Zafar et al. EURASIP Journal on Image and Video Processing (2019) 2019:10
https://doi.org/10.1186/s13640-019-0406-y

EURASIP Journal on Image and Video Processing

RESEARCH Open Access

Face recognition with Bayesian convolutional networks for robust surveillance systems

Umara Zafar[1], Mubeen Ghafoor[1], Tehseen Zia[1], Ghufran Ahmed[1], Ahsan Latif[2], Kaleem Razzaq Malik[3] and Abdullahi Mohamud Sharif[4*]

Abstract

Recognition of facial images is one of the most challenging research issues in surveillance systems due to different problems including varying pose, expression, illumination, and resolution. The robustness of recognition method strongly relies on the strength of extracted features and the ability to deal with low-quality face images. The proficiency to learn robust features from raw face images makes deep convolutional neural networks (DCNNs) attractive for face recognition. The DCNNs use softmax for quantifying model confidence of a class for an input face image to make a prediction. However, the softmax probabilities are not a true representation of model confidence and often misleading in feature space that may not be represented with available training examples. The primary goal of this paper is to improve the efficacy of face recognition systems by dealing with false positives through employing model uncertainty. Results of experimentations on open-source datasets show that 3–4% of accuracy is improved with model uncertainty over the DCNNs and conventional machine learning techniques.

Keywords: Face recognition, Unconstrained face images, Convolutional neural networks, Bayesian convolutional neural networks, Model uncertainty

1 Introduction

Face recognition became the most sought-after research area due to its applications in surveillance systems, law enforcement applications, and access control and extensive work has been reported in the literature in the last decade [1]. The process of face recognition refers to identifying the person by comparing some features of a new person (input sample) with the known persons in the database. Face recognition pipeline consists of four main phases: face region detection, alignment, feature extraction, and classification [2] where the most crucial phase is feature extraction. Hand-crafted features have achieved reasonable results for constrained environments [3–5]. However, the recognition of unconstrained face images is an evolving and challenging field in the context of the real-world issues such as varying poses, expressions, illumination, and quality of images [6]. Many researchers tried different approaches for improving the recognition accuracy of unconstrained facial images [7] using different classification techniques such as support vector machine (SVM) [8], stochastic modeling [9], neural networks [10] and ensemble classifiers [11]. Recently, deep learning (DL)-based techniques, especially deep convolutional neural networks (DCNNs) have shown excellent results in face recognition by discovering intricate features in large datasets using the backpropagation algorithm [2, 12–15]. The DCNN-based models use softmax for quantifying model confidence of a class for an input face image in order to make a decision. However, the softmax probabilities are not a true representation of model confidence and often misleading in feature space that cannot be represented with available training examples [16]. In this research, we observe (an example is shown in Fig. 5) that such a scenario happens in borderline cases (i.e., faces with smaller intra-class variations). The primary goal of this paper is to improve the efficacy of face recognition systems by dealing with false positives through employing model uncertainty. The proposed study highlights the

* Correspondence: Abdullahi.shariif@uniso.edu.so
[4]University of Somalia, Mogadishu, Somalia
Full list of author information is available at the end of the article

advantage of Bayesian deep convolutional neural networks (B-DCNN) over DCNN for robust recognition of facial images particularly in cases where intra-class variations are low.

Rest of the paper is prepared as follows: related work and background are presented in Sections 2 and 3. The algorithm proposed for the face recognition is elaborated in Section 4. Section 5 is dedicated to the results and discussions, while the last section concludes the paper.

2 Related work

2.1 Machine learning-based face recognition approaches

Face recognition has attracted lots of attention, but current systems are yet far from human perception capabilities. A critical issue in face recognition is finding apt descriptors for modeling faces. Based on the descriptors, face recognition techniques can be broadly divided into three categories; holistic, feature-based, and hybrid face matching [17]. In holistic methods, the face is modeled by extracting a set of global features [18]. In this context, principal component analysis (PCA) [18], Mahalanobis cosine PCA (MahCos PCA) [19], linear discriminant analysis (LDA) [20], and 2D PCA [21] have been explored. On the other hand, local feature-based descriptors have shown robustness to variance in pose and illumination [22]. Biswas et al. [23] described local facial landmarks with the Scale-Invariant Feature Transform (SIFT) features. At each landmark, Gabor magnitude factors are extracted as the pose-robust feature. Fischer et al. [24] suggested that extracting landmarks for non-frontal faces had a degrading effect on the recognition results and proposed robust landmark extraction around nose tip and mouth corners. Guo et al. [25] proposed local binary patterns (LBP)-based features extraction for encoding facial landmarks.

Hybrid methods make use of holistic features around essential facial points [26–28]. Ding et al. [26] fused component-level and landmark-level approaches by using Dual-Cross Pattern (DCP) feature of landmarks which belong to the same facial component. Liao et al. [28] proposed alignment-free partial face recognition by extracting Multi-Keypoint Descriptors (MKD) for sparse representation of facial images. Arashloo et al. [29] computed the normalized energy of Markov random field (MRF) features to match face images with slight pose invariance.

In scenarios where limited face images are available for training, virtual image creation based on linear combination of symmetrical face images [30] and Rotated Face Model (RFM) [31] techniques have provided an alternative solution [32]. Synthesis of virtual deformable face models using 3D model fitting [33] and Generic Elastic Model (GEM) [34, 35] have achieved promising results. Hu et al. [36] used FaceGen Modeler commercial software for 3D modeling of the single 2D image to generate different pose varied synthesized images.

2.2 Deep learning-based face recognition approaches

Although machine learning techniques for facial recognition have provided decent results, these techniques do not perform well under unconstrained environments. This is mainly because machine learning approaches rely on hand-crafted features or representations selected by human experts that may work for one scenario and fail for other situations. On the other hand, deep learning (DL)-based approaches have proven to be most suitable as the representations and features are discovered automatically from data by the back-propagation learning technique.

