Commun.Fac.Sci.Univ.Ank.Series A2-A3
Volume 61, Number 2, Pages 129-149 (2019)
DOI: 10.33769/aupse.529575
ISSN 1303-6009 E-ISSN 2618-6462
http://communications.science.ankara.edu.tr/index.php?series=A2A3

A COMPARISON OF DEEP LEARNING BASED ARCHITECTURE WITH A CONVENTIONAL APPROACH FOR FACE RECOGNITION PROBLEM

Fatima Zehra UNAL and Mehmet Serdar GUZEL

ABSTRACT. This paper addresses a new approach for face recognition problem based on deep learning strategy. In order to verify the performance of the proposed approach, it is compared with a conventional face recognition method by using various comprehensive datasets. The conventional approach employs Histogram of Gradient (HOG) algorithm to extract features and utilizes a multi-class Support Vector Machine (SVM) classifier to train and learn the classification. On the other hand, the proposed deep learning based approaches employ a Convolutional Neural Network (CNN) based architecture and also offer both a SVM and Softmax classifiers respectively for the classification phase. Results reveal that the proposed deep learning architecture using Softmax classifier outperform conventional method by a substantial margin. As well as, the deep learning architecture using Softmax classifier also outperform SVM in almost all cases.

1. INTRODUCTION

Face recognition, covering a large number of fields and disciplines such as safety and commercial applications, is an important research problem and gathers lots of attention from researchers. Deep Learning technology has dominated the field and made great progress in solving problems that have not been achieved with applications developed for a long time in artificial intelligence field. It is clear that deep learning and related technologies have improved the overall success performance of many classification problems in the field of computer vision. Especially, the adaptation of Convolutional Neural Networks (CNN) architecture to

Received by the editors: February 20, 2019; Accepted: June 15, 2019.
Key word and phrases: Convolutional neural network, deep learning, face recognition, fine tuning, softmax, svm.

computer vision problems has opened a new era at the field that achieved promising results in computer vision based applications.

Face recognition applications can be categorized into two main groups namely, security and commercial applications. A good example for security applications is real-time mapping according to video image sequences that employs previously recorded images to detect and recognize criminals and prevent unauthorized people entering restricted zones. Static mapping from credit card images, passports, driver's license and identity cards is a good example for commercial applications. It essentially provides real time transactions based on image or video sequences [1]. Face recognition applications have become enormously significant since they have offered successful results in the field of security. Essentially for such security systems, Machine learning algorithms play a crucial role, especially in recognition and verification tasks. Artificially recognizing people faces can be performed through supervised learning mechanisms by employing predefined features for training. However, this learning technique can only be successfully applied when the faces are captured in well-defined conditions. On the other hand, recognition becomes quite difficult when an irrepressible situation occurs, such as changes in the face expression or head directions, as well as lighting conditions is also crucial. The only way to overcome these problems is to employ a reliable feature extraction algorithm comprising consistent enhancement and restoration steps [2]. Deep learning based algorithms can be able to learn automatically to extract the needed properties to train a new classifier to be used to solve a different problem.

Deep Learning Technologies accomplished great progress in solving problems that have not been achieved with applications developed in the field of machine learning for many years. Deep learning technologies are able to explore the complex structures of high-dimensional data that has been applied in many areas from image and speech processing to classification and regression problems [3,4,5]. The processing power provided by the graphics processing unit (GPU) allows Deep Learning technologies to employ extensive amounts of data to train deeper or more advanced models with respect to the increased processing power.

Deep learning is essentially a multi-layer artificial neural network-based machine learning technique. The main advantage of deep learning is that layers of features are not obtained from conventional feature extractor algorithms, instead they are learned automatically from data using multilayer network hierarchy [3, 6]. The higher layers of a deep learning architecture strengthen the characteristic properties of the input given to the network and defeats the irrelevant properties. For example, it is assumed that an image consists of pixel values and simple shapes is employed as input to the deep learning architecture. The extracted properties in the first network layer most probably signify the presence or absence of edges in certain directions and positions in the image. The second layer, on the other hand, identifies patterns by detecting special arrangements of edges by considering negligible minor changes in edge locations. The third layer can estimate larger combinations or more complex patterns of familiar objects, and essentially consequent layers may perceive concrete objects as combinations of these parts. Consequently, the network first learns the raw primitive edges, followed by learning more complex shapes based on the edges it has previously learned, and learns more advanced features using those shapes. This hierarchical structure allows the architecture to extract features in a systematic manner. Deep learning models have the ability to learn to focus on the right features automatically, and therefore require little guidance from the designer to intervene feature extraction process [3].

