation metrics, followed by GhostNet and DarkNet. Conversely, GhostNet outperformed the other backbones for the highest IOU thresholds, followed by EfficientNet and CSPDarkNet.

The F1-scores for all the backbones and the classes at 0.6 IOU have been provided in Table 7. It can be observed that the F1 scores for Dataset B are higher than those of Dataset A. Also, the F1-Scores for both the eyes are lower compared to lips and nose for both the datasets, but are significantly lower in the case of Dataset A.

Tables 8 and 9 provide the results for the three backbones on precision, recall and mAP metrics for Dataset A and Dataset B respectively. In Dataset A, which comprised 300 images, CSPDarkNet-53 showcased the highest average precision across all IOUs at 63.72%, making it the preferred choice when precision is a critical consideration.

Table 7 F1-Scores (%) at 0.6 IOU threshold

Dataset	Backbone	Classes				
		All	Left eye	Right eye	Nose	Lips
Dataset A	CSPDarkNet-53	84.23	66.57	66.57	99.39	95.39
	GhostNet	84.59	66.57	66.66	99.24	97.43
	EfficientNet-b0	85.21	66.66	66.66	99.24	99.44
Dataset B	CSPDarkNet-53	96.64	94.28	94.54	99.49	99.04
	GhostNet	99.20	98.45	98.64	99.89	100
	EfficientNet-b0	98.85	97.94	97.79	99.80	99.69

Table 8 Evaluation result for Dataset A

Metrics		Precision (%)					Recall (%)					mAP(%)				
IoU		0.5	0.6	0.7	0.8	0.9	0.5	0.6	0.7	0.8	0.9	0.5	0.6	0.7	0.8	0.9
Features	Backbone															
All	CSPDarkNet-53	73.4	73.3	72.9	61.4	37.6	99	99	99	96.5	80	76.3	76.2	76.5	76.1	53.9
	GhostNet	73.3	73.3	71.8	61.5	29.8	100	100	100	99.6	98.5	81.9	82.2	82.2	81.9	69.6
	EfficientNet	74.3	74.3	71.2	58.6	31.3	99.9	99.9	99.6	99.5	95	76.9	76.9	76.9	74	62.2
Left Eye	CSPDarkNet-53	49.9	49.9	49.9	48.9	46.8	100	100	100	91.7	76	53.1	52.6	52.6	52.4	53
	GhostNet	49.9	49.9	49.4	46	30.2	100	100	100	100	98	55.7	56.7	56.7	56.1	50.4
	EfficientNet	50	50	50.1	48.5	35.6	100	100	100	100	94	57.6	57.7	57.7	57.7	50.4
Right Eye	CSPDarkNet-53	49.9	49.9	49.8	49.1	50.7	100	100	100	98.2	88	55.7	55.8	55.8	55.7	53.8
	GhostNet	50	50	49.4	50.6	41.4	100	100	100	98.2	96	72.8	73	73	73	70.2
	EfficientNet	49.9	49.9	49.6	49	35.8	99.6	99.6	98.5	98.2	86	50.9	50.9	50.9	50.8	43.4
Nose	CSPDarkNet-53	98.8	98.8	97.7	68.5	26.6	100	100	100	100	78	99.5	99.5	99.5	98.6	57.5
	GhostNet	98.5	98.5	98.5	73.4	16.9	100	100	100	100	100	99.5	99.5	99.5	99.3	67.3
	EfficientNet	98.5	98.5	88.4	53.2	20.8	100	100	100	100	100	99.5	99.5	99.4	88.2	70.2
Lips	CSPDarkNet-53	94.8	94.8	94.3	79.1	26.4	96	96	96	96	78.1	96.9	96.9	98.3	97.5	51.5
	GhostNet	95	95	89.9	75.8	30.8	100	100	100	100	100	99.5	99.5	99.5	99.4	90.3
	EfficientNet	98.9	98.9	96.7	83.6	33	100	100	100	100	100	99.5	99.5	99.5	99.5	84.6