Taigman et al. [2] performed face alignment using explicit 3D face modeling and proposed a nine-layer deep neural network for learning generic face representations in unconstrained environments. Wen et al. [37] proposed a robust DCNN using softmax loss function jointly with center loss function to increase the discriminative power of learned features for face recognition. Sun et al. [38] proposed DCNN-based face recognition system (DeepID2) that combined the classification and verification loss functions to learn more discriminative features. The generalized DeepID2 features are extracted from the different identities to increase inter-personal verification, whereas the same identity's extracted features reduce the intra-personal variations to incorporate new identities that are not available in the training data. Sun et al. [13] proposed DeepID3 that further enhanced the results of DeepID2 [38] by creating an ensemble of two DCNN architectures based on VGG net [39] and GoogLeNet [40]. Schroff et al. [14] proposed a DCNN called "FaceNet" that computed face similarity based on distances in Euclidean space learned directly from face images. The authors employed a triplet loss function to learn feature embeddings used to perform face recognition.

DL algorithms have proved to be successful in learning dominant representations from high-dimensional face data. However, in DL-based classification, predictive probabilities obtained at the end of the pipeline (i.e., softmax output) are often erroneously interpreted as model confidence, which is not true. Understanding what a model does not know is a critical part of machine learning systems. Conventional DL tools for regression and identification do not detect uncertainty of the model. To the best of our knowledge, no study has considered exploiting the recent integration of model uncertainty tool within DL to deal with uncertain faces (i.e., confusing face). In this study, the focus is on the Bayesian DCNN (B-DCNN) [41] that can efficiently model the uncertainty in the DL model for face images.

3 Background and preliminaries

DL consists of a set of techniques that can automatically learn the representations (i.e., features) from raw data used for classification tasks [12]. The ability to learn representations at multiple levels of abstraction merely by stacking non-linear layers allow DL methods to achieve better generalization on highly complex tasks such as image classification. DCNN is a type of DL methods that have recently become modus of operandi for image recognition tasks due to its remarkable achievements in this area [39]. This success is partly because of its robust and precise assumptions about the natural images (i.e., locality of associations between pixels and statistical stationarity) [12] and partially due to ease of optimization because of significantly lesser parameters as compared to feed-forward networks [42].

3.1 Convolutional neural networks

A typical architecture of DCNN is composed of convolution, pooling, fully connected layers, and softmax [43] layers as shown in Fig. 1. A short description of these component layers is given below:

- Convolution layer: In this layer, each unit is connected to a local patch of units in the previous layer through a set of weights called a filter. The unit activation is called feature map and computed by applying non-linearity functions over the locally weighted sums.
- Pooling layer: While convolution layer learns features, the pooling layer combines semantically related features into a single feature. Each unit in a pooling layer takes input from a patch of units in the previous layer and outputs a maximum or average of these values.
- Fully-connected layer: In this layer, each unit is connected to all the units in the previous layer. Typically, the convolution and pooling layer are stacked in two or three stages before using fully-connected layers.
- Softmax layer: Softmax function is used for converting the features into probabilities of the classes. This layer contains as many units as the number of classes. The softmax function is given in Eq. 1 [44]:

$$\text{Softmax}(a_i) = \frac{e^{a_i}}{\sum_{j=1}^{m} e^{a_j}} \tag{1}$$

where Softmax(a_i) and a_i represent respectively the probability and feature of ith class. The nominator is an unnormalized measure of probability, and denominator is used to normalize the probability distribution over m classes.

Different activation functions such as rectified linear units (ReLU) [45], leaky-ReLU (LReLU) [46], exponential linear unit (ELU) [47], and scaled-ELU (SELU) [15] can be used to model non-linearity for determining the output of neurons. ReLU [45] is one of the most commonly used activation functions that give non-negative outputs and prevents the vanishing gradient issue in deep learning tasks [47]. However, ReLU-based networks can result in dead neurons due to the zero gradient in the negative part of ReLU [46]. LReLUs [46] can be used to rectify this problem by introducing a small, non-negative gradient in the negative part of the function but they are not very robust against noise [47]. Recently, ELU [47] activation function was proposed which converges faster and is more robust against noise. ELUs usually perform better than ReLU and LReLUs in networks with over five layers, but ELUs can saturate for large negative values [47]. SELU is a variant of ELU with an extra scaling parameter, and it shows good results for fully connected networks [15]. Learning phase of the DCNN model deals with optimizing weights of the units with the objective to minimize misclassifications. Stochastic gradient descent is typically used as an optimization procedure where gradients over the weights are computed by using the standard back-propagation algorithm.

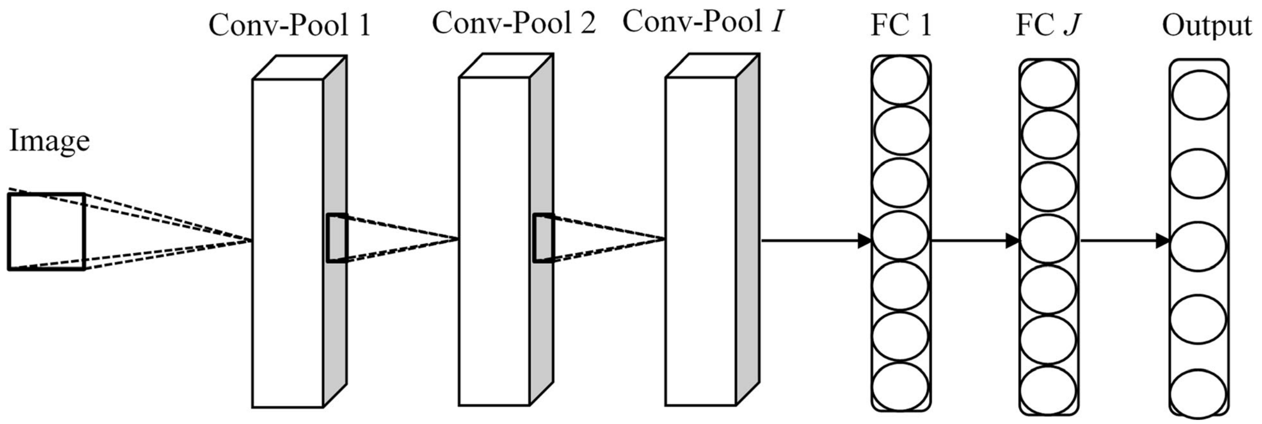

Fig. 1 Deep convolutional neural network (DCNN) architecture for face recognition