This paper, in essence, proposes a CNN architecture for a better understanding of deep learning based face recognition models. For the first architecture, the pertained AlexNet is used as feature extractor and supported by SVM classifier for face recognition problem. For the second architecture, pretrained AlexNet with fine-tuning is used for face recognition problem. In order to reveal the performance of the proposed architectures, those have been compared with a conventional face recognition system using HOG algorithm for feature extraction process and SVM classifier for data training step. Three comprehensive dataset are employed to evaluate those systems in a reliable manner. Overall, section 2 details the corresponding literature of the problem whereas section 3 details the proposed deep learning based architectures for face recognition problem. Section 4, on the other hand illustrates the experimental configuration and results. Finally, the paper is concluded at Section 5.

2. Literature review

Deep learning based architectures have recently dominated the field. This section includes some relevant studies that aims to help reader to follow the state of the art technologies. For instance, in a speech recognition study it is aimed to train large scale neural network-based speech models in large data sets. English Broadcast News has been trained on 400M symbols in this speech recognition task and the test results verify the overall accuracy the system within a small word based error rate

[7]. Another study proposes a natural processing system including, speech tagging, division, entity recognition, etc. with high speed and precision results. The critical issue within this multilayer network architecture that it does need optimized labelled data but employ unlabeled training data [8].

A deep learning based object classification system was presented in a contest that offers a deep convolutional neural network architecture called AlexNet. The systems employed more than 1 million high-resolution images and aimed to classify them into 1000 different categories and better results have been obtained from the previous technology. To reduce overfitting in fully connected layers, the recently developed method of normalization called "Dropout" has been used and proved to be very effective [9]. An outstanding study also employs deep learning approach for scene segmentation and labelling [10]. It mainly performs full scene labeling, also known as scene parsing that comprises labeling the category of the object that each pixel in the image belongs to. Once this is accomplished, every object is identified and labeled successfully. Markov Random Field model was integrated into the Deep Convolution Neural network architecture for the human exposure estimation system in molecular images [11].

Gaining the ability to machines to answer questions automatically is a crucial problem of artificial intelligence community. For this problem, an embedded system using deep learning technology was designed. This system is able to answers questions on a wide range of topics (5,810 question-answer pairs are used for training) from a knowledge base using a small number of engineering features [12].

DeepID network architecture have developed for predicting top-level facial features using a deep convolution neural network and including a formal class of 10,000 classes with the help of these features. These features have been shown to be effective in recognizing new faces that do not appear in face verification and training set. The network is trained to classify all faces in the training set according to their identities [13]. The DeepID network architecture utilizes 4 convolutional layers and a pooling layer, allowing hierarchical extraction of the features. For classification, The SoftMax output layer is involved. The developed system was trained by LFW dataset and it is claimed to have a success rate of 97.45% [13]. Alternatively, a face recognition system was developed based on DeepFace's deep learning technology so as to capture human-level performance in verification applications. The deep network contains more than 120 million parameters from standard convolutional layers. With the developed method, it is claimed to reach 97.35% performance ratio with LFW dataset. [14]. In 2015, a system called FaceNet using deep convolutional neural network for face recognition and clustering applications was proposed. FaceNet learns a direct mapping from the facial image to the Euclidean space based on the direct facial similarity measure. After the Euclidean space has been obtained, primary tasks namely, face recognition, validation and clustering can be implemented using the techniques within the FaceNet system. The developed system was tested with LFW and Youtube Faces datasets and authors declared to reach success rates of 99.63% and 95.12% correspondingly [15]. One the major challenges for face recognition problem is to exctract effective features to reduce personal changes while increasing interpersonal differences. As well as, complexity and scalability of face recognition problem is also an important challenge, corresponding papers can be seen in [16, 17]. WebFace [18] called CASIA WebFace dataset which contains about 10,000 subjects and 500,000 face images is built by collecting a semi-automatically from internet. 11 layer CNN is used to learn discriminative representation and obtain accuracy on LFW and YTF based on WebFace. This study’s aim is to create large scale public database for face recognition problems. VGG-Face [19] is a deep CNN model, consisting of 16 layers, was created with 2.6 million pictures of 2,600 people. [20] propose a novel deep architecture for person re-identification. They introduce two novel layers namely a cross-input neighborhood differences layer, and a subsequent layer. The architecture is conducted on CUHK03 data set and CUHK01 dataset. Their results comparable to the state of the art [20]. DeepID3[21] consist of two deeper neural architectures for face recognition. The networks achive the state of the art performance on LFW 99.53% for face verification accuracy and 96.0% for identification accuracy. Inception, a convolutional neural network is proposed by [22]. The main characteristic of this architecture is the improved utilization of the computing resources inside the network. GoogLeNet is the embodiment of Inception for ILSVRC14 contest and consist of 22-layer network for classification and detection. [22] Deep Pyramid Feature Learning (DPFL) model [23] is presented to extract multi-scale appearance features for person re-identification. Unlike the current methods, the proposed model is able to extract prominent scale-specific features by jointly learning multiple scales of person images by training CNN model. Model conducted on three databases namely Market-1501, CUHK03, and DukeMTMC-reID. Domain Guided algorithm [24] is proposed to improve the generic and robust feature learning procedure for person re-identification. Algorithm provides promising results.

