One level deep convolutional neural network for facial key points detection

Abdelaali Benaiss[1], Rachid El Ayachi[1], Mohamed Biniz[1], Mustapha Oujaoura[2]
[1]Computer Science Department Laboratory TIAD, Faculty of Sciences and Technics, Sultan Moulay Slimane University, Beni Mellal, Morocco
[2]Department of Computer Science, Network and Telecom (GIRT), Laboratory of Informatics Mathematics and Communication Systems (MISCOM), National School of Applied Sciences (ENSA) Cadi Ayyad University, Safi, Morocco

Article Info

Article history:

Received Jun 16, 2023
Revised Dec 1, 2023
Accepted Jan 3, 2024

Keywords:

CNN
Facial landmarks detection
Facial recognition
Features extraction

ABSTRACT

Facial landmark detection has a lot of applications in face recognition, face alignment, facial expression recognition, video surveillance and security systems. In the existing literature, there are multiple methods utilizing convolutional neural networks (CNNs) that address this problem in various ways. In many cases, the models use a tree-like structure of CNNs to achieve better results. This paper proposes a combination of three parallel deep convolutional neural networks (DCNNs) to estimate the accurate localization of each keypoint. The first one focuses on the whole face to outperform five points, including the eyes, nose, and mouth corners. The second one focuses on the eyes-nose parts to outperform three points, specifically the eyes and nose. The last one focuses on the nose-mouth parts to outperform three points, namely the nose and mouth corners. Further, we combine all outputs of the three DCNNs and take the average value of each detected key point as the final output. In the first step, we improvthe the parameter efficiency and accuracy of each DCNNs through a set of experiments using the labeled face parts in-the-wild database (LFPW) and the helen facial feature dataset (Helen). Then, we demonstrate that our approach yields more accurate estimations of facial key points than two state-of-the-art methods and commercial software in terms of accuracy.

Corresponding Author:

Abdelaali Benaiss
Computer Science Department Laboratory TIAD, Faculty of Sciences and Technics
Sultan Moulay Slimane University
Beni Mellal, Morocco
Email: benaiss.abdelaali@gmail.com

1. INTRODUCTION

Facial feature detection from 2D images is a well-studied problem, as discussed in [1], especially as a preprocessing step for face recognition [2]. Thus, detection of the accurate position of facial features allows thepredictedpose of the headand facial expression analysis [3], which are very important for face recognition algorithms. In recent years, multiple researchers have addressed this problem.

At early stages, active shape model (ASM) [4] and active appearance model (AAM) [5] are among the first methods for facial landmark detection. Based on these methods, principal component analysis (PCA) is applied to simplify the problem and learn parametric features of faces for modeling variations in facial landmarks. Many years later, the art is iteratively fit into new instances. Additional algorithms proposed in [6], [7] involve matching a 3D deformable face model to 2D images using point distribution model (PDM) and constrained local model (CLM) algorithms during the fitting process. Regression-based methods are attracting increasing attention, especially the cascaded regression methods achieve great success on facial landmark detection, cascaded pose regression (CPR) [8] was the first to propose a cascaded regression framework for general object pose estimation. The method proposes cascade pose regressors combined with a random fern classifier regressor to achieve minimize the difference between the true object pose and the pose computed. This framework show better results in shapes prediction of faces, mice and fish. Another representative approach in cascaded regression methods, tree of regressors combined with boosting algorithms approaches is introduced in the field. A gradient boosting framework introduced in [9], which use a cascade of regression trees to localize the facial landmarks points through a subset of image pixels. The supervised descent method (SDM) in [10] cascades several linear regressions to predict the shape stage by stage, using shape-indexed SIFT features. The model in [11] propose robust cascaded pose regression (RCPR) to explicitly handle occlusion problems, which simultaneously predicts the locations and occlusion conditions of each landmark. Valstar *et al.* [12] proposed the boosted regression coupled with markov networks approach, which has been coined BoRMaN. It proposes an iterative method that uses Haar-like filters as descriptors of local appearance, combined with the adaptive boosting algorithm (Adaboost) for feature selection to build a regressor model. All previous methods are specifically used for building models to represent the global facial appearance and shape information.

In recent years, by introducing deep convolutional neural networks (DCNNs), the field has made significant progress in accuracy. Therefore, those approaches have gained prominence over classical ones. However, Kowalski *et al.* [13], the deep alignment network (DAN) model was introduced, which use a cascade of convolutional neural network (CNN) which performs both features extraction and regression, this method by the cascade shape regression (CSR) framework. However, the model begins with an initial estimate of the face shape, which is improved upon through multiple stages. At each stage (regressor stage), we have deep CNN structure which combined entire face image, landmark heatmap and features extraction through a connection layer and transmits it to the next stage, after a set of iteration the algorithm predict the final position of landmarks (the landmark heatmap indicates the currents estimates of landmarks positions). Some studies have demonstrated that performance can be affected by intrinsic variations in image style, such as the use of grayscale versus color images or differences in lighting conditions (e.g., bright versus dim). Therefore, it is necessary to investigate approaches for dealing with style variance. Furthermore, the approach in [14] consists of two components: The first module is the style-aggregated face generation module, which transforms the input image into various styles and combines them into a single style-aggregated face to mitigate intrinsic variance. The second is the facial landmark prediction module. This module processes both the original image and the style-aggregated image to extract complementary features, which are then fused to produce heatmap predictions. All process is repeated in cascaded manner. Additionally, the state-of-the-art is progressing towards deep structures, the model propose a novel facial landmarks detection framework coined pixel-in-pixel network (PIPNet), which consists of three new modules, namely PIP regression, neighbor regression, and self-training [15]. Each model use CNN architecture like (MobileNet [16]) or residual neural network (ResNet [17]) as backbone of structure trained on (ImageNet [18]). Another approach is presented in [19], which uses two subnets. The first subnet (lower branch) predicts landmark coordinates, and the auxiliary subnet (upper branch) estimates geometric information (Euler angles). Each subnet uses MobileNet blocks.