3.2 Bayesian convolutional neural networks

In order to deal with the lack of visual discernibility between face images, we want a model capable of representing prediction uncertainty. Current methods such as [48–50] are based on kernel methods where image pairs are fed for measuring similarity. The similarity is then used as an input to a classifier such as SVM. However, we are using DCNN models and are interested in a principled Bayesian approximation of uncertainties. A Bayesian equivalent of DCNN is proposed in [51]. These Bayesian DCNNs (B-DCNN) are a type of DCNNs that have prior probability distributions over a set of model parameters $\omega = \{W_1, \ldots, W_L\}$:

$$\omega \sim p(\omega) \tag{2}$$

A likelihood model can be defined by assuming a standard Gaussian prior p (ω) for classification as given in Eq. 3 [44]:

$$p(y = c|\mathbb{x}, \omega) = \mathrm{softmax}(f^{\omega}(\mathbb{x})) \tag{3}$$

The inference in the B-DCNN model is performed by employing stochastic regularization techniques such as dropout [52, 53]. To perform the inference, a model is trained with dropout before every network layer. Also, the dropout is used at the time of testing and sampling from the approximate posterior. This is formally equivalent to perform an approximate variational inference where the task is to find a tractable distribution $q_{\theta}^{*}(\omega)$ using a training dataset $\mathcal{D}_{\text{train}}$. This is achieved by minimizing Kullback-Leibler (KL) divergence with the true model posterior $\mathrm{p}(\omega|\mathcal{D}_{\text{train}})$ [44]. Dropout can be considered as a type of variational Bayesian approximation, where the approximated distribution is a blend of two Gaussians with small variances and one of the Gaussians is fixed at zero mean. The uncertainty in the weights brings uncertainty in the prediction through marginalizing the approximate posterior by Monte Carlo integration as given in Eqs. 4–6 [41]:

$$p(y = c|\mathbb{x}, \mathcal{D}_{\text{train}}) = \int p(y = c|\mathbb{x}, \omega) p(\omega|\mathcal{D}_{\text{train}}) d\omega \tag{4}$$

$$\approx \int p(y = c|\mathbb{x}, \omega) q_{\theta}^{*}(\omega) d\omega \tag{5}$$

$$\approx \frac{1}{T} \sum_{t=1}^{T} p(y = c|\mathbb{x}, \hat{\omega}_t) \tag{6}$$

where $q_{\theta}^{*}(\omega)$ is referred to as dropout distribution [54].

4 Proposed methodology

Face recognition task can be formulated as given a face images dataset $X = \{x_1, \ldots, x_N\}$ where $X \in [0; 1]^{h \times w}$ (h and w symbolizes height and width of the N images) and set of corresponding labels $Y = \{y_1, \ldots, y_N\}$ where each label belongs to a set of unique classes C. The objective is to learn a function f that maps the set of input images X to a set of labels Y such that the output label C^{out} is similar to ground-truth label C^{gt}.

The method we employ to form a B-DCNN architecture is dropout [41]. In [51], the authors have shown a relationship between dropout and variational inference in B-DCNN with Bernoulli distributions over the network's weights. We used this approach to represent model uncertainties while classifying facial images. We want to find the posterior distribution over the convolutional weights of B-DCNN, given the face training data X and labels Y as given in Eq. 7:

$$p(W|X, Y) \tag{7}$$

Generally, this is not a tractable distribution; hence, the distribution over the weights is required to be approximated [51]. We employ variational inference for approximating these weights [51]. This approach facilitates to optimize the approximate distribution over weights, $q(W)$, by minimizing the Kullback-Leibler (KL) divergence between $q(W)$ and $p(W| X, Y)$ as given in Eq. 8 [44]:

$$KL(q(W) \mid |p(W|X, Y)) \tag{8}$$

where $q(W_i)$ can be defined for every $K \times K$ dimensional convolutional layer i containing j units as given in Eq. 9:

$$b_{i,j} \sim \mathrm{Bernoulli}(p_i) \quad for\ j = 1, 2, \ldots, K_i \qquad W_i = M_i \ \mathrm{diag}(b_i) \tag{9}$$

Here, b_i and M_i represent vectors of random variables distributed with Bernoulli distribution and variational parameters respectively. Hence, the B-DCNN model is obtained [51]. Although we can optimize the dropout probabilities p_i, they are fixed to a standard value of 0.5 [41]. It is shown in [51] that minimizing the cross entropy loss function leads to minimize KL divergence. Thus, the learning of a network with stochastic gradient descent leads to learn a distribution over network's weights. We train our B-DCNN model for face recognition with dropout. In order to get the posterior distribution of class probabilities, the dropout is used at test time also to sample the posterior distribution over the weights. The mean and variance of the samples are used respectively as confidence and uncertainty for each class. The final classification decision is made on the basis of a simple heuristic function as given in Eq. 10:

$$\begin{array}{ll} i^{\text{th}}\text{class}, & \text{if } d_i - c_i > 0 \\ D, & \text{otherwise} \end{array} \tag{10}$$

Here c_i is the confidence of ith class (the class predicted by the model), D indicates doubt or rejection class and d_i is rejection threshold of ith class and defined on the basis of model confidence c_i and uncertainty u_i for each class i as $d_i = c_i - u_i$.