3. Methodology

In this section conventional face recognition method and deep learning based face recognition architectures are detailed respectively and also, the convolution neural network layers are generally described for a better understanding of architectures. One of the main contribution of this paper to compare success rates of conventional machine learning and deep learning techniques for face recognition problem. Besides, performance of different classifiers, integrated into the deep learning based architecture, are compared using comprehensive face recognition dataset.

3.1 CONVENTIONAL FACE RECOGNITION METHOD

The general face recognition algorithm has the following logic: system consists of two parts, namely the registration and the recognition phases. During the registration process; the system is trained using thousands of face image data and a trained model is created. On the other hand, in recognition process, the facial features extracted from the test image are compared with facial features stored in the database to perform recognition.

Our conventional face recognition model consists of four main parts: pre-processing, face detection, feature extraction and classification. Viola & Jones algorithm [25], HOG [26] and SVM [27] are used for face detection, feature extraction and classification stages, respectively. The proposed system performs detection, alignment and recognition processes successfully. Figure 1 demonstrates the steps of processing the conventional face recognition model. Detailed explanations of the stages and methods used are made in the following subsections.

3.1.1 Pre-Processing and Face Detection Stage

The pre-processing step is important in order to remove the variation effects in the images such as illumination, expression, occlusion, the background of the image and the like. The face recognition performance will improve as the features not related to the facial image to be processed decrease. The face detection module can be considered as the most critical part of the overall framework, and it basically detects and extracts faces from the input images.

The presence of the face and the calculation of the corresponding region have been performed and images are subjected to pre-processing to come from the noise, lighting, pose / turn problems. For detecting phases, the conventional Viola & Jones algorithm is used. It is a commonly used real-time object detection method. The algorithm has a very high rate of object detection, and instead of scaling the image itself, it scales the properties. It consists of four stages, namely haar features, integral image, adaboost training and cascading classifier. The details can be seen in [25]. The Haar feature classifier consists cascading trained strong classifier, which is based on the adaboost algorithm by Viola and Jones [28].