Furthermore, multi-levels-based approaches gain more attention in recent research works. For example, the work in [20] proposed a multi-task cascaded CNN-based framework for joint key point detection, which captures global high-level features at the first level, and then refines the position of key points in the next two levels. On the other hand, [21] proposed a convolutional neural network model that combines one hidden layer neural network, three convolutional layers, and outputs a 30-element vector, for detect 15 facial key points on given face. Xiang and Zhu [22], a model called MTCNN was proposed, which leverages the relationship between face detection and alignment to improve performance. It essentially consists of three parts: i) a proposal network (P-Net) for generating a list of the candidate windows. Then, we use it for classifying face and non-face and estimate bounding box regression vectors to face location, and non-maximum suppression (NMS) candidate merge. ii) A refine network (R-Net), rejects a lot of false candidates. iii) It is similar to R-Net, called O-Net, which outputs five facial landmarks' positions. Alternatively, the model proposed in [9] use a cascade of regressors to form a regression tree, which can be used to regress the location of facial landmarks using a sparse subset of intensity values extracted from the input image, each regressor is learning using the gradient boosting tree algorithm. CNN-based approaches deliver satisfactory state-of-the-art performance compared to classical methods but require more computational resources, especially during the learning steps (big datasets and efficient graphic processing units (GPUs)).

All the works listed above have achieved satisfactory performance in facial landmark detection, when these approaches are trained using facial database images that are typically collected under 'controlled' conditions, where the facial poses and facial expressions can only be in certain categories. However, they face challenges when facial images are collected under 'in-the-wild' conditions, where they may encounter arbitrary facial expressions, head poses, variations in illumination, facial occlusions, and other factors. In general, there is still a lack of a robust method capable of handling all variations in head pose and environmental factors. In addition, the previous works present deeper structure [13]–[19] or cascade-based approaches [9]–[22], which require more computational resources, that implement these approaches on small devices like smart phone and small personal computer or laptop is very difficult. Therefore, the contributions of this research aim to present: i) a model with minimum number of layers and levels, which show better or similar results compared with deeper structures (MobileNet and ResNet) and cascade-based approaches and ii) a procedure for select appropriate values of the model hyperparameters. The remaining sections of the paper are organized as follows. In section 2, we briefly introduce the related works and the proposed method in section 3, we present an overview of our method and implementation details, and in section 4, we provide the experimental results and discussion. Finally, we conclude our work in section 5.

2. METHOD

As a baseline for our proposed approach, we were inspired by the model in [20]. However, a cascade of CNNs, was proposed for effective facial point detection. The first level of the CNN in this model provides robust initial estimations, and the following two levels fine-tune the initial prediction to achieve high accuracy. Despite this, we cannot claim with certainty that this particular model, with three deep levels, is the best for improving the accuracy of facial landmark positions. Especially, in this work, we extract the DCNN from the first level of the baseline model and conduct experiments on model hyperparameters, we can reduce the number of levels from 3 to 1 and achieve high accuracy in detection.

The main idea of our model is to adopt the three DCNNs presented in [20] named for the whole face (F), for eyes-nose (EN), and for nose-mouth (NM), and trained them to take whole face as input and then, the first one (F) provide five points as outputs (left eye center, right eye center, nose tip and mouth corners), the second one (named EN) provide three points as outputs (two eyes centers and nose tip) and the last one (named NM) provide three points as outputs (nose tip and mouth corners). Finally, we combined all outputs of the previous DCNNs and took the average value as the output of the model. An overview of our approach is summarized in Figures 1 and 2 illustrates the deep structure of DCNN (F).

Figure 1 is shown of one level deep convolutional neural network architecture; (A) whole face image; (B) face bounding box returned by haar-like detector [23]; (C) shown output localization of five key points return by DCNN; (F) the position present on face by oval shape for each key points (LE, RE, N, LM, RM); (D) shown output localization of three key points returned by DCNN (EN) (the position of each key point (LE, RE, N) drawn on face by square shape); (E) show the localization of three key points returned by DCNN (NM) (the position of each key point (N, LM, RM) drawn on face by disk shape); (F) shown the predictions of all three DCNNs (F, EN, NM); (G) shown the final predictions given by our model (the position of each key points has been drawn on face by star shape). The final position of each key points has been calculated by mean value by each individual DCNN

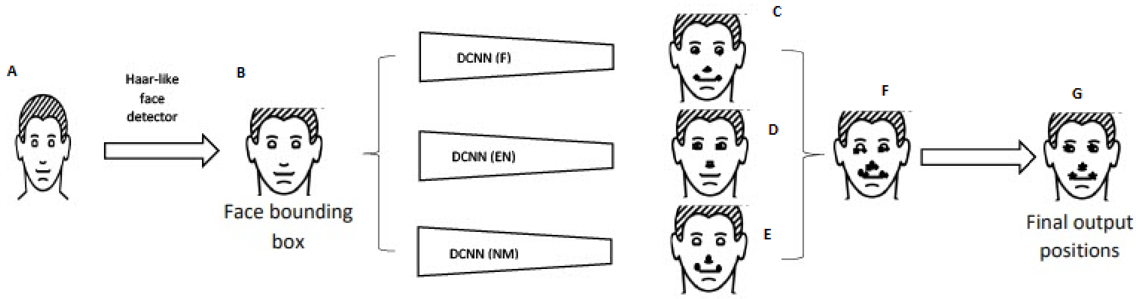


Figure 1. One level deep convolutional neural network architecture

Our contribution in this work is to simultaneously increase the accuracy of key point detection and reduce the number of model parameters compared to the related and public work in [20]. In general, one important factor in the performance of the models is their hyperparameters to select appropriate values for these hyperparameters to significantly improve the model's performance. We have been using the grid search algorithm [24] to find the optimal values for the hyperparameters of DCNNs (F, NM, and EN) and the procedure in section 2.3 to find the optimal values of training epochs and input size of model for cropped faces.