For image classification, we used DCNN because of its state-of-the-art performance in image classification tasks [39]. Figure 2 shows a schematic of the face recognition and model uncertainty representation procedure. Mainly, it consists of three types of modules: feature extraction, feature selection, and prediction. Each module includes a series of operations that define layer-wise functionality. The feature extraction module at stage l represented as $g^{(l)}$extracts features $H(l)$ as given in Eq. 11:

$$\begin{aligned} H^{(l)} &= g^{(l)}\left(H^{(l-1)}; W^{(l)}, b^{(l)}\right) \\ &= \text{normalize}\left(\text{pool}\left(\text{relu}\left(W^{(l)} * H^{(l-1)} + b^{(l)}\right)\right)\right) \end{aligned} \tag{11}$$

where $*$ operator represents convolution, $W(l)$ and $b(l)$ are the weights and biases of the l^{th} layer, respectively, and $H(l-1)$ is either the input image X for $l = 1$ (i.e., $H(0) = X$) or activation of $l - 1th$ layer for $l > 1$. Specifically, feature extraction involves operations in the following order: convolution, non-linear transformation, max-pooling, and local normalization [42]. The feature selection module $f^{(l)}$involves dot product operation followed by non-linear transformation as given in Eq. 12:

$$\begin{aligned} H^{(l)} &= f^{(l)}\left(H^{(l-1)}; W^{(l)}, b^{(l)}\right) \\ &= \left(\text{relu}\left(W^{(l)}.H^{(l-1)} + b^{(l)}\right)\right) \end{aligned} \tag{12}$$

where (.) indicate dot product, and $H(l-1)$ represents activation of $l - 1$th hidden layer. Finally, the prediction module involves a softmax [16] operation to gives the probability over each output class C as given in Eq. 13:

$$p(C|X; W, b) = \text{softmax}\left(W^{(l)}.H^{(l-1)} + b^{(l)}\right) \tag{13}$$

The feature extraction, selection, and prediction modules are stacked together to construct the DCNN model architecture as given in Eq. 14:

$$p(C|X; W, b) = \text{softmax}\left(f^{(5)}\left(f^{(4)}\left(g^{(3)}\left(g^{(2)}\left(g^{(1)}(X)\right)\right)\right)\right)\right) \tag{14}$$

5 Results and discussion

The results of the proposed face recognition algorithm are presented by comparing recognition accuracies with other methods available in the literature on two open source databases [55, 56]. The experimental setup is discussed in the following section.

5.1 Experimental setup

The two databases used for experimentation are specifically selected to account for variation in pose, facial accessories, position, and illumination. Both databases are mentioned below:

1) AT&T Face Database (formerly called ORL) [56]: This database consists of 400 grayscale images of 40 different individuals taken with the varying pose (straight, left, right). Some sample images from this database are shown in Fig. 3a. This database is divided into 320 images for training and 80 images for testing.
2) EURECOM Kinect Face Database (EKFD) [55]: This database consists of 936 images of 52 different individuals, taken with the varying pose (straight, left, right and up), expression (neutral, happy) and eyes (wearing glasses or not), and illumination. Some sample images from this database are shown

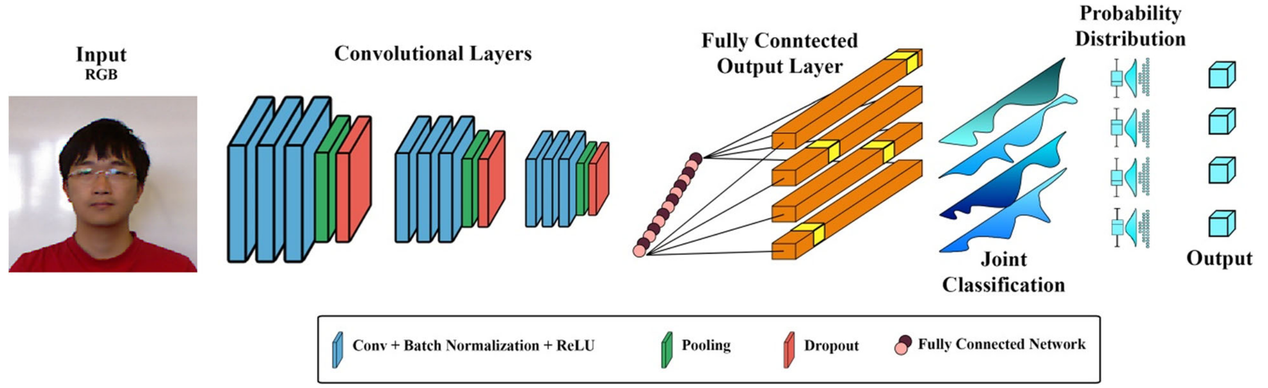

Fig. 2 Schematic illustrations of the proposed face recognition method

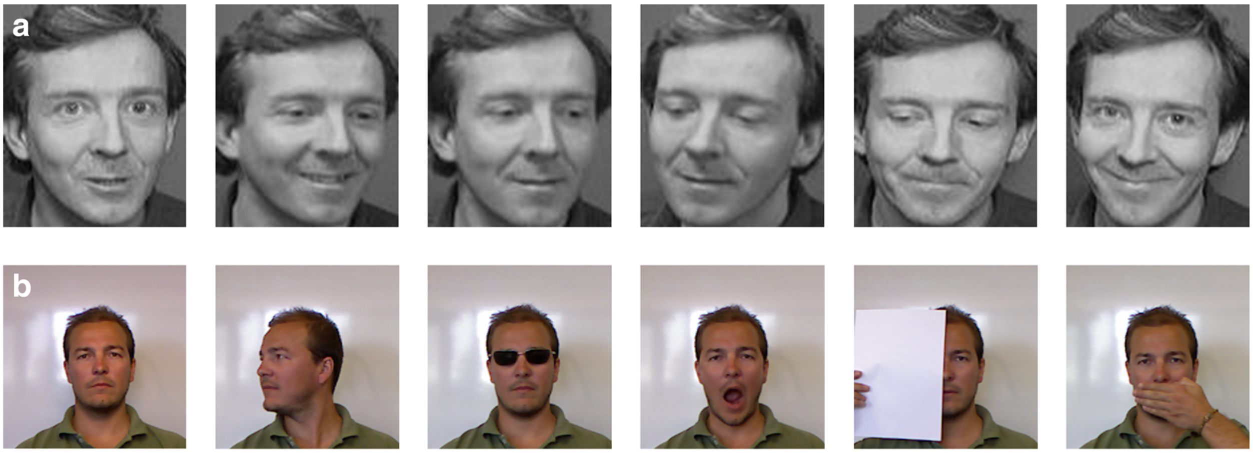

Fig. 3 Sample images from both datasets. **a** AT&T face dataset. **b** EURECOM Kinect Face Database

in Fig. 3b. This database is divided into 780 images for training and 156 images for testing.

