Face Identification Based on Active Facial Patches Using Multi-Task Cascaded Convolutional Networks

Krishnaraj M. * and Jeberson Retna Raj R.

Department of Computer Science and Engineering, School of Computing, Sathyabama Institute of Science and Technology, Chennai, Tamil Nadu 600119, India
Email: monykrishnaraj@gmail.com (K.M.); jebersonretnarajr@gmail.com (J.R.R.R.)
*Corresponding author

***Abstract*—Face recognition technology is widely used for access control, security, identification, safeguarding, verification, timekeeping, and machine vision, etc. a new face identification algorithm referred to as Multi-Task Cascaded Convolutional Network (MTCCN) has emerged and has been widely used in high accuracy and efficiency in facial recognition, active facial patch identification framework face detection, selection of eyes, nose, lip, and eyebrow, identifying facial patches location and extraction of patches. This paper aims to discuss the recognition and identification of faces using layers of the Convolutional Neural Network (CNN). It is done to process camera frames as they appear and subsequent identification of the person. With three convolutional networks, MTCCN outperforms many face detection tests incredibly well, even though it maintains real-time performance. An active facial patch using MTCNN method is introduced for recognizing human faces in real time was developed, evaluated, and 97.62% of the time, the technique could recognize human faces correctly.**

***Keywords*—face identification, Multi-Task Cascaded Convolutional Network (MTCCN), active facial patches, convolutional networks, classification, region of interest**

I. Introduction

Face detection in images taken in uncontrolled environments is a challenging task. It has been demonstrated that Deep Convolutional Neural Network (DCNN) can significantly improve the Performance of detection; however, these techniques are not fully exploited information about the erection of the face. It is, therefore, impossible to detect faces whose rotation angles are more extreme. Biometric authentication methods that utilize face recognition are among the most popular because of their convenience and feasibility [1]. DCNN has improved computer vision performance to an unprecedented level by applying deep learning techniques. An algorithm for face recognition can be more accurate by building a more profound and complex network [2].

Information about static and dynamic images exchanged in two-stream networks can be combined. In contrast, partial images can be used for fatigue-related data to be focused on network inputs, which results in improved enactment [3]. Integrating static and dynamic information about images with two-stream networks is possible, while partial pictures of the face serve as web input. Fatigue-related information can be provided, thus improving the network performance [4]. Significant progress has been made in detecting facial landmarks using Convolutional Neural Network (CNN) in recent years, but partial occlusion and extreme head pose still pose challenges [5]. A face detection system is the infrastructure for facial recognition; additional topics include, but are not limited to, traffic surveillance, stereo videos, finding criminals among a large group in a terrorist incident, calibrated stereo images, and the alignment of face images by sensors. Face identification in photos captured in uncontrolled circumstances is a complex problem. DCNN has been shown to enhance detection performance significantly; nevertheless, these algorithms do not fully leverage information regarding the erection of the face [6]. There is a great deal of information conveyed through facial expressions rather than verbally. Various styles of expression of emotions have made for face alignment; a deep convolutional network with multi-task capabilities is planned, which is capable of achieving high Performance even under conditions with many pose variations and severe occlusions [7].

Many face-related applications, including 3D reconstruction and face recognition, are predicated on detecting and localizing faces and landmarks. There currently needs to be more methods that address both problems simultaneously [8]. The recognition of facial expressions is one of the most challenging problems in machine learning. The field of facial expression recognition has seen substantial research; however, increased accuracy is needed, especially in uncontrolled conditions [9]. As a recent development, face recognition has become an integral part of social cognition and has been used in various requests, pedestrian tracking, and surveillance systems. There has been significant progress in facial recognition using DCNN [10].

Manuscript received February 27, 2023; revised June 28, 2023; accepted July 31, 2023; published January 18, 2024.

doi: 10.12720/jait.15.1.118-126

Several latest studies have utilized Video data of the face to measure vital signs, assess pain caused by facial expressions, detect jaundice, and recognize faces. As a Region of Interest (ROI), the face of the patient must be accurately defined [11]. There have been extensive studies conducted on the location of eye landmarks. Yet, the accuracy of this recognition is easily affected by lighting and changes in eye state when one is exposed, and another is closed [12]. Several facial recognition and face analysis systems rely on face detection as the first step. Face detection was initially achieved using classifiers built on hand-crafted features obtained from local regions of an image, such as Haar cascades and Histograms of Oriented Gradients (HOG) [13].

Detecting and aligning faces is challenging because of variations in image angles, background lighting conditions, and intervening objects [14]. The high computational complexity of such an approach makes it challenging to implement in a low-end edge AI system [15]. Face recognition has recently been an essential aspect of social cognition and has been employed in various requests, pedestrian monitoring, and surveillance systems. There have been previous attempts to discriminate facial features based on age-related and age-invariant components. However, this has resulted in a loss of information regarding facial identity.

Overall, the research contributions highlighted in the text revolve around the challenges faced in face detection and recognition in uncontrolled environments. The proposed use of deep learning techniques, integration of static and dynamic information, and multi-task learning methods aim to improve accuracy and performance in real-world applications. However, the text also raises awareness of the limitations and challenges that need to be addressed for further advancements in the domain.

II. Literature Review

Biometrics and security applications of face recognition are used in various ways. In a constrained environment, most present-day face recognition techniques demonstrate satisfactory results. There are, however, many problems with these techniques when they are applied in real-world situations [16]. Face identification based on active facial patches using Multi-Task Cascaded Convolutional Networks (MTCNN) is a widespread facial recognition and analysis approach. While this method offers several advantages, it also comes with challenges. Here are some of the challenges associated with face identification based on active facial patches using MTCNN: Active facial patches focus only on specific regions of the face, such as the eyes, nose, and mouth. While this approach can be effective in specific scenarios, it may result in limited facial information being available for identification. This can reduce accuracy, especially when dealing with low-quality or partially occluded face images.