Table 9 Evaluation result for Dataset B

Metrics		Precision (%)					Recall (%)					mAP(%)				
IoU		0.5	0.6	0.7	0.8	0.9	0.5	0.6	0.7	0.8	0.9	0.5	0.6	0.7	0.8	0.9
Features	Backbone															
All	CSPDarkNet-53	99.3	96.3	96	94.8	88.8	99.4	97	96.7	95.1	89	99.4	98.4	98.4	98.2	95.5
	GhostNet	99.2	99.2	99.2	99.2	97.8	99.2	99.2	99.2	99	98.2	99.3	99.3	99.3	99.2	98.8
	EfficientNet	98.8	98.8	98.7	98.5	96	98.9	98.9	98.9	98.8	96.9	99.2	99.2	99.2	99.1	98.4
Left Eye	CSPDarkNet-53	99.3	93.3	93.5	95.2	94	98.8	95.3	95.1	92.4	80.8	99.4	97.5	97.5	97.2	94.2
	GhostNet	98.4	98.4	98.4	98.5	96.2	98.5	98.5	98.5	98.2	97.5	99.1	99.1	99.1	99.1	98.5
	EfficientNet	98.2	98.1	98.1	98.1	94.7	97.8	97.8	97.8	97.7	94.3	99.1	99.1	99.1	98.9	97.6
Right Eye	CSPDarkNet-53	98.5	93.7	93.8	94.5	90.8	99	93.4	92.7	90	79.3	99.4	97.6	97.6	97	91.4
	GhostNet	98.8	98.8	98.8	98.4	95.5	98.5	98.5	98.5	98	96.3	99.1	99.1	99.1	98.9	98
	EfficientNet	97.6	97.6	97.5	97	95.4	98	98	98	97.6	93.8	98.9	98.9	98.9	98.7	97.4
Nose	CSPDarkNet-53	99.8	99.2	97.6	92.7	84.1	99.8	99.8	99.8	99.8	99.7	99.5	99.4	99.4	99.4	98.9
	GhostNet	99.8	99.8	99.8	99.8	99.8	100	100	100	100	99.9	99.4	99.4	99.4	99.4	99.4
	EfficientNet	99.8	99.8	99.8	99.7	97.4	99.8	99.8	99.8	99.8	99.8	99.5	99.5	99.5	99.5	99.5
Lips	CSPDarkNet-53	99.6	98.8	98.8	97	86.2	99.8	99.3	99.1	98.2	96.2	99.4	99.2	99.2	99.2	97.5
	GhostNet	100	100	100	100	99.5	100	100	100	99.7	99.2	99.5	99.5	99.5	99.5	99.5
	EfficientNet	99.6	99.6	99.5	99.1	96.6	99.8	99.8	99.8	99.8	99.7	99.4	99.4	99.4	99.4	99.3

Fig. 11 Facial feature prediction results by YOLOv5. **a** Input image (**b**) Facial features

GhostNet, on the other hand, excelled in recall with the highest average value of 99.62%, highlighting its strength in capturing a high proportion of relevant instances. GhostNet also demonstrated the leading average mAP of 79.56%, showcasing its overall performance. EfficientNet_B0 provided a balanced performance across these metrics.

It was observed that while the recall was high, the precision scores were significantly lower. While the models were sensitive to detecting positive instances, their predictions were not very precise, the models misclassified the left eye and the right eye.

However, as the dataset size increased to 3000 images in Dataset B, GhostNet consistently performed better than its counterparts, exhibiting the highest average precision, recall, and mAP across all IOUs for all facial features. With an exceptional average precision(across all IOUs) of 98.2, GhostNet stood out in terms of precision, while maintaining superior recall and mAP. The larger and more diversified dataset significantly improved precision, particularly in detecting left and right eyes. GhostNet's persistent superiority across varied dataset sizes makes it the optimal choice for this object detection task. In real-world scenarios where capturing a high proportion of relevant instances is crucial, GhostNet's emphasis on recall makes it a valuable choice, while its consistent precision across diverse datasets solidifies its overall performance. The choice between precision and recall considerations may depend on specific use cases, but GhostNet's consistent excellence establishes it as the best-performing backbone for this detection task.

The below curve shows the Recall Confidence based on the three backbones' performance.

Figure 12 represents a recall vs. confidence curve of the model on Dataset A. It shows that GhostNet and EfficientNet work better and can be preferred over CSPDarknet, which is the default backbone of YoloV5.

In Fig. 13, we observed a significant improvement in our model; however, GhostNet still works better than other architectures.

All the models can detect the features easily with an accuracy of as good as 80. Although with some more fine- tuning, we achieved even better results. But even with limited Dataset accuracy of around 70 is achievable.