In this section, we first describe the evaluation metrics has been used to measure de precision of detection points. We then briefly describe the datasets that have been used for training, testing and validating our model. Finally, we describe the experimental tests applied to find out the best hyperparameters of our model to achieve optimal detection. We have been conducted tests on two public challenging datasets (helen and LFPW).

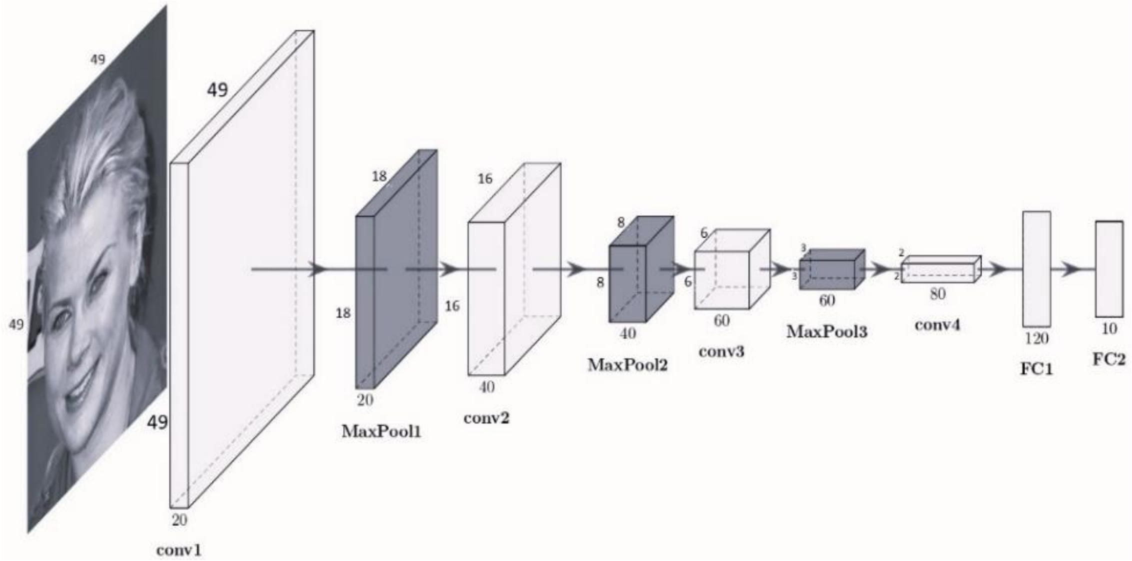


Figure 2. Structure of each DCNN. Convolutional and full connected layers illustrated by dark gray cuboids and max pooling layers illustrated by light gray cuboids whose length, width, and height denote the number of maps, and the size of each map

2.1. Evaluation metrics

Key points localization error is often normalized by the inter-ocular distance [25] or inter-pupil [26] distance; this however, presumes both eyes are visible. This is not always true, in the faces with large pose variations, the inter-ocular and inter-pupil distance of near-profile faces is much shorter than that of frontal faces. To overcoming this drawback, we have been inspired from works in [20]–[27], where the error is normalized by face width in [20] or face size in [27] (computed as the mean value of face height and width). In this work, the normalized mean error (NME) has been calculated based on (1), the euclidean distance between each detected feature point on the face and its corresponding ground truth annotation, normalized by the width of the face bounding box. Although, NME can be considered a good evaluation metric, it is very sensitive to outliers. Failure rate (%); we define the failure rate as the proportion of cases where the normalized errors are larger than 10%.

$$NME = \frac{1}{N}\sum_{i=0}^{N-1}\left(\frac{1}{K}\sum_{j=0}^{K-1}\left(\frac{\sqrt{(x_{ij}-x'_{ij})^2+(y_{ij}-y'_{ij})^2}}{l_i^{facewidth}}\right)\right) \quad (1)$$

Where $\left(x_{ij}, y_{ij}\right)$ and $\left(x'_{ij}, y'_{ij}\right)$are the ground truth and the detected position, $l_i^{facewidth}$is the width of the bounding box returned by haar-like face detector [23], K is the number of landmarks, 5 in our case and N is the number of detected faces.

2.2. Datasets

The datasets used in this work all contain uncontrolled images of faces in the wild: in indoor and outdoor environments, under varying illuminations, in the presence of occlusions, with different poses, and captured using cameras of different qualities. The first one is the labeled face parts in-the-wild dataset

(LFPW) [25], which contains 1,432 face images from the web and is split into 1,132 training images and 300 test images. Some of image links are no longer valid. We downloaded only 811 images for training and 224 images for testing. Each image is annotated with 68 landmarks. We split 811 training images to 700 for training set and 111 for validating set. The second one is the helen facial feature dataset (Helen) [28], another challenging database that contains a total of 2,330 images, each image is annotated with 194 landmarks. As suggested by the authors, we split 2,000 training images to 1,200 for training set and 800 for validating set and the rest 330 images for the testing set. The helen dataset does not provide the eye center. We use mean value of points around the eye as pupil center.

2.3. Deep CNN parameters

In the literature, the appropriate number of epochs depends on the complexity of the dataset used. A good rule of thumb is to start with a value that is three times the number of columns in your data. In this work, we started with 50 epochs and we increased the number to 12,000 epochs in steps of 50 epochs, as shown in Figure 3. The NME value of all five key points decreases as the number of epochs increases. The error percentage decreases from 10% at 50 epochs to less than 4% at 10,000 epochs and for epochs beyond 10,000, the error remains close to 3.5%. In this experimental set, all tests are made on DCNN (F) and we resized all cropped face into 49 pixels for width and height. However, at each stage of the experimental process, we have been training the network from zero to specific number of epochs presented in x-axis of Figure 3. The error function used can be represented as (2).

$$Error = \frac{1}{k}\sum_{i=1}^{k}(y_i - {y'}_i)^2 \tag{2}$$

Where $y = \{y_1, y_2, \ldots, y_k\}$ are the ground truth position of key points and $y' = \{{y'}_1, {y'}_2, \ldots, {y'}_k\}$are the estimated position by our model, (k) number of key points, in our case $(k = 5)$.