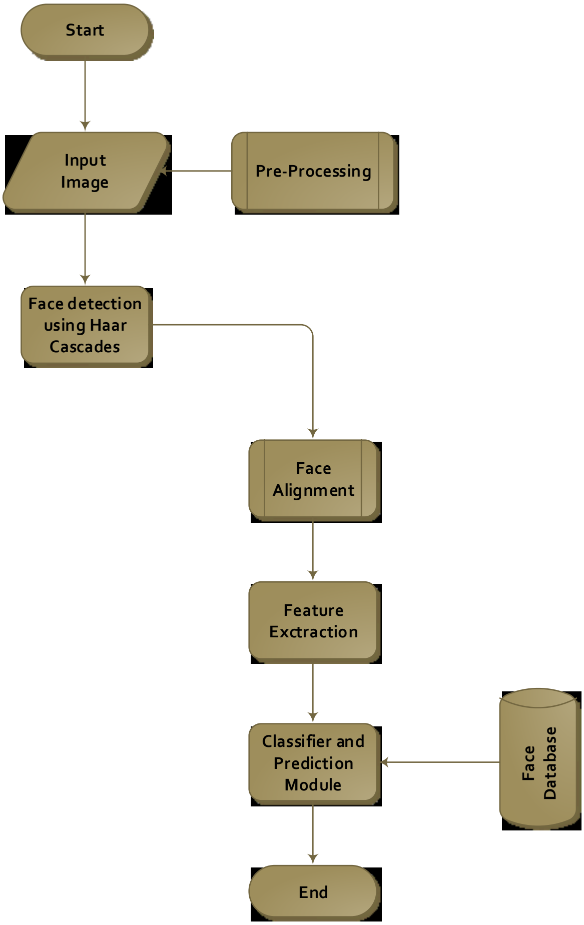


FIGURE 1. Conventional Face Recognition Architecture.

In our method; the original image is taken as input, the image is converted to gray color, and the face is detected using the Viola & Jones algorithm. The detected faces are cropped and resized to 100 × 100 pixels. The images in the dataset are divided into 60% -40%, 70% -30%, 80% -20%, 90% -10% ratio rate for training and test data, respectively.

3.1.2 Face Alignment and Feature Extraction Stage

Face alignment is an intermediate process required before the face recognition module and is crucial for better recognition results. Extraction and localization of facial landmarks are the critical issues in this process. Facial feature extraction and localization problems have been addressed by algorithms combining shape and texture modelling. Correspondingly, in this study, a shape model-based approach was utilized [29]. The algorithm proposes a shape constraint technique basically employing a multi-stage algorithm to automatically locate facial features. The individual face detectors are combined and applied to the facial images in order to predict facial landmarks.

Feature extraction is the central process for the success of the face recognition algorithm [29]. Histogram of Gradients (HOG) is used in our conventional face recognition method as feature extractor. HOG algorithm which is a powerful descriptor for object recognition and particularly in face recognition was proposed by Dalal et al. [26] for goal of human detection. For feature extraction using HOG method; the images are divided into small bounded regions, called cells, and for each cell a histogram of edge orientations is obtained. The histogram counts are normalized to recover the illumination. The final HOG descriptor is represented by combination of these histograms [30]. For obtaining hog feature, gradient calculation, gradient vote calculation and normalization calculation are combining. The formulas necessary to extract the HOG features of an image are as follows.

$$f_x(x,y) = I(x+1,y) - I(x-1,y) \tag{1}$$

$$f_y(x,y) = I(x,y+1) - I(x,y-1) \tag{2}$$

For each pixel (x,y) horizontal and vertical gradients are calculated by equation (1) and (2).

$$|m| = \sqrt{f_x(x,y)^2 + f_y(x,y)^2} \tag{3}$$

$$\varphi = arctan\frac{f_x(x,y)}{f_y(x,y)} \tag{4}$$

For each pixel, gradient magnitude and gradient direction are calculated by equation (3) and (4). When the HOG algorithm is applied, it is possible to obtain more understandable results by grouping the orientations of the pixels in the generated histogram. This grouping is possible by drawing the angular values in the range 0-360 to a desired range [31]. In our method; 9 groups of 20 degrees were created for directions between 0-180 degrees (20 degrees in each zone). In the last step HOG feature vector is obtained for faces after normalization by equation (5).

$$f = \frac{v_k}{\sqrt{||v_k|| + \varepsilon^2}} \tag{5}$$

where $\boldsymbol{\varepsilon}$ refers constants, v_k is the normalized histogram vector obtained from a block, and f is the HOG feature vector. After these calculations, 4356 dimensional feature vector obtained from 100×100 pixel of size image.