Using the Multi-Task Collaboration Network (MTCNet) to detect rotation-invariant faces, a new method for detecting faces that improves the detection of facial landmarks accuracy was developed. MTCNet is to see rotation-invariant faces based on facial landmarks in integration with face alignment to improve detection performance [17]. Regarding localization accuracy, MTCNN relies on face detection and alignment to identify the active facial patches. However, accurately localizing these patches can be challenging, mainly when dealing with variations in pose, illumination, and facial expressions. Errors in patch localization can negatively impact the overall face identification performance. Different individuals may have similar facial appearances within the active patches, making it difficult to distinguish them accurately. For instance, people with similar eye colour, lip shape, or nose structure can lead to confusion during identification.

Considering a single camera as an example, this paper offered an efficient and unified Algorithm for facial recognition using neural networks; the entire face recognition process involves four steps: detection of face, detection of vivo, detection of a critical point, and verification of face [18]. A combination of these four algorithms is used in the face recognition process. The article presents a new Multi-feature Fusion and Decomposition (MFD) outline for an age-invariant face recognition system with more sophisticated features of robustness learned and minimized variations [19].

Herein, a novel method is presented for estimating the pose of the head from a monocular camera. The suggested algorithm in [20] uses a small grayscale image to train a Deep Neural Network (DNN) that can perform multi-task learning. Ref. [21] contributes to the field of driver fatigue detection by combining with a multi-facial feature; images can be combined in two-stream networks in both a static and dynamic manner. A simple noise-based data augmentation scheme was used to assess the efficacy of Deep Learning (DL) methods for Disguise Invariant Face Recognition (DIFR) [22]. An image is detected using Viola Jones’ face detector and classified using a pre-trained CNN tuned specifically for DIFR.

A CNN is offered in [23] for robustly and efficiently detecting faces from blurry and noisy images. MTCNN consists of multiple cascaded neural networks, including a face detection network, landmark localization network, and patch classification network. The computational complexity of these networks can be high, especially when dealing with real-time or large-scale face identification applications. Efficient implementation and optimization techniques are required to address this challenge. The paper presents a Cascaded Structure-Learning Network (CSLN) by considering the structure of facial landmarks. To enhance the precision of detecting 2D facial landmarks, adversarial training is used in [24]. Some recent research has used video data of the face to measure vital signs, analyse pain sensations produced by facial expressions, diagnose jaundice, and recognize faces. Based on the geometry and appearance features, Facial Expression Recognition (FER) systems attempt to perceive and acknowledge an expression of emotion [25].

In face identification datasets, the number of positive samples (faces of the same person) is often significantly smaller than the number of negative pieces (faces of different people). This data imbalance can affect the training process of the network, leading to biased performance. Appropriate techniques, such as data augmentation and sample weighting, should be employed to mitigate this challenge. MTCNN-based face identification systems are trained on a specific dataset, and their performance may degrade when applied to unseen faces or different populations. This challenge arises due to variations in facial appearances across demographics, ethnicities, ages, and other factors. Ensuring the system's generalization capability is crucial for its real-world deployment.

The distribution of facial points was guided by estimating the pose as an auxiliary task instead of facial alignment alone [26]. Through the implementation of multi-task learning as opposed to traditional single-task education, the detection of the Action Unit (AU) is enhanced by analysing additional facial attributes [27]. Face recognition has recently been a critical aspect of social cognition and has been employed in various requests, pedestrian monitoring, and surveillance systems. Using a coupled encoder-decoder network, recommended a method for jointly detecting faces and locating critical facial points in [28, 29]. It describes a robust framework for recognizing faces and tracking them in unconstrained settings based on lightweight CNNs [30].

The illustrated approach in [31] aims to resolve these issues by evaluating Neo-natal Intensive Care Unit (NICU) face detection technology settings [32]. NICU face models demonstrate fine-tuning can enhance the performance of the robustness of new face detectors in complex NICU environments [33]. Umamageswari *et al.* [34] comprehensively overviews some of the most effective DL-based face detection methods, categorizing them into a few major categories [35]. Addressing these challenges requires ongoing research and development in face identification. Advancements in network architectures, training strategies, data augmentation techniques, and dataset diversity can help improve the accuracy, robustness, and real-world applicability of face identification systems based on active facial patches using MTCNN.

In conclusion, while the text highlights the importance of deep learning techniques like DCNN and MTCNN for face detection and recognition, it also points out several limitations and challenges that existing methods face. These limitations include handling facial rotation, partial occlusion, extreme head pose, data imbalance, computational complexity, and generalization across diverse populations. Addressing these limitations is essential for improving the performance of face detection and recognition systems in real-world, uncontrolled environments.

III. Materials and Methods

Local variations in active facial patches' appearance are usually symptoms of face identification. In a facial image, however, local functional areas are challenging to locate automatically. The regional appearance of a patch is represented by Local Binary Patterns (LBP) features, which indicate the patch's local appearance. It has been proven that these features are compelling descriptors when recognizing expressions and verifying faces. As shown in Fig. 1, the planned method is presented in a high-level manner. According to the research, detecting facial landmarks accurately and extracting appearance features improves the recognition of facial expressions. In this case, the first step is to locate the face before the landmarks can be detected. This approach uses a learning-free method to see the eyes and nose around an ROI.