The proposed face recognition algorithm is tested on three different DCNN architectures given in Table 1. The deep learning library used is Tensorflow [57], and all experiments are performed on Google Collaboratory platform (https://colab.research.google.com).

5.2 Results of the proposed DCNN methodology

Results of the proposed DCNN and B-DCNN-based face recognition for architectures mentioned above from Table 1 are calculated based on the model learning curves (accuracy and loss) for both EKFD [55] and AT&T [56] databases. Figure 4a, b presents the model accuracy and loss graphs of the best performing architecture (Arc-2) for both EKFD [55] and AT&T [56] databases respectively. The proposed DCNN model Arc-2 achieves recognition accuracies of 94.2% and 97.5% on EKFD [55] and AT&T [56] respectively.

5.3 Results of the proposed Bayesian DCNN (B-DCNN) methodology

The results of the proposed B-DCNN-based models have achieved an additional improvement of around 3–4% as compared to the proposed DCNN models for all model architectures of Table 1. Specifically for Arc-2, the proposed B-DCNN methodology achieved accuracies of 98.1% and 100% on EKFD [55] and AT&T [56] databases respectively. Table 2 presents a comparison of face recognition accuracies of proposed DCNN and B-DCNN methodologies with other methods in literature such as PCA [18], MahCos PCA [19], and DCNNs proposed by Lee et al. [58] and Vinay et al. [59].

Table 1 Description of network architectures used for experimentation

Layer No.	Architecture # 1 (Arc-1)		Architecture # 2 (Arc-2)		Architecture # 3 (Arc-3)	
	Layer type @ size	Activation	Layer type @ size	Activation	Layer type @ size	Activation
1	C1 @ 3 × 3 × 32	ReLU + dropout	C1 @ 3 × 3 × 32	ReLU + dropout	C1 @ 3 × 3 × 32	ReLU + dropout
2	P1 @ 2 × 2	–	P1 @ 2 × 2	–	P1 @ 2 × 2	–
3	C2 @ 3 × 3 × 32	ReLU	C2 @ 3 × 3 × 32	ReLU	C2 @ 3 × 3 × 32	ReLU
4	P2 @ 2 × 2	–	P2 @ 2 × 2	–	P2 @ 2 × 2	–
5	C3 @ 3 × 3 × 64	ReLU	C3 @ 3 × 3 × 64	ReLU	C3 @ 3 × 3 × 32	ReLU
6	P3 @ 2 × 2	–	P3 @ 2 × 2	–	P3 @ 2 × 2	–
7	FC1 @ 512	ReLU + dropout	C4 @ 3 × 3 × 64	ReLU	C4 @ 3 × 3 × 64	ReLU
8	FC2 @ 128	ReLU + dropout	P4 @ 2 × 2	–	P4 @ 2 × 2	–
9	FC3	Softmax	FC1 @ 512	ReLU + dropout	C5 @ 3 × 3 × 64	ReLU
10	–	–	FC2 @ 128	ReLU + dropout	P5 @ 2 × 2	–
11	–	–	FC3	Softmax	FC1 @ 512	ReLU + dropout
12	–	–	–	–	FC2 @ 128	ReLU + dropout
13	–	–	–	–	FC3	Softmax

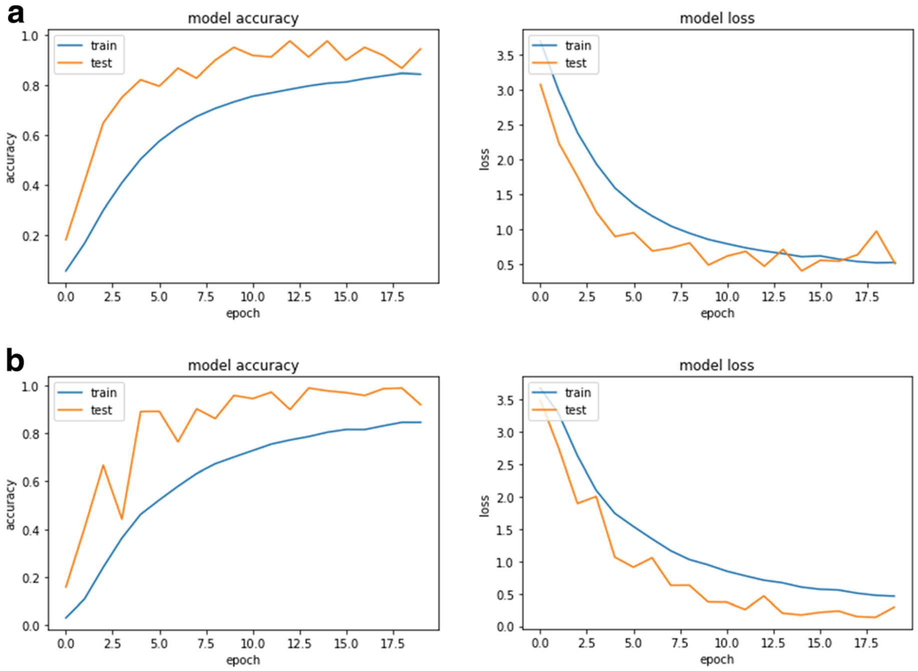

Fig. 4 Model accuracy and loss of proposed DCNN architecture, Arc-2. **a** EKFD [55]. **b** AT&T Database [56]

Face recognition using PCA [18] achieved accuracies of 89.0% and 91.0% on EKFD and AT&T databases respectively, whereas the MahCos PCA [19] achieved accuracies of 90.4% and 92.5% on the same databases. The proposed DCNN and B-DCNN methodologies outperformed both of these techniques comfortably. Lee et al. [58] achieved a recognition accuracy of 97.0% on EKFD as compared to 98.1% accuracy achieved by the proposed B-DCNN on the same database. The proposed B-DCNN achieved higher accuracy even though Lee et al. [58] used non-occluded images whereas the proposed methodology included both the occluded and non-occluded images which make the proposed method more robust to partial face images. The complexity of the proposed B-DCNN is much lower as only four layers were used as compared to 12 layers in the DCNN proposed by Lee et al. [58]. On the AT&T face database, Vinay et al. [59] achieved an accuracy of 95.2% compared to the 100% accuracy achieved by the proposed B-DCNN.