3.1.3 Classification Stage

In our conventional model, multi-class SVM method, one vs all approach is used for classification of faces. Face recognition is k class problem that k is the number of each individual [32]. The input for the SVM is obtained as the result of the feature extraction process by HOG descriptor. Support Vector Machines are popular supervised learning models, developed for the solution of classification and regression analysis problems by Vapnik et al. [27]. The SVMs aim to find the best hyperplane to maximize the distance between support vectors of different classes. SVMs are able to solve multi-class classification problems [33]. There are two approaches used in SVMs to solve multi-class problems. In the proposed method, one vs all approach is employed that a number of SVMs are trained for a number of classes. Class is the number of each individual. Each SVM separates a single class from all remaining classes [34]. For instance, the data from the nth class is trained as a positive example with the n binary classifier, while the remaining (k-1) class is trained as a negative example. During testing, the class label is determined by the binary classifier giving the highest output value [35]. The mathematical equations of model is given below (6-8):

Consider a M class problem and N training samples: $\{x_1,y_1\}, \dots, \{x_\mathrm{N},y_\mathrm{N}\}$ $x_i \in \mathbb{R}^m$ is an m-dimensional feature vector representing the ith training sample and $y_i \in$

$\{1,2,\dots,\mathrm{M}\}$ is the class label of x_i. A hyperplane in the feature space can be described by the equation:

$$f_i(x) = w_i^{\mathrm{T}}\Phi(x) + b_i w \tag{6}$$

Where w, is the weight, b is the bias and scalar value. Φ, is the feature vector in the multidimensional extension of the input vector x.

$$L(w_i,\xi_j^i) = \frac{1}{2}||w_i||^2 + C\sum_{j=1}^{N}\xi_j^i \tag{7}$$

Where ξ is slack variable which is relevant to the soft margin. The tuning parameter, C>0, which is applied to balance the weight of the margin and the training error. L, is the error penalty. Lagrange multipliers is used to solve the optimization problems which transform them to quadratic programming problems.

$$i^* = \underset{i=1,\dots,M}{argmax} f_i(x) = \underset{i=1,\dots,M}{argmax}(w_i^T\Phi(x) + b_i) \tag{8}$$

At the classification phase, x is classified for the maximum value $i\,^*$provided by f_i.The result of the classifier is the output of this argmax function.

3.2 DEEP LEARNING BASED FACE RECOGNITION MODEL

In this subsection, the CNN layer's working logics are generally described for a better understanding of proposed architectures of our study. Deep learning, in particular CNN's are generally trainable multi-layered architecture designed to learn the unchanging features of the neural network which is inspired by the biological neuron. CNN learns end-to-end training features in a hierarchical manner due to the characteristics of multi-layered architecture [6-10]. The architecture of convolutional neural networks consists of three basic layers such as convolution layer, a pooling layer, and a fully connected layer that follow the input layer.

A. Convolutional Layer

The convolution layer is the basis of the convolution neural network and its primary task is to extract the properties of the input image. In this layer; each filter (kernel or neuron) detects a different feature on the input. A different filter is applied to each convolution layer and these results are combined for feature extraction. Essentially features from primitive to advance are obtained hierarchically. The output of a filter of the previous layer is the input of the filter in a next layer and contains features of the previous layer. The feature is learned by scanning the filter at a certain size that the network learns its values through filtering.

The original image pixel value and filter values are multiplied together to obtain a single value. This process is repeated for every position in the input volume. Finally, the values obtained are called feature map (activation map). In essence each feature map signifies a certain feature at the output layer. Each filter creates different features of the view. The mathematical description of convolution operation is illustrated in equation 9 as follows [6]:

$$y_j = b_j + \sum_{i}^{n} x_j \ * \left(k_{ij}\right) \tag{9}$$

where, y_j: Input activation map; x_j: Input activation map ; b_j: Bias parameter; k_{ij}:Trainable filter.

Convolutional layers are usually followed by a nonlinear activation function, activation layer. There are many activation functions like sigmoid, tanh, maxout, Relu (Rectified Linear Unit), leakly Relu, elu,etc. Despite the tanh and sigmoid functions are also preferred as activation functions, the ReLU has superiority over them due to its efficient calculations characteristics. This essentially accelerates the training period of the network and allows faster convergence process. An example ReLU function $y = \max(0, x)$ is illustrated in the Figure 2 below. In other words, the activation is simply thresholded at zero when x<0.

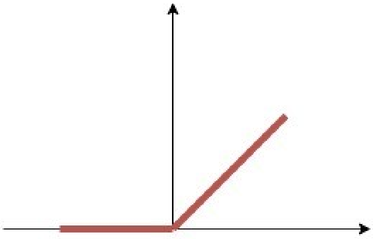


FIGURE 2. The ReLU function.

No parameters are learned in this layer. The aim of this layer is provide nonlinearity to the systems, applying linear operations through convolution layers [36].