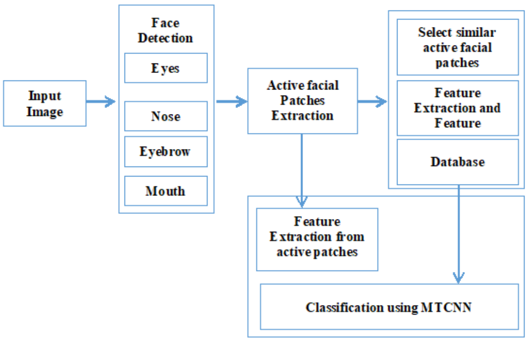
Input Image
Face Detection
Eyes
Nose
Eyebrow
Mouth
Active facial Patches Extraction
Select similar active facial patches
Feature Extraction and Feature
Database
Feature Extraction from active patches
Classification using MTCNN

Fig. 1. Overall systematic diagram for the active facial patches using MTCNN.

Several facial patches appear to be expected in the emergence of all fundamental facial expressions, while others seem to be specific to a particular word. Based on the study results, Active patches can be found under the eyes, near the eyebrows, and on the corners of the mouth and nose, as shown in Fig. 2. When analyzing a face image, the facial components must be located first, then the patches around them must be extracted.

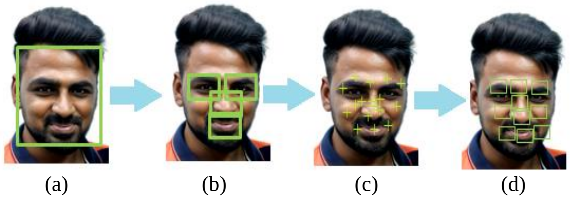

Fig. 2. Active facial patch identification framework. (a) Face detection, (b) Selection of eyes, nose, lip, and eyebrow, (c) Identifying facial patches location, and (d) Extraction of patches.

The experiment was conducted with a *p*-value of 64 and an image size, as illustrated in Fig. 3(a). As shown in Fig. 3(b), the uniform LBP features are extracted for each patch using the LBP operator, which is then converted to a histogram of m dimensions.

Through the analysis of local patches, common patches are learned that can be used to recognize all expressions. To further enhance the performance, certain patches are explored for each expression.

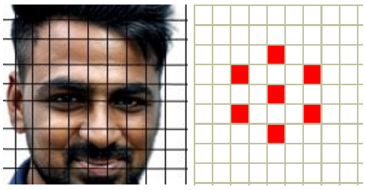

Fig. 3. Active patch extraction using the LBP operator. (a) 64 Patches in a Cropped Facial Image, (b) Feature of LBP.

A. Face Identification Based on Active Facial Patches

It is similar to learning the discriminative patches shared by all expressions. Multi-task Sparse Learning (MTSL) can teach representations shared between various tasks; MTSL can be used to solve the problem. *T discriminative patch learners are defined* separately for each T facial expression. It is assumed that if each image consists of patches, An LBP-based histogram can represent this data with three dimensions. The objective is to select standard patches rather than features individually. It is assumed that a set of consecutive features represents a patch, and if there are not many patches in common, it is possible to take a group sparsity prior. As a result, model the problem as the in-depth MTSL problem by substituting a sparse constraint at the patch level for the regularization term in Eq. (1):

$$arg\min \sum_{T=1}^{t} \frac{1}{Nt} \sum_{i=1}^{Nt} J^t(w^t, x_i^t, y_i^t) + \lambda \sum_{j=1}^{p} \|wG_j\| \quad (1)$$

where wG_j is a submatrix, G_j represents the j^{th} patch. A specific expression is considered a positive sample in each task, whereas other expressions are considered negative samples. In most patches, this regularization term results in a zero-representation coefficient for each feature, while the remaining non-zero patches indicate that all expressions share an important representation. To calculate the cost function of J^t, define it as a logistic loss function as follows:

$$J^t(w^t, x_i^t, y_i^t) = \ln(1 + \exp(-y_i^t, x_i^t . w^t)) \quad (2)$$

The patch-based multi-task sparse learning problem is solved, and the coefficients of the regression are determined heuristically by representations by steps 5–9 in Algorithm 1. A summary of the detailed steps involved in this process can be found in Algorithm 1.

Algorithm 1. Learning Active Facial Patches

```
Step 1: Input: Trained data {(x, y ←Task index)}
Step 2: Initialize weights equally in Tasks (($T^w$)
Step 3: for size S= 1, 2,…, n do
Step 4: $T_S^w = n \left(\frac{1}{1+(x,y \leftarrow \text{Task index})}\right)$
Step 5: if $\|T_S^w\| \geq \frac{n}{2}$ thenSt
Step 6: set $T_S^w = \left(\frac{1}{1-(x,y \leftarrow \text{Task index})}\right)$
Step 7: else
Step 8: set $T_S^w = 0$
Step 9: end if
Step 10: end for
Step 11: Image Normalization:$T_S^w = \frac{T^w}{\|x+y \leftarrow \text{Task index}\|}$
Step 12: $T_S^w = \sum_{S=1}^{n} \frac{T^w}{\left(x+y\,(T_S^w)\right)}$
where $\left(x + y\,(T_S^w)\right)$→weight for active facial patch
Step 13: Output: Active facial patches.
```

Face recognition utilizes a user scene in which the user is not far away from the imaging device, and the image focuses primarily on the face area. As a result, it automatically detects facial points throughout the image to identify a candidate's facial area, thereby determining whether the candidate's site contains a face.