In order to show the effect of activation functions, further analysis has been made by utilizing two more activation functions namely LReLU [46] and ELU [47] in addition to ReLU [45]. Table 3 presents the comparison

Table 2 Summary of results

Method	Recognition accuracy (%)	
	EURECOM Kinect Face Database [55]	AT&T face database [56]
PCA [18]	89.0	91.0
MahCos PCA [19]	90.4	92.5
DCNN by Lee et al. [58]	97.0 [58]	–
DCNN by Vinay et al. [59]	–	95.2 [59]
Proposed DCNN		
Arc-1	90.4	87.5
Arc-2	94.2	97.5
Arc-3	92.3	92.5
Proposed B-DCNN		
Arc-1	94.2	92.5
Arc-2	*98.1*	*100*
Arc-3	96.2	97.5

The results of the best performing network architecture are shown in italic

Table 3 Comparison of different activation functions tested on proposed B-DCNN Arc-2 model on EKFD

Activation function	Time comparison		Accuracy comparison on EKFD (%)	
	Average training time (s/epoch)	Average prediction time (ms)	DCNN	B-DCNN
ReLU	96.8	2.94	94.2	98.1
LeakyReLU	98.1	3.21	94.2	98.1
ELU	99.1	3.38	92.3	94.2

results of these activation functions tested on the proposed architecture Arc-2 on EKFD [55]. The effect of activation functions is observed based on model training time, model accuracy, and average prediction time. The average prediction time is measured by predicting the same image 100 times using the trained feed-forward network. As it can be seen from Table 3, ReLU and Leaky-ReLU achieve similar testing accuracies, but ReLU performs slightly faster than Leaky-ReLU and ELU since it is less computationally expensive. ELU achieves lower accuracy since the network depth is four layers and ELU usually performs better for much deeper networks [47].

Figure 5a presents an example case where the conventional DCNN model incorrectly predicted a class with the softmax probability of 98.9%. The proposed B-DCNN model correctly classified the person with 74.6% probability and reduces the incorrect class probability to 14.5%. Samples of incorrectly predicted class by DCNN model are shown in Fig. 5b. The reason for misclassification by the DCNN model can be due to several misleading similarities between the images of two classes such as the similar color of clothes and spectacles being worn by both the persons.

The comparison results presented in this section have shown that the proposed DCNN and B-DCNN-based face recognition give highly accurate results in comparison with other methods presented in the literature. Furthermore, the proposed B-DCNN methodology has shown improvement in the recognition accuracy as compared to the DCNN methodology, which shows that the proposed B-DCNN can successfully exploit model uncertainty and reduce erroneous recognition.

6 Conclusion

Facial image recognition is one of the most challenging tasks in surveillance systems due to problems such as low quality of images and significant variance in pose, expression, illumination, and resolution. Although a number of face recognition algorithms have been proposed in the literature, face recognition in an unconstrained environment still presents low accuracy. Recently, deep convolutional neural network (DCNN)-based techniques have shown excellent results in face recognition by discovering intricate features in large data-sets. However, DCNN-based models struggle to suggest uncertainty in the prediction of the output class which can be useful to reduce false positives. In this study, Bayesian deep convolutional neural network (B-DCNN) is employed to represent model uncertainty to improve the accuracy of facial image recognition.

In this study, the B-DCNN architecture is implemented by employing dropout at both training and testing phases [41] to get the posterior distribution of class probabilities. The mean and variance of the class

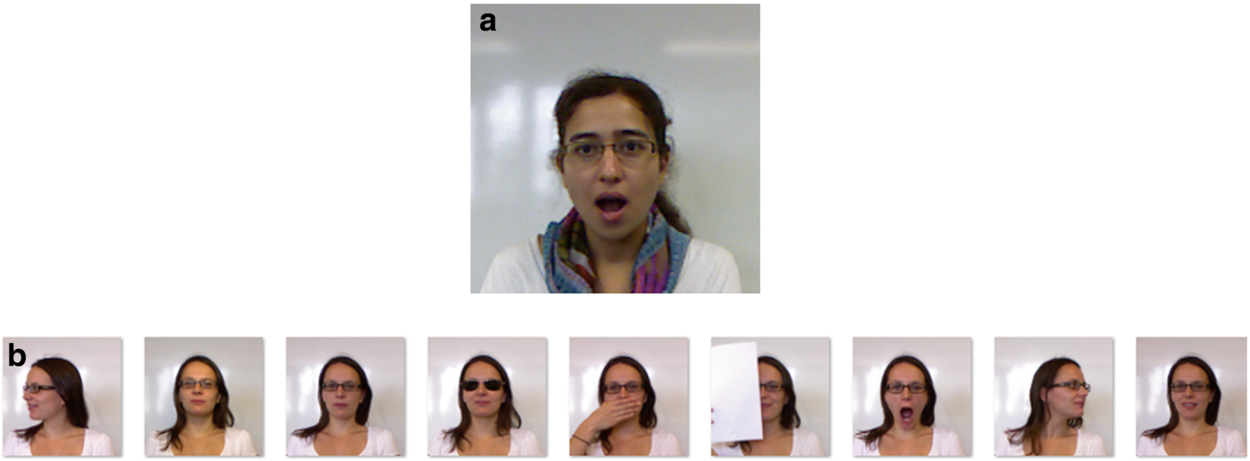

Fig. 5 Example case to show the significance of B-DCNN over DCNN. **a** Image "rgb_0031_s2_OpenMouth" from EKFD incorrectly classified by DCNN however correctly classified by the proposed B-DCNN. **b** Samples of incorrectly predicted class by the DCNN

probabilities are then used as confidence and uncertainty respectively for each class. The final classification decision is made by applying heuristic function. The experimentations are performed on two open-source databases: AT&T Face Database and EURECOM Kinect Face Database. The B-DCNNs are comparatively analyzed with DCNNs, and conventional machine learning approaches such as PCA and MahCos PCA are carried out. The results have demonstrated that the B-DCNN outperformed these methods and achieved an improvement of 3–4% in the accuracy of face recognition. In future, we intend to incorporate face-alignment for 3D face data and then apply B-DCNN for face recognition. We will observe how the alignment step affects the overall accuracy of 3D face recognition in extension to B-DCNN. Moreover, the proposed architecture can also be evaluated in terms of multi-scale/multi-view deep learning architectures for face data.