B. Pooling Layer

Convolution and ReLU layers are usually followed by the Pooling layer. The main purpose of the pooling layer is to gradually reduce the spatial dimension of the input image which reduces the computational complexity of the model and therefore controls overfitting. A pool operator including max, average, sum is applied to the feature map obtained from the previous layer. The pooling operator returns a value for each filter [37].

C. Fully Connected Layer

The fully connected layer is usually employed at the end of the convolution and pooling layers. This layer resembles the conventional neural networks that each pixel is considered to be a separate neuron and contains as many neurons as the number of expectable classes. Output obtained as a result of convolution, ReLU and pooling layers; contains distributed features of the input image. The purpose of this layer is to use all of these properties to create properties with strong capabilities in the next stage. Accordingly, this layer classifies the input image using top-level properties that come from the previous layers. Besides, this layer allows to learn non-linear combinations of top-level features.

In the following subsections, CNN based face recognition architectures relying on two different classifiers are explained respectively.

3.2.1 CNN based Face Recognition Architecture using SVM

CNN based face recognition architecture “relying on a SVM classifier” has been employed to overcome face recognition problem. CNN, which can determine which parts of a face should be measured, is used in the step of extracting the distinguishing features from the images. For this purpose; a modified version of AlexNet convolutional neural network is used for feature extraction stage and a multi-class SVM is used classification stage. The proposed CNN based Face Recognition Architecture is illustrated in Figure 3.

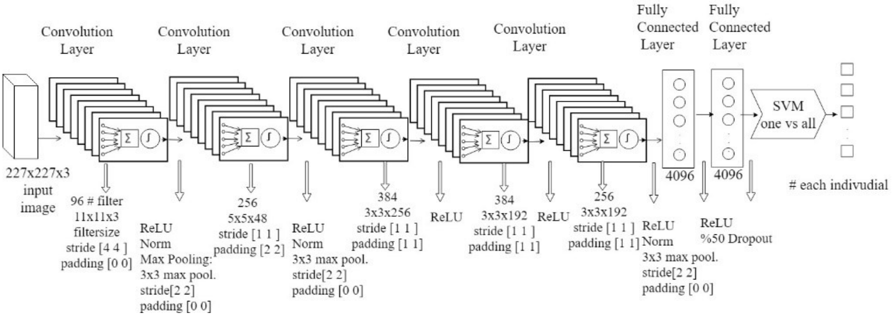

Convolution Layer
Convolution Layer
Convolution Layer
Convolution Layer
Convolution Layer
Fully Connected Layer
Fully Connected Layer
Σ
∫
SVM one vs all
227x227x3 input image
96 # filter 11x11x3 filtersize stride [4 4] padding [0 0]
ReLU Norm Max Pooling: 3x3 max pool. stride[2 2] padding [0 0]
256 5x5x48 stride [1 1] padding [2 2]
ReLU Norm 3x3 max pool. stride[2 2] padding [0 0]
384 3x3x256 stride [1 1] padding [1 1]
ReLU
384 3x3x192 stride [1 1] padding [1 1]
ReLU
256 3x3x192 stride [1 1] padding [1 1]
ReLU Norm 3x3 max pool. stride[2 2] padding [0 0]
4096
4096
ReLU %50 Dropout
# each indivudial

FIGURE 3. The CNN based face recognition architecture using SVM

AlexNet is a deep convolutional neural network was proposed by Krizhevsky et al. [9] to classify the 1.2 million labelled images in the ImageNet into the 1000 different classes. The architecture consists of five convolution layers with weights and decreasing filter size; followed by some of the pooling layers and 3 fully connected layers. One of the main characteristics of the AlexNet is the speed of downsampling of the intermediate representations through following convolutions and max-pooling layers. The final convolutional activation map is formed as a vector and send as an input to following two fully connected layers of 4096 units in size. The image descriptor produced by AlexNet is represented by the output of this layer [38].