As a result, the amount of structure and calculation caused by face detection is reduced, resulting in a more focused and concise network. The algorithm can simplify detecting facial points by combining the model and the final target image. In the case of multiple individuals in the picture, the primary user should be located in the center of the image. At this point, the model will perform face point identification, facial recognition, face verification, etc. In the center of the image, on the face closest to the camera. Active facial patches functions include the following:

$$Active\ facial\ patches\ loss = \alpha \frac{1}{N} \sum_{i=0}^{N} \sum_{I=0}^{5} \sqrt{(x_{il} - X_{il})^2 - (y_{il} - Y_{il})^2} \quad (3)$$

$$Face\ Detect\ Loss = \gamma \frac{1}{N} \sum_{i=0}^{N} \sqrt{(c_i - C_i)^2} \quad (4)$$

A batch of samples in training consists of N samples, x_{il} with the shared projection being that sample's basic facial point l, Y_{il} is the true coordinates of that point, *a* relates to Active facial patches loss, and *g* signifies the weight of Face Detect Loss.

B. Active Facial Patches Extraction

Based upon the position of active facial muscles, local patches were derived based on the pattern of the active facial muscles. As part of the analysis, examined the appearance of facial regions that exhibit considerable variation over time. For example, disgust expressions show prominent wrinkles on the upper nose region, while other words do not. As illustrated in Fig. 4, active facial patches represent the experimental design derived from the observations.

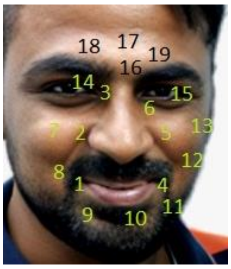

Fig. 4. Position of facial patches.

The position of the patches on the face image is not very fixed. Instead, they are located according to a facial landmark's position. A ninth of the width of the face was used as each facial patch was the same size. From here on, we will refer to patches by their number. 16 is located in both centres of the eyes, and Patch 17 was used above 16. 3 and 6 were positioned midpoints between the eye and the nose. Just below the watches were 14 and 15. 2, 7, and 8 were combined, and nose position nine was located below 1.

C. Detection of Accurate Facial Points

To find accurate facial points, the suggested framework uses active facial points. In the clipping local feature map, the corresponding receptive field contains the information the model attempts to obtain regarding the target area.

Furthermore, a more precise scope has been defined. This results in a network that is more focused on a specific site of the face and improves the efficiency of the face detection task. Detecting facial points accurately has the following objective function:

$$Accurate\ loss = \beta \frac{1}{N} \sum_{i}^{N} \sum_{l}^{4i} \sqrt{(m_{il} - M_{il})^2 - (n_{il} - N_{il})^2} \quad (5)$$

There are N samples in a batch of samples in training, (m_{il}, n_{il}) corresponds to specifiesthe location of the facial point i in the sample, (M_{il}, N_{il}) represents the coordinates the sample i, and b is the intensity of accurate loss.

D. Process of MTCNN

Constructing the image. The image can be resized at different scales and build a Pyramid of an image, the face detection will be able to fit faces of dissimilar sizes.

As shown in Fig. 5, the planned approach follows an overall pipeline. For this cascaded framework to function, first resize the image to various sizes to create a pyramid of ideas, which will be used as inputs in the following three stages:

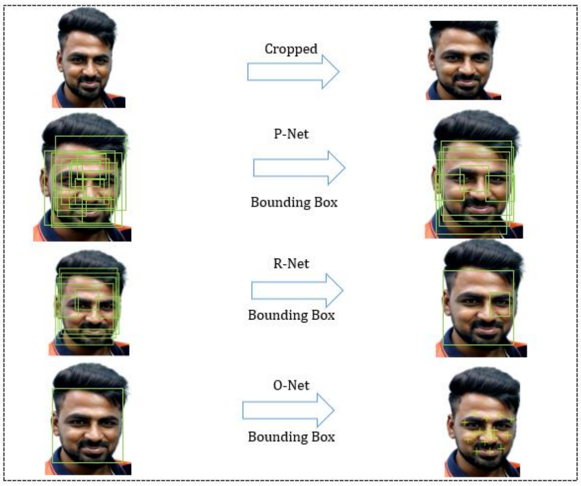

Fig. 5. Overall pipeline of face identification with MTCNN.

P-net: The Proposal Network (P-net) is an open network with a shallow depth and simple design of a fully connected network (FCN), which is used to propose face areas according to the following steps:

Step1: To determine the bounding box, FCN is used to extract the initial facial features;

Step 2: The classification of faces by a human is identified by placing feature layers divided into three convolution network layers.

Step 3: A Bounding-Box Regression and NMS filter most of the candidate windows, and then a locator for face features is used to locate the possible face locations, followed by a face area proposal.

R-net: A more complex convolutional network, the Refine Network, filters and predicts face windows based on P-network analysis. The objective is to obtain face area windows that are more credible and accurate. To accomplish this, the R-net adds the final convolution layer consisting of 128 FCNs and stores additional image information than the P-net, which holds one feature per integer. It removes Candidates' faces displayed in significant windows effects that aren't adequate to be retained. In addition to Bounding-Box Regression, R-net can likewise optimize the result using NMS.

O-net: In this case, the Output Network (O-net) is a moderately complex convolutional network with one additional layer compared to the R-net, which outputs the five final features by regressing the features of the face and managing the face area. Compared to R-net, O-net incorporates 256 FCNs at the end to preserve more image features. The intricate steps are completed, and the five facial features are displayed along with the coordinates for the top left and bottom right corners.