Abbreviations

B-DCNN: Bayesian deep convolutional neural network; DCNN: Deep convolutional neural network; DCP: Dual-cross pattern; DL: Deep learning; EKFD: EURECOM Kinect Face Database; ELU: Exponential linear unit; GEM: Generic elastic model; KL: Kullback-Leibler; LBP: Local binary patterns; LDA: Linear discriminant analysis; LReLU: Leaky rectified linear unit; MKD: Multi-keypoint descriptors; MRF: Markov random field; PCA: Principle component analysis; ReLU: Rectified linear unit; RFM: Rotated face model; SELU: Scaled exponential linear unit; SIFT: Scale-invariant feature transform

Acknowledgements

Not applicable

Funding

Not applicable

Availability of data and materials

The AT&T face database (formerly called ORL) [56] is available at http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html
The EURECOM Kinect Face Database (EKFD) [55] is available at http://rgb-d.eurecom.fr/

Authors' contributions

This work was carried out in collaboration between all authors. Authors UZ, MG, and TZ have designed the study and wrote the first draft of the manuscript and revised version. Authors UZ and GA carried out methodologies work, performed the thresholds settings, and obtained the results. Author AL led the literature searches and wrote related work. Also, he contributed sufficiently in improving the manuscript especially in the phase of manuscript revision. Authors KRM and AMS edited the manuscript. All authors read and approved the final manuscript.

Competing interests

The authors declare that they have no competing interests.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Author details

[1]Department of Computer Science, COMSATS University Islamabad, Islamabad, Pakistan. [2]University of Agriculture, Faisalabad, Pakistan. [3]Department of Computer Science and Engineering, Air University, Multan Campus, Multan, Pakistan. [4]University of Somalia, Mogadishu, Somalia.

Received: 30 May 2018 Accepted: 1 January 2019
Published online: 11 January 2019

References

1. M. Chihaoui, A. Elkefi, W. Bellil, C. Ben Amar, A survey of 2D face recognition techniques. Computers **5**, 21 (2016)
2. Y. Taigman, M. Yang, M.A. Ranzato, L. Wolf, Deepface: Closing the gap to human-level performance in face verification, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, (2014), pp. 1701
3. P.J. Grother, G.W. Quinn, P.J. Phillips, NIST Interagency Report, Report No. 7709, (2010)
4. M.A. Rahim, M.S. Azam, N. Hossain, M.R. Islam, Face recognition using local binary patterns (LBP). Global J. Comp. Sci. **13** (2013).
5. T. Ahonen, E. Rahtu, V. Ojansivu, J. Heikkila, Recognition of blurred faces using local phase quantization, in 19th International Conference on Pattern Recognition, (2008), pp. 1
6. G. Hua, M.-H. Yang, E. Learned-Miller, Y. Ma, M. Turk, D.J. Kriegman, T.S. Huang, Introduction to the special section on real-world face recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence. **33**, 1921 (2011).
7. M. Günther, L. El Shafey, S. Marcel, Face recognition across the imaging spectrum (Springer, 2016), pp. 247
8. E. Gumus, N. Kilic, A. Sertbas, O.N. Ucan, Evaluation of face recognition techniques using PCA, wavelets and SVM. Expert Systems with Applications. **37**, 6404 (2010)
9. F.S. Samaria, A.C. Harter, Parameterisation of a stochastic model for human face identification, in Proceedings of the Second IEEE Workshop on Applications of Computer Vision, (1994), pp. 138.
10. R. Patel, N. Rathod, A. Shah, Comparative analysis of face recognition approaches: a survey. Int. J. Comp. Appl. **57** (2012)
11. N.I. Ratyal, I.A. Taj, U.I. Bajwa, M. Sajid, 3D face recognition based on pose and expression invariant alignment. Comp. Elec Eng. **46**, 241 (2015)
12. Y. LeCun, Y. Bengio, G. Hinton, Deep learning. Nature. **521**, 436 (2015)
13. Y. Sun, D. Liang, X. Wang, X. Tang, Deepid3: Face recognition with very deep neural networks. arXiv preprint arXiv:1502.00873 (2015)
14. F. Schroff, D. Kalenichenko, J. Philbin, Facenet: A unified embedding for face recognition and clustering, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, (2015), pp. 815
15. G. Klambauer, T. Unterthiner, A. Mayr, S. Hochreiter, Self-normalizing neural networks, in Neural Information Processing Systems, (2017), pp. 971
16. C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, R. Fergus, Intriguing properties of neural networks. arXiv preprint arXiv:1312.6199 (2013)
17. S. Soltanpour, B. Boufama, Q.J. Wu, A survey of local feature methods for 3D face recognition. Pattern Recognition. **72**, 391 (2017)
18. M. Turk, A. Pentland, Eigenfaces for recognition. J. Cogn. Neurosc. **3**, 71 (1991)
19. U.I. Bajwa, I.A. Taj, M.W. Anwar, X. Wang, A multifaceted independent performance analysis of facial subspace recognition algorithms. PloS one. **8**, e56510 (2013)
20. P.N. Belhumeur, J.P. Hespanha, D.J. Kriegman, Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. IEEE Transactions on Pattern Analysis and Machine Intelligence. **19**, 711 (1997)
21. J. Yang, D. Zhang, A.F. Frangi, J.-y. Yang, Two-dimensional PCA: a new approach to appearance-based face representation and recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence. **26**, 131 (2004)
22. A. Nestor, D.C. Plaut, M. Behrmann, Feature-based face representations and image reconstruction from behavioral and neural data. Proceedings of the National Academy of Sciences. **113**, 416 (2016)
23. S. Biswas, G. Aggarwal, P.J. Flynn, K.W. Bowyer, Pose-robust recognition of low-resolution face images. IEEE Transactions on Pattern Analysis and Machine Intelligence. **35**, 3037 (2013)
24. M. Fischer, H.K. Ekenel, R. Stiefelhagen, Analysis of partial least squares for pose-invariant face recognition, in 2012 IEEE Fifth International Conference on Biometrics: Theory, Applications and Systems (BTAS), (2012), pp. 331
25. Z. Guo, L. Zhang, D. Zhang, X. Mou, Hierarchical multiscale LBP for face and palmprint recognition, in 17th IEEE International Conference on Image Processing (ICIP), (2010), pp. 4521
26. C. Ding, C. Xu, D. Tao, Multi-task pose-invariant face recognition. IEEE Transactions on Image Processing. **24**, 980 (2015)