Figure 14a shows the input label (ground truth) (b) shows labels predicted by CSP-Darknet53 with Dataset A (c) shows labels predicted by GhostNet with Dataset B

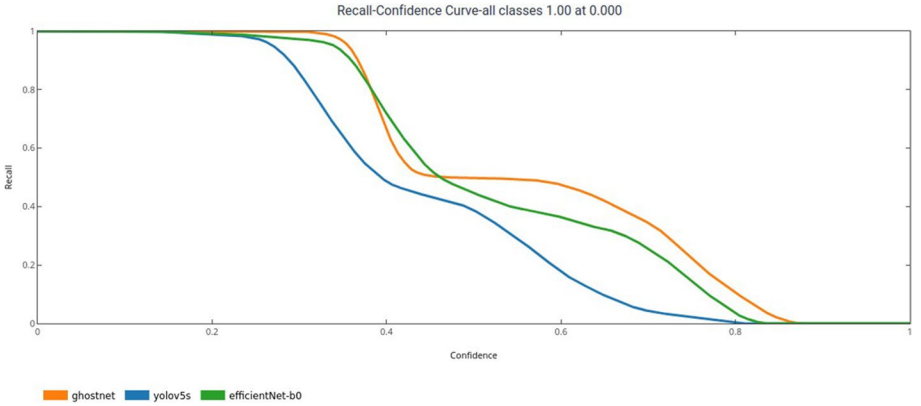

Fig. 12 Curve of Dataset A

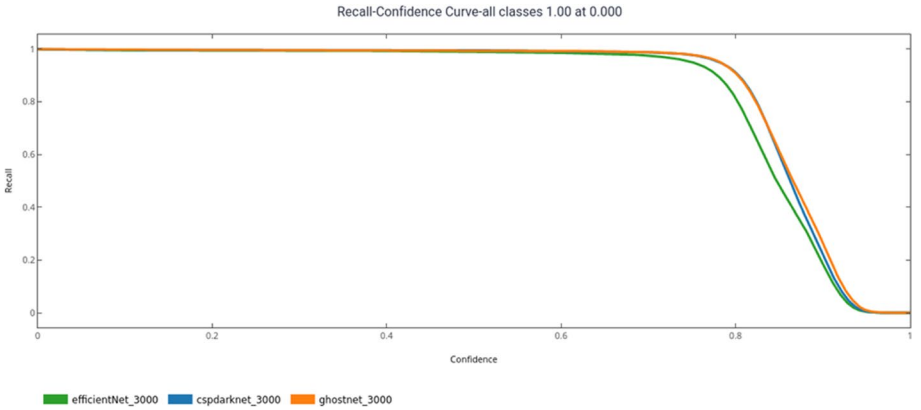

Fig. 13 Curve of Dataset B

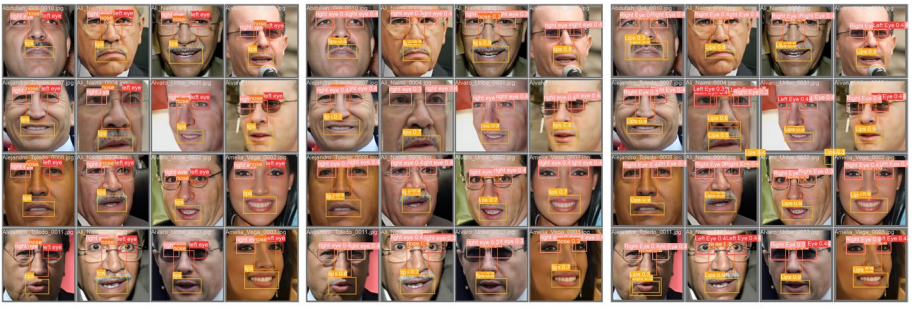

Fig. 14 Input labels and predicted labels

6 Conclusion

The research and observations presented in this study on the performance of different backbones in object detection, particularly in the context of facial feature detection, have several potential contributions to society and real-world problem-solving. Understanding the strengths and weaknesses of different backbones provides valuable insights for refining the facial feature detection systems, that can contribute to the development and enhancement of facial recognition systems. In the medical field, accurate facial feature detection is crucial for applications such as facial landmark localization in medical imaging, contributing to improved diagnostic tools and applications in plastic surgery planning. Moreover, the research's relevance extends to human-computer interaction, such as emotion recognition or gaze tracking, where advancements in facial feature detection can lead to more natural and responsive interfaces, benefiting various technological applications. In criminal investigation and forensics, the high-quality facial feature detection emphasized in this research can aid in developing advanced tools for law enforcement, facilitating accurate identification and tracking based on facial features captured in surveillance footage. By addressing challenges and offering improvements in facial feature detection through detailed comparative analyses, this research provides a comprehensive foundation for advancing technology in numerous fields, ultimately benefiting society through applications that range from security and healthcare to human-computer interaction and beyond.