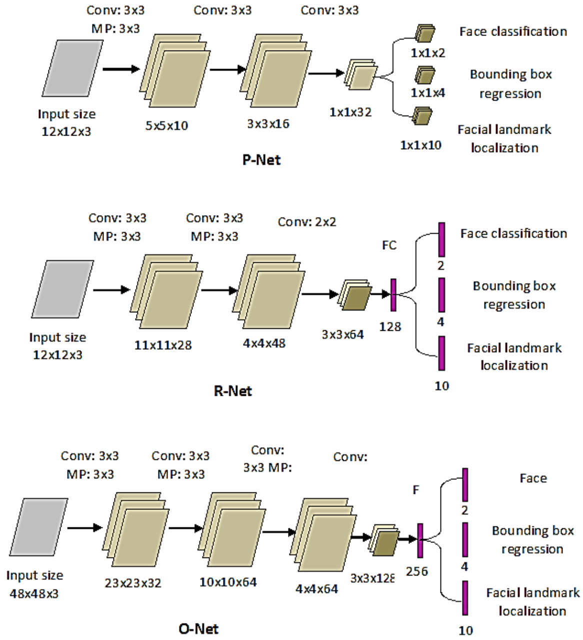

Fig. 6. MTCNN: Model for face detection stage architecture.

Fig. 6 illustrates the Multitasking Cascade Convolutional Neural Network (MTCNN), a convolutional network consisting of three layers (P-Net, R-Net, and O-Net) that can outperform most face detection tests within a short period. Taking an image and resizing it to Using different scales, can construct an image pyramid, which will be incorporated into the next three-stage cascaded network.

E. Face Verification

As the last task in MTCNN, face verification involves comparing two facial images to determine if they belong to the same individual. The layer's output image is rotated before rotating the output image. There is a separation between the eyes of 1.66 times the width of the clipping, and there is a separation between the eyes of twice the length of the cut. The model will implement the ROI alignment method to resize the feature map as the size of each sample's cropped feature map is inconsistent.

For this part of the network, a loss function may be defined as either a central or a cross-entropy loss function. The following are their definitions:

$$Cross\ Entropy\ Loss = \varphi \ \frac{1}{N}\sum_{i}^{N} f_i ln\left(Softmax\ (f_i)\right) \quad (6)$$

$$Centre\ Loss = \omega \ \frac{1}{N}\sum_{i}^{N} \|fv_i - c_i\|_2^2 \quad (7)$$

where N samples in the training of a batch, f_i is an estimate of the sample's probability vector i, f_i is the sample of true label i, fv_i is the prediction vector, c_i is a grouping of sample i, φ is the intensity of cross-entropy loss, ω is the weightiness of center loss.

It is determined whether the two images are $f_{v1}$and f_{v2} identical by calculating the cosine similarity between and corresponding to them. Cosine similarity can be calculated using the following formula:

$$\text{cosine}\, similarity = \frac{\sum_{i=1}^{256} f_{v1i} f_{v2i}}{\sqrt{\sum_{i=1}^{256} f_{v_{i1}^2} {}_i f_{v_{i1}^2}} \sqrt{\sum_{i=1}^{256} f_{v_{i1}^2}}} \quad (8)$$

It is assumed that two images belong to the same identity when their similarity exceeds the threshold. From the validation set, a threshold is calculated.

IV. Result and Discussion

This section aims to determine whether the suggested method is an effective strategy for mining complex samples. The offered face detector and alignment are then compared with existing works. A set of 2,845 images in the FDDB dataset contains annotations for 5,171 faces. As part of the face dataset, 393,703 bounding boxes with labeled faces were used in 33,103 impressions. A similar subset of tests is used, which contains annotations for 24,386 faces. The projected face detector is then evaluated in terms of its computational efficiency.

A. Training Data

In Regression based on bounding boxes, faces have recycled the estimation of bounding boxes, and faces of landmarks are used to identify the estimation of facial landmarks. For each network, the training data are as follows:

- P-Net: The system aimlessly cropped a number of a patch to gather positives, negatives, and fragment faces.
- R-Net: As part of the suggested framework, as a first step, use to detect faces from the dataset to collect positives, negatives, and part faces, while landmark faces are detected.
- O-Net: In the suggested work, use the first two stages to detect faces, similar to the way R-Net collects data.

To train the planned MTCNN detectors, use three tasks: classification of faces and non-faces, bounding box regression, and localization of facial landmarks.

1) Face classification

A two-class classification problem is used to formulate the learning objective. To calculate the cross-entropy loss for each sample, use the following formula:

$$L_{\boldsymbol{i}}^{\boldsymbol{det}} = -\left(y_{\boldsymbol{i}}^{\boldsymbol{det}} \log(pr_i) + \left(1 - y_{\boldsymbol{i}}^{\boldsymbol{det}}\right)\left(1 - log(pr_i)\right)\right) \quad (9)$$

where pr_i the probability is calculated by the network, $y_{\boldsymbol{i}}^{\boldsymbol{det}} \epsilon \{0,1\}$representsthe Ground-truth labels indicated by the notation

2) Bounding box regression

Each candidate's offset from the ground truth is determined to the nearest candidate window. A regression problem is used to formulate the learning objective, and the Euclidean loss is used for each sample xi:

$$L_{\boldsymbol{i}}^{\boldsymbol{box}} = \left\|\hat{y}_i^{box} - y_i^{box}\right\|_2^2 \quad (10)$$

$\boldsymbol{y^\wedge ibox}$ is the network's regressed target. $\boldsymbol{yibox}$ is includes the upper left coordination, the elevation, and the size of the ground-truth four-dimensional coordinate. Several types of relevant information are contained in the Bounding-Box property, including occlusion, invalid, expression, blur, illumination, and pose.

3) Facial landmark localization

In this work, the problem of detecting face landmarks $L_{\boldsymbol{i}}^{\boldsymbol{landmark}}$ is presented as a regression problem and a minimization of Euclidean loss is achieved as follows:

$$L_{\boldsymbol{i}}^{\boldsymbol{landmark}} = \left\|\hat{y}_i^{landmark} - y_i^{landmark}\right\|_2^2 \quad (11)$$

Likewise, $\hat{y}_i^{landmark}$ is Feature coordinates from the network that have been regressed. $\hat{y}_i^{landmark}$is composed of five coordinates.