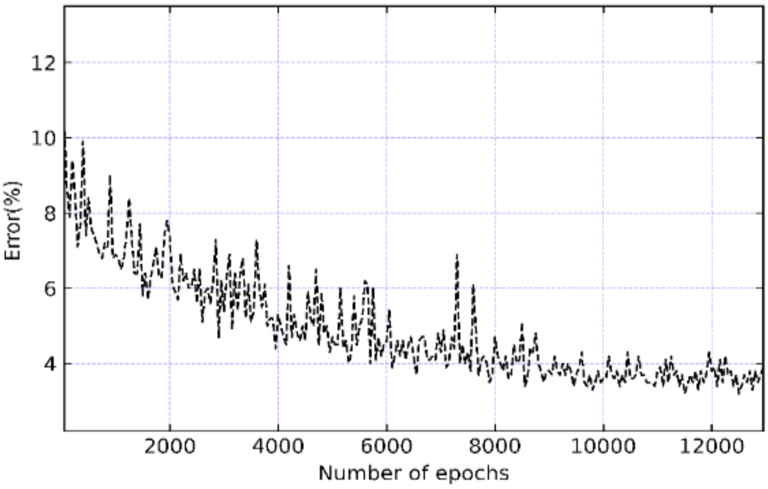


Figure 3. The average error of five key points goes down from 10% in 1,000 epochs to achieve less than 4% for epochs number more than 10,000. The average error used in this test is calculate by (2)

The input size is an important parameter of DCNN in general. each model in the state-of-the-art uses a specific size. For example, Sun *et al.* [20], the input size of the face bounding box has been resized to 39×39, in [10], the size of the cropped face has been resized to 256×256 and in [29], we have an input size of 96×96. In this work, we have traced the plot in Figure 4 that shows the variation of learning time of the DCNN (F) model and NME (%) for a range of input sizes from 40×40 to 140×140. As we can see, for small input sizes (between 40 and 45 pixels) the model takes 90 seconds to learn and NME (%) exceeds 22%. However, for larger input sizes (more than 50 pixels), the model's learning time starts to increase and takes more than 150 seconds. Afterwards, NME (%) began to decrease to less than 10%. We have determined a specific value (threshold) that minimizes both the learning time and NME (%). The optimal input size for the face bounding box is 49 pixels. In this experimental set, we fixed the number of epochs at 10,000. Figure 4 is curve in dashed line show the variations of our model learning time (second) for inputs sizes of face bounding box in pixels and the curve in simple line show the variations of our model NME (%) for the same the range of inputs sizes. The tests have been made on LFPW dataset and NME (%) have been calculate based on the difference between the predicted and ground truth position of each key points (eyes centers, nose tip and mouth corners) normalized by face width.

In the two previous sections, we have been specified the number of epochs and the input size of the face bounding box as the first two parameters of our model. Therefore, we have been used the grid search algorithm [24], implemented in the keras framework to determine the values of other hyperparameters such as the activation function, batch size, convolution layer filter size, max pooling layer filter size, stride, kernel padding types, and the optimizer type see Table 1. As for the optimizers, we used stochastic gradient descent (SGD) with a learning rate of 0.01, a decay rate of 1e-6, and a momentum of 0.9.

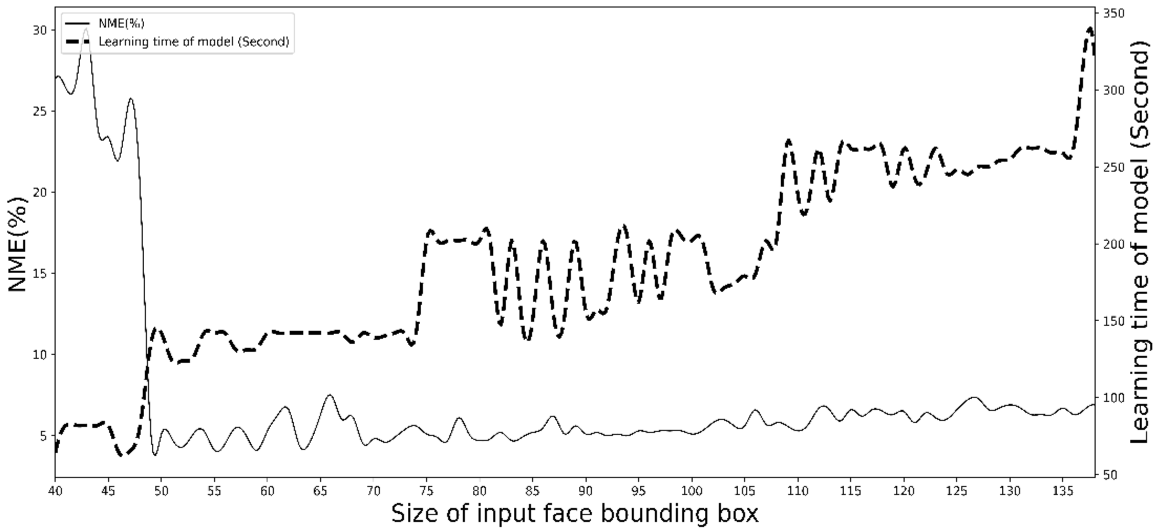


Figure 4. The variation of learning time of the DCNN (F) model and NME (%)