7 Future work

While biometric-based recognition methods, such as iris and fingerprint recognition, have been extensively studied, limited research has been conducted on recognition systems based on facial regions. Future research will focus on utilizing the extracted facial features as a dataset to develop a face recognition system. This system will leverage a fusion network to combine the learned deep features for accurate face recognition effectively. Given the widespread availability of face images compared to the scarcity of fingerprint and iris images, this research will contribute to the advancement of facial recognition technology.

Data availability The datasets analyzed during the current study are available with the authors and may be provided on request.

Declarations

Conflict of interest On behalf of all authors, the corresponding author states that there is no conflict of interest.

References

1. Dhingra A (2017) Face identification and clustering. Rutgers The State University of New Jersey, School of Graduate Studies
2. Hjelmås E, Low BK (2001) Face detection: a survey. Comput Vis Image Underst 83(3):236–274

3. Lam KM, Yan H (1994) Facial feature location and extraction for computerized human face recognition. In ISITA'94: International Symposium on Information Theory & Its Applications 1994; Proceedings. Institution of Engineers, Australia, Barton, pp 167–171
4. Crowley JL, Berard F (1997) Multi-modal tracking of faces for video communications. In: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, (pp 640–645). IEEE
5. Bagherian E, Rahmat RWO (2008) Facial feature extraction for face recognition: a review. In: 2008 International Symposium on Information Technology (vol 2, pp 1–9). IEEE
6. Ryu YS, Oh SY (2001) Automatic extraction of eye and mouth fields from a face image using eigenfeatures and multilayer perceptrons. Pattern Recogn 34(12):2459–2466
7. Cristinacce D, Cootes TF (2003, September) Facial feature detection using AdaBoost with shape constraints. In BMVC, pp 1–10
8. Wiskott L, Fellous JM, Krüger N, Von Der Malsburg C (2022) Face recognition by elastic bunch graph matching. In Intelligent biometric techniques in fingerprint and face recognition. Routledge, pp 355–396
9. Feris RS, Gemmell J, Toyama K, Kruger V (2002) Hierarchical wavelet networks for facial feature localization. In: Proceedings of Fifth IEEE International Conference on Automatic Face Gesture Recognition. IEEE, pp 125–130
10. Cootes TF, Edwards GJ, Taylor CJ (2001) Active appearance models. IEEE Trans Pattern Anal Mach Intell 23(6):681–685
11. Xiao J, Baker S, Matthews I, Kanade T (2004) Real-time combined 2D+ 3D active appearance models. In CVPR (2), pp 535–542
12. Wu Y, Ji Q (2019) Facial landmark detection: a literature survey. Int J Comput Vision 127:115–142
13. Szegedy C, Toshev A, Erhan D (2013) Deep neural networks for object detection. Advances in neural information processing systems, 26.
14. Zhang K, Zhang Z, Li Z, Qiao Y (2016) Joint face detection and alignment using multitask cascaded convolutional networks. IEEE Signal Process Lett 23(10):1499–1503
15. Ren S, He K, Girshick R, Sun J (2015) Faster r-cnn: Towards real-time object detection with region proposal networks. Advances in neural information processing systems, 28.
16. Sun Y, Wang X, Tang X (2013) Deep convolutional network cascade for facial point detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp 3476–3483)
17. Dong X, Yu S, Wu Z, Guo Y, Yang Y (2017) Face alignment with coarse- to-fine topology. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 5325–5334
18. Hou Q, Wang J, Cheng L, Gong Y (2015) Facial landmark detection via cascade multi-channel convolutional neural network. In: 2015 IEEE International Conference on Image Processing (ICIP). IEEE, pp 1800–1804
19. Zhang J, Li H, Wang Y, Wang R, Li Z, Zuo W (2018) Robust facial landmark detection via a fully-convolutional local-global context network. arXiv Preprint arXiv :180303073
20. Deng J, Trigeorgis G, Zhou Y, Zafeiriou S (2019) Joint multi-view face alignment in the wild. IEEE Trans Image Process 28(7):3636–3648
21. Colaco S, Han D (2022) Deep learning-based facial landmarks localization using compound scaling. IEEE Access 1–1
22. Yang S, Luo P, Loy C-C, Tang X (2015) From facial parts responses to face detection: A deep learning approach. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV)
23. Feng ZH, Kittler J, Awais M, Huber P, Wu XJ (2018) Wing loss for robust facial landmark localisation with convolutional neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 2235–2245
24. Huang G, Mattar M, Lee H, Learned-Miller E (2012) Learning to align from scratch. Advances in Neural Information Processing Systems, pp 25
25. Viola P, Jones MJ (2004) Robust real-time face detection. Int J Comput Vis 57:137–154
26. Wang X, Li Y, Zhang H, Shan Y (2021) Towards real-world blind face restoration with generative facial prior. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 9168–9178
27. Alqahtani H, Kavakli-Thorne M, Kumar G, SBSSTC F (2019 An analysis of evaluation metrics of GANs. In: International Conference on Information Technology and Applications (ICITA) (vol 7)
28. Salimans T, Goodfellow I, Zaremba W, Cheung V, Radford A, Chen X (2016) Improved techniques for training gans. Advances in Neural Information Processing Systems, pp 29
29. Heusel M, Ramsauer H, Unterthiner T, Nessler B, Hochreiter S (2017) Gans trained by a two time-scale update rule converge to a local nash equilibrium. Advances in Neural Information Processing Systems, pp 30
30. Cheng Z, Sun H, Takeuchi M, Katto J (2018) Performance comparison of convolutional autoencoders, generative adversarial networks and super-resolution for image compression. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp 2613–2616