4) Multi-source training

As each CNN employs a different task, Images used in training are employed in the process of learning. To illustrate, only compute Li(det) for the sample of background region, while the other two losses are left unchanged. Using a sample type indicator, can be accomplished directly. Therefore, the overall learning target is as follows:

$$min \sum_{i=1}^{N} \sum_{j\epsilon\{det,box,landmark\}} \alpha_j \beta_i^j L_i^j \quad (12)$$

P-Net: FCN begins with this stage. Their architecture does not include the dense layer. Candidate windows can be obtained by using the P-Net and the regression vectors for their bounding boxes.

R-Net: P-Net candidates are all considered for R-Net. As Network architecture's final stage, there is a dense layer; this network becomes CNN rather than FCN, as was the case previously. An R-Net is capable of determining whether an image contains a face using four elements, which represent the bounding box for the face, and ten elements, which represent the facial landmarks.

O-Net: In this step, the R-Net is similar to the output network; however, the R-Net has the following characteristics: it is aimed at describing the face in greater detail and generating the five facial landmark positions.

B. Evaluation Criteria

The following are their descriptions:

$$ACC = \frac{T}{ALL} \quad (13)$$

$$FAR = \frac{NFA}{NIRA} \quad (14)$$

$$FRR = \frac{NFR}{NGRA} \tag{15}$$

$$err\ 0 = \frac{1}{N}\sum_{i}^{N}\frac{\frac{1}{L}\sum_{j}^{L}|pij-yij|_2}{|lo_{ij}-ro_{ij}|_2} \tag{16}$$

$$err\ c = \frac{1}{N}\sum_{i}^{N}\frac{\frac{1}{L}\sum_{j}^{L}|pij-yij|_2}{|lc_{ij}-rc_{ij}|_2} \tag{17}$$

where, *T* and ALL indicate the tests, NIRA represents the number of errors accepted, and NFR signifies the correct rejections. *L* represents the points and *p* represents the point on prediction. NGRA identifies the same class tests and lc and rc represent the positions of left and right center eyes.

C. Effectiveness and Efficiency

A comparison of the loss curves of two O-Nets is a study was conducted to assess the effectiveness of the planned strategy for mining hard samples online. As a means of making a more direct comparison, train the O-Nets only for the task of face classification. These two O-Nets have the same training parameters, including the initialization of the network. Fig. 7 illustrates a comparison of the loss curves produced by different methods of training. Performance can be improved by using the hard sampling mining method.

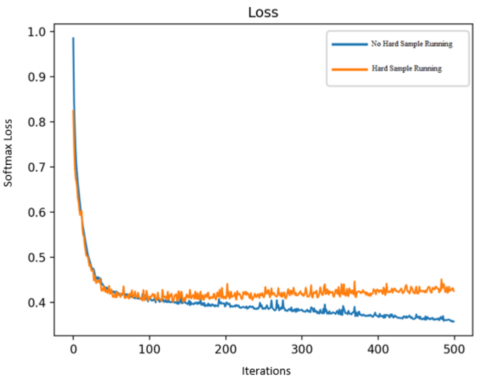

Fig. 7. Softmax loss validation of active facial patches using MTCNN with hard sample and without hard sample.

A comparison of two different O-Nets on Face Detection Data Set Benchmark (FDDB) was conducted to the bounding box regression concerned. Fig. 8 shows that the results show that the suggested approach outperforms all existing methods by a considerable margin.

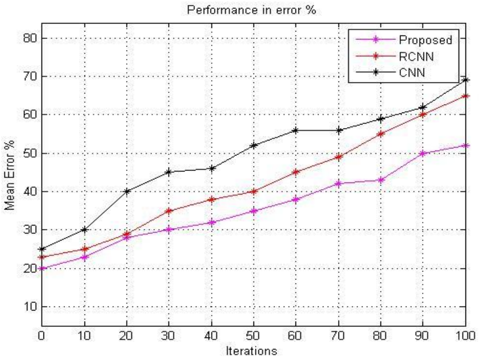

Fig. 8. Performance measures based on mean error.

Before analysing the detection accuracy of the active patches, have to look at how it performs whenever a variable number of prevalent patches is used. Fig. 9 shows the findings with varying numbers of active patches.

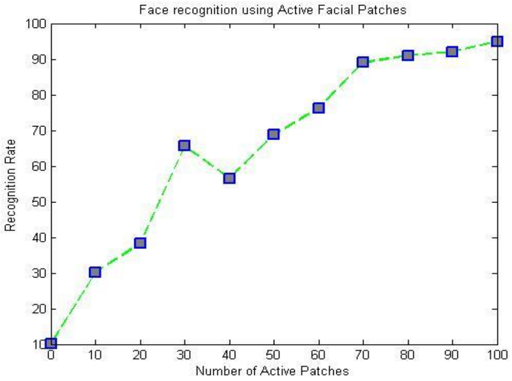

Fig. 9. Active patches recognition rate.

Here used empirical analysis to estimate the histogram's ideal resolution and bin width. With face resolutions ranging from 48×48 to 192×192, the performance of several feature vectors for expression recognition was examined. Down sampling the original photographs produced the low-resolution versions. The classifiers were examined and trained using various face picture resolutions.

TABLE I. VARIOUS IMAGE NUMBERS WERE COMPARED FOR ACCURACY

No. of Faces Tested	Precision
Image-1	0.97
Image-2	0.91
Image-3	0.99
Image-4	1.02
Image-5	1.03

According to Table I, an image calculation center with three image calculation centers can provide 99.9% accuracy, while an image calculation center with four or more can provide 100% accuracy.