Table 1. Details of search space for each parameter in grid search algorithm [24]. We have been inspired from work in [30] to choose the search space of batch size

Activation functions	Batch size	Optimizers	Weight initializers
rectified linear unit (ReLu)	16	SGD	Random normal
Sigmoid	28	RMSprop	Random uniform
Softmax	32	Adam	Truncated normal
Softplus	64	AdamW	Zeros
Softsign	128	Adadelta	Ones
Tanh	256	Adagrad	Glorot normal
Exponential		Adamax	Glorot uniform
Leaky_relu		Adafactor	He normal
Relu6		Nadam	He uniforms
Silu		Ftrl	Orthogonal initializer
Gelu			Constant
Hard_sigmoid			
Linear			
Mish			
Log_softmax			

2.4. Network architecture of DCNNs

In the experimental process, for each database, we crop the faces from original images using the haar-like detector [23] and resize the cropped face images to 49×49×3 and we change image format from colors to grayscale images (49×49×1). Then, the three DCNNs in our model are trained using the keras framework with SGD as optimizer, the mini-batch size is set to 28, the initial learning rate is set to 0.01, decay is set to 1e-6 and momentum is set to 0.9 and we have adopted "he normal" as the weight initialization method. we use mean squared error (MSE) as our loss function with "tanh" as the activation function for all layers in network structure. The detail of three DCNNs (F, EN and MC) are described in Table 2.

Table 2. Details of DCNNs structure. The coefficient c is the number of key points DCNN (F); c=5 and c=3 for two DCNN (EN and MC)

Layer	Kernel	Stride	Padding	Activation function	Output shape
Conv1	4×4	1	1	Tanh	46×46×20
MaxPool1	2×2	0	1		23×23×20
Conv2	3×3	1	1	Tanh	21×21×40
MaxPool2	2×2	0	1		10×10×40
Conv3	3×3	1	1	Tanh	8×8×60
MaxPool3	2×2	0	1		4×4×60
Conv4	2×2	1	1	Tanh	3×3×80
Spacial dropout 2D (0.25)					3×3×80
Global average pooling 2D					80
FC				Tanh	120
Dropout (0.5)					120
FC				Tanh	2×c

3. RESULTS AND DISCUSSION

In this section, we compared our model with the state-of-the-art methods in [9]-[31] and the commercial software (luxand face SDK). The evaluation was conducted on two public databases, helen and LFPW. We utilized three types of evaluation metrics: failure rate (%), mean error normalized by face width NME (%), as defined in section 2.1, and mean error normalized by inter-ocular distance.

3.1. Results

To evaluate the performance of our model, we compared it with four approaches. In first time, we compare with two state-of-art methods (MFAERT [9], MTCNN [22]) and one commercial software (luxand face SDK). The methods in [9] and [22] are very competitive methods, which perform significantly better than their contemporaries. In the evaluation process, we use the evaluation metrics described in section 2.1, on the two test datasets described in section 2.2. The results are summarized in Table 3, Table 4, and Figure 5.

The graphs in Figure 5(a) to 5(d) shows that our method consistently outperforms the MTCNN [22] and luxand face SDK by a large margin in five key points (N, LE, RE, LM, and RM), in terms of average error and failure rate with number of levels less than MTCNN [22] (#Levels=3) see Table 5). In comparison to MFAERT [9], our model performed similarly, in four key points (N, LE, RE, and LM) and achieved better results for the RM key point with number of levels less than MFAERT [9] (#Levels=10) see Table 5). Furthermore, to compared our approach with deeper structure like the work in [31], we have executed the same experimental protocol on helen dataset with NME normalized by inter-ocular distance instead of face width. Figures 5(a) and 5(b) show the comparison of average error (%). Figures 5(c) and 5(d) show the comparison of failure rate (%). The evaluation has been made with the state-of-the art methods (MTCNN [22], MFAERT [9]), and commercial system luxand face SDK for five key points localizations on LFPW and Helen datasets.

Table 3. The performance of our model compared with the methods MTCNN [22], MFAERT [9], and commercial system luxand face SDK on LFPW dataset