31. Wang Z, Bovik AC, Sheikh HR, Simoncelli EP (2004) Image quality assessment: from error visibility to structural similarity. IEEE Trans Image Process 13(4):600–612
32. Zhang Y, Guo Z, Wu J, Tian Y, Tang H, Guo X (2022) Real-time vehicle detection based on improved yolo v5. Sustainability 14(19):12274
33. Jocher G, Stoken A, Borovec J, Chaurasia A, Changyu L, Hogan A, …, Ingham F (2021) ultralytics/yolov5: v5. 0-YOLOv5-P6 1280 models, AWS, Supervise. ly and YouTube integrations. Zenodo
34. Liu S, Qi L, Qin H, Shi J, Jia J (2018) Path aggregation network for instance segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 8759–8768
35. He K, Zhang X, Ren S, Sun J (2015) Spatial pyramid pooling in deep convolutional networks for visual recognition. IEEE Trans Pattern Anal Mach Intell 37(9):1904–1916
36. Redmon J, Divvala S, Girshick R, Farhadi A (2016) You only look once: Unified, real-time object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 779–788
37. Bochkovskiy A, Wang CY, Liao HYM (2020) Yolov4: Optimal speed and accuracy of object detection. arXiv preprint arXiv:2004.10934
38. Han K, Wang Y, Tian Q, Guo J, Xu C, Xu C (2020) Ghostnet: More features from cheap operations. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 1580–1589
39. Tan M, Le Q (2019) Efficientnet: Rethinking model scaling for convolutional neural networks. In: International Conference on Machine Learning. PMLR, pp 6105–6114
40. Nowozin S (2014) Optimal decisions from probabilistic models: the intersection-over-union case. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 548–555

Authors and Affiliations

Srishti Chanda[1] · Yachika N. Kumar[1] · Shrankhla Srivastava[1] · Ritu Rani[1] · Manu Shree[1] · A. K. Mohapatra[1]

✉ Ritu Rani
riturani@igdtuw.ac.in

Srishti Chanda
srishti055btit19@igdtuw.ac.in

Yachika N. Kumar
yachika007btit19@igdtuw.ac.in

Shrankhla Srivastava
shrankhla070btit19@igdtuw.ac.in

Manu Shree
manu005phd21@igdtuw.ac.in

A. K. Mohapatra
akmohapatra@igdtuw.ac.in

[1] Indira Gandhi Delhi Technical University for Women, Kashmere Gate, Delhi 110006, New Delhi, India