27. A. Mian, M. Bennamoun, R. Owens, An efficient multimodal 2D-3D hybrid approach to automatic facerecognition. IEEE Transactions on Pattern Analysis and Machine Intelligence. **29**, 1927 (2007)
28. S. Liao, A.K. Jain, S.Z. Li, Partial face recognition: Alignment-free approach. IEEE Transactions on Pattern Analysis and Machine Intelligence. **35**, 1193 (2013)
29. S.R. Arashloo, J. Kittler, Energy normalization for pose-invariant face recognition based on MRF model image matching. IEEE Transactions on Pattern Analysis and Machine Intelligence. **33**, 1274 (2011)
30. T. Zhang, X. Li, R-Z Guo, Producing virtual face images for single sample face recognition. Optik- International Journal for Light Electron Optics. **125**, 5017 (2014)
31. X. Hu, W-x Yu, J. Yao, Multi-oriented 2DPCA for face recognition with one training face image per person. Journal of Computational Information Systems. **6**, 1563 (2010)
32. L. Li, Y. Peng, G. Qiu, Z. Sun, S. Liu, A survey of virtual sample generation technology for face recognition. Artificial Intelligence Review. **50**, 1 (2018)
33. D. Yi, Z. Lei, S. Li Z, Towards Pose Robust Face Recognition, in IEEE Conference on Computer Vision and Pattern Recognition, (2013)
34. U. Prabhu, J. Heo, M. Savvides, Unconstrained pose-invariant face recognition using 3D generic elastic models. IEEE Transactions on Pattern Analysis and Machine Intelligence. **33**, 1952 (2011)
35. F. Juefei-Xu, K. Luu, M. Savvides, Spartans: Single-Sample Periocular-Based Alignment-Robust Recognition Technique Applied to Non-Frontal Scenarios. IEEE Transactions on Image Processing. **24**, 4780 (2015)
36. X. Hu, S. Peng, W. Li, Z. Yang, Z. Li, Surveillance video face recognition with single sample per person based on 3D modeling and blurring. Neurocomputing. **235**, 46 (2017)
37. Y. Wen, K. Zhang, Z. Li, Y. Qiao, A Discriminative Feature Learning Approach for Deep Face Recognition, in European Conference on Computer Vision, (2016), pp. 499
38. Y. Sun, Y. Chen, X. Wang, X. Tang, Deep learning face representation by joint identification-verification, in Neural Information Processing Systems, (2014), pp. 1988
39. K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
40. C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov et al., in IEEE Conference on Computer Vision and Pattern Recognition, (2015), pp. 1
41. Y. Gal, Z. Ghahramani, Dropout as a Bayesian approximation: Representing model uncertainty in deep learning, in International Conference on Machine Learning, (2016), pp. 1050
42. A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, in Neural Information Processing Systems, (2012)
43. N. M. Nasrabadi, Pattern recognition and machine learning. Journal of Electronic Imaging 16 (2007)
44. C.M. Bishop, Pattern Recognition and Machine Learning (Springer, 2006)
45. V Nair, GE Hinton, Rectified linear units improve restricted boltzmann machines, in Proceedings of the 27th International Conference on Machine Learning, (2010), pp. 807
46. A.L. Maas, A.Y. Hannun, A.Y. Ng, Rectifier nonlinearities improve neural network acoustic models, in International Conference on Machine Learning, (2013)
47. D.A. Clevert, T. Unterthiner, S. Hochreiter, Fast and accurate deep network learning by exponential linear units (elus). arXiv preprint arXiv:1511.07289 (2015)
48. X. Zhu, J. Lafferty, Z. Ghahramani, in ICML 2003 Workshop on the Continuum from Labeled to Unlabeled Data in Machine Learning and Data Mining, (2003)
49. X. Li, Y. Guo, Adaptive active learning for image classification, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, (2013), pp. 859
50. A.J. Joshi, F. Porikli, N. Papanikolopoulos, Multi-class active learning for image classification, in Computer Vision and Pattern Recognition, (2009), pp. 2372
51. Y. Gal, Z. Ghahramani, Bayesian convolutional neural networks with Bernoulli approximate variational inference. arXiv preprint arXiv:1506.02158 (2015)
52. G.E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, R.R. Salakhutdinov, Improving neural networks by preventing co-adaptation of feature detectors. arXiv preprint arXiv:1207.0580 (2012)
53. N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, Dropout: A simple way to prevent neural networks from overfitting. J. Mac. Learn. Res. **15**, 1929 (2014)
54. Y. Gal, Dissertation, University of Cambridge, 2016
55. R. Min, N. Kose, J.L. Dugelay, Kinectfacedb: A kinect database for face recognition. IEEE Transactions on Systems, Man, and Cybernetics: Systems. **44**, 1534 (2014)
56. The AT&T Database of Faces, http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html. (Accessed 24 Oct 2018)
57. M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean et al., Tensorflow: a system for large-scale machine learning, in Operating Systems Design and Implementation (OSDI), (2016), pp. 265
58. Y.C. Lee, J. Chen, C.W. Tseng, S.H. Lai, Accurate and robust face recognition from RGB-D images with a deep learning approach, in British Machine Vision Conference (BMVC), (2016)
59. A. Vinay, D.N. Reddy, A.C. Sharma, S. Daksha, N.S. Bhargav, M.K. Kiran et al., G-CNN and F-CNN: Two CNN based architectures for face recognition, in International Conference on Big Data Analytics and Computational Intelligence (ICBDAC), (2017), pp. 23