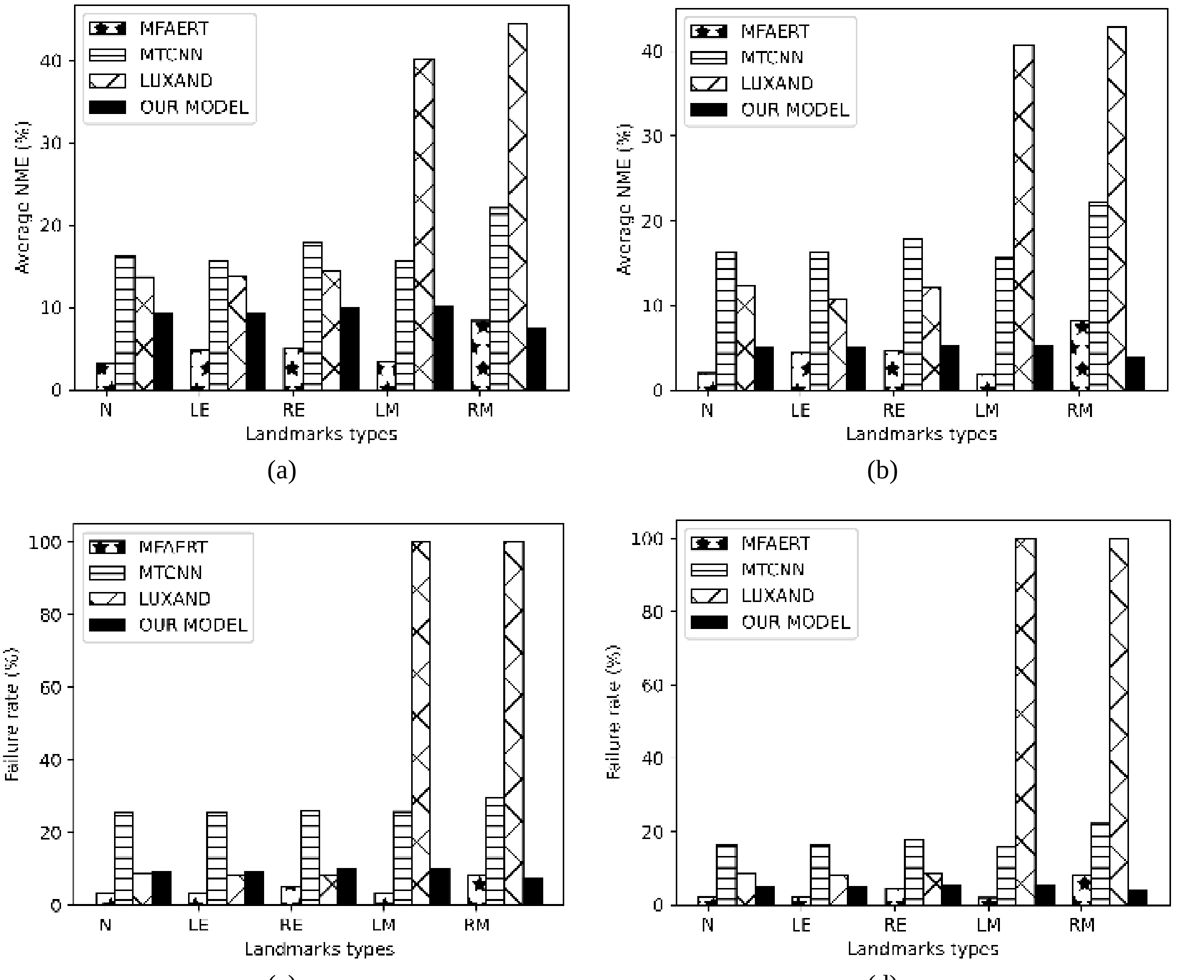
Method	Average NME/Failure rate (%)				
	N	LE	RE	LM	RM
MFAERT [9]	3.2/3.2	4.9/3.2	5.0/5.0	3.4/3.4	8.4/8.4
MTCNN [22]	16.2/25.6	15.6/25.6	17.9/26.1	15.7/25.8	22.1/29.5
Luxand face SDK	13.7/8.6	13.8/8.2	14.4/8.2	40.1/100	44.5/100
Our model	9.4/9.4	9.4/9.4	10.0/10.0	10.1/10.1	7.5/7.5

Table 4. The performance of our model compared with the methods MTCNN [22], MFAERT [9], and commercial system luxand face SDK on Helen dataset

Method	Average NME/Failure rate (%)				
	N	LE	RE	LM	RM
MFAERT [9]	2.0/2.0	4.5/2.0	4.6/4.6	1.9/1.9	8.2/8.2
MTCNN [22]	16.2/16.2	16.2/16.2	17.9/17.9	15.7/15.7	22.1/22.1
Luxand face SDK	12.3/8.7	10.7/8.1	12.2/8.4	40.7/100	42.9/100
Our model	5.0/5.0	5.0/5.0	5.2/5.2	5.2/5.2	3.8/3.8

Table 5. The performance of our model compared with MTCNN [22], MFAERT [9], and STAR loss [31] on helen dataset. The mean error is normalized by inter-ocular distance

Method	#levels	#parameters	Average NME (%)
MTCNN [22]	3	N/A	17.62
MFAERT [9]	10	25k	4.24
STAR loss [31]	4	14M	4.32
Our model	1	60k	4.20



(a)

(b)

(c)

(d)

Figure 5. Method consistently outperforms the MTCNN and luxand face SDK by a large margin in five key points (N, LE, RE, LM, and RM): (a) average error on LFPW dataset, (b) average error on helen dataset, (c) failure rate on LFPW dataset, and (d) failure rate on helen dataset

3.2. DISCUSSION

In the evaluation set made on helen dataset shown in Table 5, our model shows better results compared with the methods in MTCNN [22], MFAERT [9], and STAR loss [31] in term of levels number (#Levels) and accuracy (average NME (%)). The mean error has been normalized by inter-ocular distance that commonly used in literatures and it has been calculated by mean value of five key points (N, LE, RE, LM, RM). In the same manner, the Tables 3 and 4 shown that our model gives better than the approaches in MTCNN [22] and luxand face SDK, for each specific key points in the term of failure rate (%) and average NME (%) on two public dataset helen and LFPW. However, the approach in MFAERT [9] shown better results than our model see Tables 3 and 4 but this method has 10 levels compared with the approach proposed in this with one level see Table 5. The majority of works proposed in the literature has a deeper structure as backbone or a cascade level of CNNs, which need more computational resources for training and implementation the model. Even though, we propose a model with one level of three DCNNs (F, EN, and MC), which we can trained and implemented in the small devices. Moreover, the experimental set summarized in Figure 5 and Tables 3 to 5 show that our model shows better results (4.20% average NME see Table 5) compared with state-of-the-art methods. In addition, the model has one level in cascade process of detection and small number of parameters 60k see Table 5 compare with STAR loss [31] which has a number of parameters equal to 14M. In other words, the proposed model has various benefits regarding resource optimization and detection accuracy compared with the others methods make this method better chois to deal with on small devices like smart phone, onboard car system and existing medical machines.