A comparison was made between different algorithms on the Labeled Faces in the Wild (LFW) dataset in terms of accuracy as shown in Fig. 10.

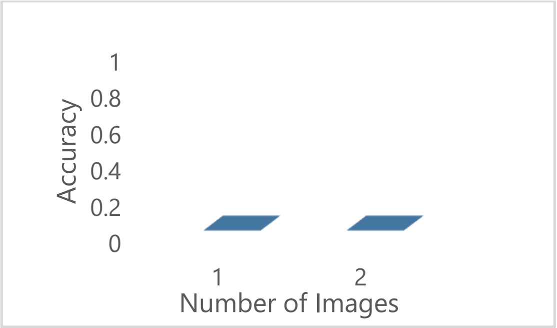

Fig. 10. Comparison of accuracy.

D. Efficiency of Runtime

Due to the cascade structure of the expected method, can detect and align joint faces very quickly. On a Central

Processing Unit (CPU) of 2.60G Hz, it takes 16 frames per second and on a Graphics Processing Unit (GPU) (Nvidia Titan Black), it takes 99 frames per second. Currently using un-optimized MATLAB code in the implementation. Table II describes the comparison speed of the suggested system during validation and is plotted in Fig. 11.

TABLE II. COMPARISON OF SPEED IN VALIDATION WITH THE ACTIVE FACIAL PATCHES USING MTCNN MODEL

Hider	Image 1	Image 2	Image 3	Image 4	Image 5
P-Net	0.31	0.32	0.35	0.31	0.33
R-Net	0.42	0.41	0.43	0.42	0.41
O-Net	1.02	1.02	1.035	1.04	1.05

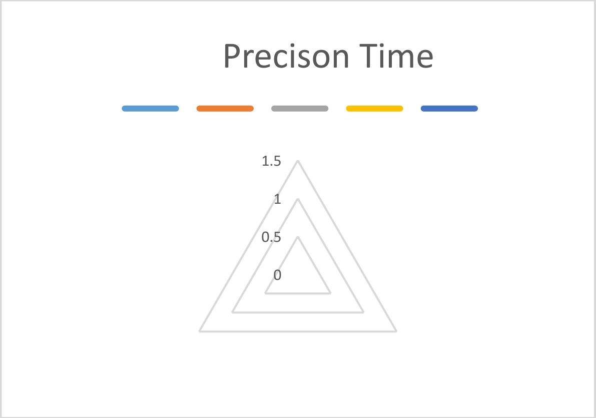

Fig. 11. Precision time.

Table III shows that the suggested structure gives better accuracy in face identification using active facial patches and takes less time to detect.

TABLE III. AN ANALYSIS OF METHODS FOR DETECTING ACTIVE FACIAL PATCHES USING THE MTCNN MODEL

Active facial patches detection method	CNN	Active facial patches using MTCNN
Face Recognition Accuracy (%)	92.42	97.62
Time (in sec. per image)	1.6746	0.2955

V. CONCLUSION

This paper offered a framework for active facial patch recognition using MTCNN to analyze camera frames as they appear and to identify the individual. MTCCN outperforms several face identification tests exceptionally well using three convolutional networks while maintaining real-time performance. The Active facial patches using MTCNN for detecting human faces in real time was created and analyzed, and the system accurately recognized human faces 97.62% of the time. In testing, the methods outperformed approaches while maintaining real-time performance on several challenging benchmarks. The primary goal is to further enhance the performance of the suggested algorithm by leveraging the inherent correlation among various aspects of face analysis in the future.

Further research can be extended to develop more sophisticated algorithms for selecting the most discriminative patches from a face. It could involve exploring different feature representations, patch selection criteria, or incorporating deep learning approaches to improve the accuracy and efficiency of patch selection.

CONFLICT OF INTEREST

The authors declare no conflict of interest.

AUTHOR CONTRIBUTIONS

Conceptualization, methodology, Data collection, Validation and formal analysis Krishnaraj M; Writing—original draft preparation, Writing—review, editing, supervision and project administration Jeberson Retna Raj R.; all authors had approved the final version.

REFERENCES

[1] Y. Guan, J. Fang, and X. Wu, "Multi-pose face recognition using cascade alignment network and incremental clustering," *Signal, Image and Video Processing*, vol. 15, pp. 63–71, 2021.

[2] H. A. H. Mahmoud and H. A. Mengash, "A novel technique for automated concealed face detection in surveillance videos," *Personal and Ubiquitous Computing*, vol. 25, pp. 129–140, 2021.

[3] S. Wang, S. Yin, L. Hao, and G. Liang, "Multi-task face analyses through adversarial learning," *Pattern Recognition*, vol. 114, 107837, 2021.

[4] Y. Zhao *et al.*, "Joint face alignment and segmentation via deep multi-task learning," *Multimedia Tools and Applications*, vol. 78, pp. 13131–13148, 2019.

[5] H. P. P. Win *et al.*, "Face recognition system based on convolution neural networks," *International Journal of Image, Graphics and Signal Processing*, pp. 1923–1927, 2021.

[6] D. Zeng, R. Veldhuis, and L. Spreeuwers, "A survey of face recognition techniques under occlusion," *IET Biometrics*, vol. 10, no. 6, pp. 581–606, 2021.

[7] B. F. Wu, B. R. Chen, and C. F. Hsu, "Design of a facial landmark detection system using a dynamic optical flow approach," *IEEE Access*, vol. 9, 2021.