The accurate estimation of facial key points can help in daily life. Specifically, car accidents frequently occur as a result of drivers nodding off while driving, and the emergency braking features in smart vehicles are not dependable. Vigilance for signs of driver fatigue can help prevent accidents. In-vehicle cameras have the capability to observe the driver and identify signs of fatigue or distraction through a computer vision model. The model will trigger an alarm when it discerns a lapse in the driver's concentration. To effectively replicate one face on another, it is essential to obtain an estimate of the locations of landmark features on both faces for proper alignment. Moreover, facial animation algorithms leverage facial landmark detection to generate animated characters from images with specifically marked facial landmark. These algorithms find application in generating subsequent frames for use in 3D movies and games. In addition, facial landmarks detection has a vital application in relation to medical applications, Currently, there are a considerable number of individuals suffering from paralysis. These individuals have the capability to transmit a request signal to a system by using their facial expressions. Kowalski *et al.* [3], provides a device that can handle individual emotions based on accurate detection of facial landmark positions. On the whole, the importance of facial landmark detection can be reached by avoiding deeper structures and turning to small models with minimum levels in the detection process, as long as they can give good results. The opportunity of embedded devices on small devices can attract more attention from the research community to this kind of model.

4. CONCLUSION

In this work, we introduce several approaches to address facial landmarks detection. To illustrate, CNN-based approaches, deeper structure like MobileNet and ResNet and cascade-based approaches, which can show better results in term of accuracy. These models required more computational resources (high performance GPUs). In fact, the previous methods are not suitable to be deployed on small devices. On the other hand, we present a model consists of three DCNNs (F, NE, and MC) combined to achieve better accuracy as the final output of the model. We also present a protocol to improve the hyperparameters of our architecture. Through a set of experiments using the LFPW and helen datasets, we demonstrate that our model can estimate the position of the five key points more accurately than the MTCNN method and commercial software (luxand face SDK) and performed similarly to the MEART model in the term of accuracy (4.20% of average NME of all key points) and failure rate (3.8% on RM key point) in addition to efficient, our model use one level in the cascade process of detection and a small number of parameters (60k) compared with state-of-the-art methods, which make our approach suitable to be deployed on small devices like smart phone. In future work, we will extend our model's detection from five key points to 68 key points and we will explore facial expression analysis and recognition further.

REFERENCES

[1] A. Bätz *et al.*, "Facial landmarks detection : a brief chronological survey & practical implementation," no. August, pp. 0–16, 2021, doi: 10.13140/RG.2.2.36199.98722.

[2] A. H. Ahmad *et al.*, "Real time face recognition of video surveillance system using haar cascade classifier," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 21, no. 3, pp. 1389–1399, 2021, doi: 10.11591/ijeecs.v21.i3.pp1389-1399.

[3] R. Praditsangthong and P. Bhattarakosol, "Facial action coding-based facial sub-structures for anxiety emotion classification," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 30, no. 1, pp. 208–218, Apr. 2023, doi: 10.11591/ijeecs.v30.i1.pp208-218.

[4] T. F. Cootes, C. J. Taylor, and A. Lanitis, "Active shape models: evaluation of a multi-resolution method for improving image search," in *Procedings of the British Machine Vision Conference 1994*, 2013, pp. 32.1-32.10, doi: 10.5244/c.8.32.

[5] G. Tzimiropoulos and M. Pantic, "Optimization problems for fast AAM fitting in-the-wild," in *Proceedings of the IEEE International Conference on Computer Vision*, Dec. 2013, pp. 593–600, doi: 10.1109/ICCV.2013.79.

[6] P. Martins, R. Caseiro, and J. Batista, "Generative face alignment through 2.5D active appearance models," *Computer Vision and Image Understanding*, vol. 117, no. 3, pp. 250–268, Mar. 2013, doi: 10.1016/j.cviu.2012.11.010.

[7] J. M. Saragih, S. Lucey, and J. F. Cohn, "Face alignment through subspace constrained mean-shifts," in *Proceedings of the IEEE International Conference on Computer Vision*, Sep. 2009, pp. 1034–1041, doi: 10.1109/ICCV.2009.5459377.

[8] P. Dollár, P. Welinder, and P. Perona, "Cascaded pose regression," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Jun. 2010, pp. 1078–1085, doi: 10.1109/CVPR.2010.5540094.

[9] V. Kazemi and J. Sullivan, "One millisecond face alignment with an ensemble of regression trees," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Jun. 2014, pp. 1867–1874, doi: 10.1109/CVPR.2014.241.

[10] S. Qian, K. Sun, W. Wu, C. Qian, and J. Jia, "Aggregation via separation: boosting facial landmark detector with semi-supervised style translation," *Proceedings of the IEEE International Conference on Computer Vision*, vol. 2019-October, pp. 10152–10162, 2019, doi: 10.1109/ICCV.2019.01025.

[11] Y. Ge, X. Ren, C. Peng, and X. Wang, "Extended robust cascaded pose regression for face alignment," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 9967 LNCS, 2016, pp. 50–58.

[12] M. Valstar, B. Martinez, X. Binefa, and M. Pantic, "Facial point detection using boosted regression and graph models," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Jun. 2010, pp. 2729–2736, doi: 10.1109/CVPR.2010.5539996.

[13] M. Kowalski, J. Naruniec, and T. Trzcinski, "Deep alignment network: a convolutional neural network for robust face alignment," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, vol. 2017-July, pp. 2034–2043, 2017, doi: 10.1109/CVPRW.2017.254.

[14] X. Dong, Y. Yan, W. Ouyang, and Y. Yang, "Style aggregated network for facial landmark detection," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 379–388, 2018, doi: 10.1109/CVPR.2018.00047.

[15] H. Jin, S. Liao, and L. Shao, "Pixel-in-Pixel net: towards efficient facial landmark detection in the wild," *International Journal of Computer Vision*, vol. 129, no. 12, pp. 3174–3194, 2021, doi: 10.1007/s11263-021-01521-4.

[16] A. G. Howard *et al.*, "MobileNets: efficient convolutional neural networks for mobile vision applications." 2017, doi: 10.48550/arXiv.1704.04861.

[17] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Jun. 2016, vol. 2016-Decem, pp. 770–778, doi: 10.1109/CVPR.2016.90.

[18] O. Russakovsky *et al.*, "Imagenet large scale visual recognition challenge 2010," *ArXiv Preprint*, 2010, [Online]. Available: https://arxiv.org/abs/1409.0575.

[19] X. Guo *et al.*, "PFLD: a practical facial landmark detector," *ArXiv Preprint*, 2019, [Online]. Available: http://arxiv.org/abs/1902.10859.

[20] Y. Sun, X. Wang, and X. Tang, "Deep convolutional network cascade for facial point detection," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 3476–3483, 2013, doi: 10.1109/CVPR.2013.446.

[21] N. Agarwal, A. Krohn-Grimberghe, and R. Vyas, "Facial key points detection using deep convolutional neural network - NaimishNet," *ArXiv Preprint*, 2017, [Online]. Available: http://arxiv.org/abs/1710.00977.

[22] J. Xiang and G. Zhu, "Joint face detection and facial expression recognition with MTCNN," in *Proceedings - 2017 4th International Conference on Information Science and Control Engineering, ICISCE 2017*, Jul. 2017, pp. 424–427, doi: 10.1109/ICISCE.2017.95.

[23] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2001, vol. 1, pp. I-511-I–518, doi: 10.1109/cvpr.2001.990517.

[24] P. Liashchynskyi and P. Liashchynskyi, "Grid search, random search, genetic algorithm: a big comparison for NAS," *ArXiv Preprint*, 2019, [Online]. Available: http://arxiv.org/abs/1912.06059.

[25] P. N. Belhumeur, D. W. Jacobs, D. J. Kriegman, and N. Kumar, "Localizing parts of faces using a consensus of exemplars," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 12, pp. 2930–2940, 2013, doi: 10.1109/TPAMI.2013.23.

[26] R. Ranjan, V. M. Patel, and R. Chellappa, "HyperFace: a deep multi-task learning framework for face detection, landmark localization, pose estimation, and gender recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 1, pp. 121–135, Jan. 2019, doi: 10.1109/TPAMI.2017.2781233.

[27] X. Zhu and D. Ramanan, "Face detection, pose estimation, and landmark localization in the wild," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Jun. 2012, pp. 2879–2886, doi: 10.1109/CVPR.2012.6248014.

[28] V. Le, J. Brandt, Z. Lin, L. Bourdev, and T. S. Huang, "Interactive facial feature localization," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 7574 LNCS, no. PART 3, 2012, pp. 679–692.

[29] P. Dileep, B. K. Bolla, and E. Sabeesh, "Revisiting facial key point detection-an efficient approach using deep neural networks," *Lecture Notes in Electrical Engineering*, vol. 1053 LNEE, pp. 511–525, 2024, doi: 10.1007/978-981-99-3481-2_39.

[30] I. Kandel and M. Castelli, "The effect of batch size on the generalizability of the convolutional neural networks on a histopathology dataset," *ICT Express*, vol. 6, no. 4, pp. 312–315, Dec. 2020, doi: 10.1016/j.icte.2020.04.010.

[31] Z. Zhou, H. Li, H. Liu, N. Wang, G. Yu, and R. Ji, "STAR loss: reducing semantic ambiguity in facial landmark detection," *ArXiv Preprint*, pp. 15475–15484, 2023, doi: 10.1109/cvpr52729.2023.01485.

BIOGRAPHIES OF AUTHORS

Abdelaali Benaiss received his master's degree in Computer Science, Telecom and Image Processing in 2008 from the Faculty of Science Rabat, University Mohamed V Rabat. He is currently a Ph.D. degree student. His research activities are located in the area of face recognition and biometric systems; it deals with facial landmarks detection, facial expression, face verification and identification. He can be contacted at email: benaiss.abdelaali@gmail.com.

Rachid El Ayachi is a professor of higher education at the Faculty of Sciences and Techniques of Beni Mellal (Computer Science Department) since 2013, he obtained a Master’s degree in Computer Science, Telecom and Multimedia (ITM) in 2006 at the Faculty of Sciences of Rabat (Mohammed V University) and Ph.D. degree in computer science at the Faculty of Sciences and Techniques of Beni Mellal (Sultan Moulay Slimane University) in 2012. Currently, he is a member of the information processing and decision support laboratory (TIAD). His research focuses on image processing, pattern recognition, machine learning, and natural language processing (NLP). He can be contacted at email: rachid.elayachi@usms.ma.

Mohamed Biniz obtained his master’s degree in business intelligence in 2014 and completed his Ph.D. in computer science in 2018 at the Faculty of Science and Technology, University Sultan Moulay Sliman Beni-Mellal. Currently, he serves as a professor of computer science at the Polydisciplinary Faculty of the University Sultan Moulay Slimane Beni Mellal in Morocco. His research focuses on the field of semantic web engineering and deep learning, with a specific interest in studying the evolution of ontology, big data, natural language processing, machine learning, and dynamic programming. He can be contacted at email: mohamedbiniz@gmail.com.

Mustapha Oujaoura was born in Morocco. He received the Postgraduate degree in electrical engineering and space telecoms from Mohammadia Engineering School (EMI), Rabat, in 2009, and the Ph.D. degree in computer sciences from the Faculty of Science and Technology in Sultan Moulay Slimane University, Beni Mellal, on April 12, 2014. From 2009 to 2016, he’s temporarily an Adjunct Associate Professor and member of the Information Processing and Telecommunications Laboratory in the same institution. Since 2016, he moved as permanent Professor of computer sciences to the National Engineering School of Applied Sciences (ENSA) in Cadi Ayyad University, Marrakech. He’s an active member of Informatics, Mathematics and Communication Systems Laboratory in the same Engineering School. His main areas of research interest cover: pattern recognition, artificial intelligence, machine learning, data science, decision and information retrieval, computer vision, video analysis, and image understanding. He can be contacted at email: mustapha.oujaoura@uca.ac.ma.