[8] A. R. Hazourli, A. Djeghri, H. Salam, and A. Othmani, "Multi-facial patches aggregation network for facial expression recognition and facial regions contributions to emotion display," *Multimedia Tools and Applications*, vol. 80, pp. 13639–13662, 2021.

[9] K. Y. Tsai *et al.*, "Frontalization and adaptive exponential ensemble rule for deep-learning-based facial expression recognition system," *Signal Processing: Image Communication*, 116321, 2021.

[10] R. He, Z. Xing, W. Tan, and B. Yan, "Feature pyramid network for multi-task affective analysis," arXiv preprint, arXiv:2107.03670, 2021.

[11] P. Cai and H. M. Quan, "Face anti-spoofing algorithm combined with CNN and brightness equalization," *Journal of Central South University*, vol. 28, no. 1, pp. 194–204, 2021.

[12] S. Minaee, M. Minaei, and A. Abdolrashidi, "Deep-emotion: Facial expression recognition using attentional convolutional network," *Sensors*, vol. 21, no. 9, p. 3046, 2021.

[13] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, "Joint face detection and alignment using multitask cascaded convolutional networks," *IEEE Signal Processing Letters*, vol. 23, no. 10, pp. 1499–1503, 2016.

[14] L. B. Rahmadi, K. M. Lhaksmana, and D. Rhomanzah, "LBP advantages over CNN face detection method on facial recognition system in NOVA robot," *Indonesia Journal on Computing (Indo-JC)*, vol. 5, no. 2, pp. 67–80, 2020.

[15] J. Du, "High-precision portrait classification based on MTCNN and its application on similarity judgement," *Journal of Physics: Conference Series*, vol. 1518, no. 1, 012066, 2020.

[16] H. N. Vu, M. H. Nguyen, and C. Pham, "Masked face recognition with convolutional neural networks and local binary patterns," *Applied Intelligence*, vol. 52, no. 5, pp. 5497–5512, 2020.

[17] L. Zhou, H. Zhao, and J. Leng, "MTCNet: Multi-task collaboration network for rotation-invariance face detection," *Pattern Recognition*, vol. 124, 108425, 2020.

[18] H. Li *et al.*, “UFaceNet: Research on multi-task face recognition algorithm based on CNN,” *Algorithms*, vol. 14, no. 9, 268, 2021.

[19] C. Yan *et al*., “Age-invariant face recognition by multi-feature fusion and decomposition with self-attention,” *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 18, pp. 1–18, 2020.

[20] N. Bharathiraja *et al*., “Abnormality detection in video using gaussian mixture model and recurrent conditional random field,” *Solid State Technology*, pp. 5440–5449, 2020.

[21] W. Liu *et al*., “Convolutional two-stream network using multi-facial feature fusion for driver fatigue detection,” *Future Internet*, vol. 11, no. 5, 115, 2019.

[22] M. U. Ahmed *et al*., “Wild facial expression recognition based on incremental active learning,” *Cognitive Systems Research*, vol. 52, pp. 212–222, 2018.

[23] M. J. Khan, M. J. Khan, A. M. Siddiqui, and K. Khurshid, “An automated and efficient convolutional architecture for disguise-invariant face recognition using noise-based data augmentation and deep transfer learning,” *The Visual Computer*, pp. 1–15, 2022.

[24] K. M. Roozbahani and H. S. Zadeh, “Face detection from blurred images based on convolutional neural networks,” in *Proc. 2022 International Conference on Machine Vision and Image Processing (MVIP)*, 2022, pp. 1–10.

[25] S. Feng, X. Nong, and H. Hu, “Cascaded structure-learning network with using adversarial training for robust facial landmark detection,” *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 18, no. 2, pp. 1–20, 2022.

[26] K. Pradeepa *et al*., “Artificial neural networks in healthcare for augmented reality,” in *Proc. 2022 Fourth International Conference on Cognitive Computing and Information Processing (CCIP)*, 2022, pp. 1–5.

[27] D. Vinod, N. Bharathiraja, M. Anand, and A. Antonidoss, “An improved security assurance model for collaborating small material business processes,” *Materials Today: Proceedings*, vol. 46, 2021.

[28] C. Zhou and R. Zhi, “Learning deep representation for action unit detection with auxiliary facial attributes,” *International Journal of Machine Learning and Cybernetics*, pp. 1–13, 2020.

[29] L. Wang, X. Yu, T. Bourlai, and D. N. Metaxas, “A coupled encoder–decoder network for joint face detection and landmark localization,” *Image and Vision Computing*, vol. 87, pp. 37–46, 2019.

[30] N. Bharathiraja *et al*., “The smart automotive webshop using high end programming technologies,” *Intelligent Communication Technologies and Virtual Mobile Networks*, pp. 811–822, 2023.

[31] H. Sadeghi and A. A. Raie, “HistNet: Histogram-based convolutional neural network with Chi-squared deep metric learning for facial expression recognition,” *Information Sciences*, vol. 608, pp. 472–488, 2022.

[32] A. Khalifa *et al*., “Face recognition and tracking framework for human–robot interaction,” *Applied Sciences*, vol. 12, no. 11, 5568, 2022.

[33] Y. S. Dosso *et al*., “NICUface: Robust neonatal face detection in complex NICU scenes,” *IEEE Access*, vol. 10, pp. 62893–62909, 2022.

[34] A. Umamageswari, N. Bharathiraja, and D. S. Irene, “A novel fuzzy c-means based chameleon swarm algorithm for segmentation and progressive neural architecture search for plant disease classification,” *ICT Express*, pp. 2405–9595, 2021.

[35] M. Yuan *et al*., “Minor privacy protection through real-time video processing at the edge,” in *Proc. 2020 29th International Conference on Computer Communications and Networks (ICCCN)*, 2020, pp. 1–6.