The steps of the face recognition application performed by the deep learning method are detailed as follows:

- In the pre-processing stage; there are some modifications. Since the AlexNet network is trained on 227x227 pixel and colour images, all images are resized to 227x227 pixel. The network requires 3 channel input. The dataset with grey colour images (1 channel) are converted to RGB by the other two channels are simply copied and a three channel image is obtained [39].
- In the face detection stage; faces are detected by Viola & Jones algorithm.
- The images in the dataset are divided into 60% -40%, 70% -30%, 80% -20%, 90% -10% ratio for training and test data, respectively, and randomly selected.
- The features of the images in the training set are extracted by CNN, with pre-trained AlexNet architecture, and these properties are used to train and test the multi class SVM classifier as illustrated in Figure 3. The first layers

of the network extract basic image features such as edges, corners etc. These basic features are then processed by deeper layers of the network to produce higher-level image features. This higher-level feature is more suitable for classification. Because they combine all basic features with a richer image presentation. While each layer of a CNN generates a response to an input image only a few layers are suitable for feature extraction.

- For AlexNet architecture the final layer is for the classification problem. AlexNet has been trained to classify a 1000-class problem that is not suitable for the datasets have used in this study. Therefore, the classification layer of this network is not utilized, whereas the features (4096 dimension feature vector) obtained from the second fully connected layer of this network is used to train a multi-class SVM classifier (illustrated in figure3).
- The outputs of last fully connected layer are used as input for training the one vs all design SVM classifier (working logic is detailed in Section 3.1). Test images are classified by using trained SVM. The actual result is compared with the classifier's prediction. The operation is performed correctly if the two results are the same.

3.2.2 CNN based Face Recognition Architecture using Softmax

Training a deep network from scratch which means optimizing millions of parameters to learn weights requires a large amount of data, computational and memory resources for training stage [40,41]. Therefore, fine tuning a network with transfer learning is an alternative method and frequently used in deep learning applications. The second architecture is essentially tuning the pre-trained AlexNet network weights by backpropagation algorithm and the Softmax classifier, which is a probabilistic approach minimizes cross-entropy between the appraised class probabilities and the “true” distribution. The architecture of this model is also illustrated in Figure 4.

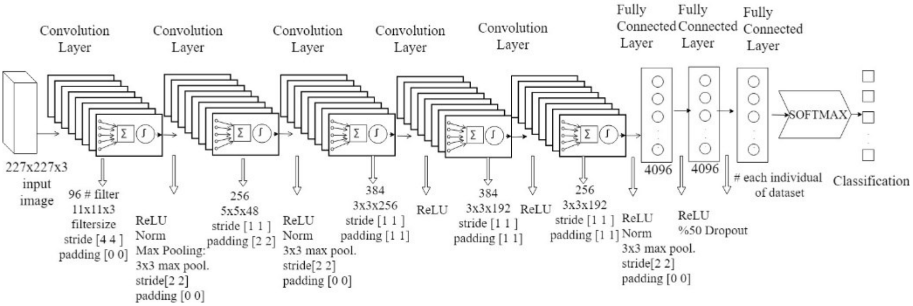

Convolution Layer
Convolution Layer
Convolution Layer
Convolution Layer
Convolution Layer
Fully Connected Layer
Fully Connected Layer
Fully Connected Layer
SOFTMAX
227x227x3 input image
96 # filter 11x11x3 filtersize stride [4 4] padding [0 0]
ReLU Norm Max Pooling: 3x3 max pool. stride[2 2] padding [0 0]
256 5x5x48 stride [1 1] padding [2 2]
ReLU Norm 3x3 max pool. stride[2 2] padding [0 0]
384 3x3x256 stride [1 1] padding [1 1]
ReLU
384 3x3x192 stride [1 1] padding [1 1]
ReLU
256 3x3x192 stride [1 1] padding [1 1]
ReLU Norm 3x3 max pool. stride[2 2] padding [0 0]
4096
4096
ReLU %50 Dropout
# each individual of dataset
Classification

FIGURE 4. Fine tuning pre-trained network architecture

The steps of the face recognition application performed by a pre-trained model are detailed as follows:

- In the pre-processing and face detection steps are the same as the CNN based face recognition architecture. (see section 3.1)
- The images in the dataset are divided into 60% - 40%, 70% - 30%, 80% - 20%, 90% - 10% ratio for training and test data sets respectively, which are randomly selected as well.
- The last three layers of the architecture represent fully connected layers and also a softmax layer is added as the classifier layer, providing a probabilistic approach. And a classification layer of AlexNet are replaced for our task. For this purpose; earlier layers of AlexNet is fixed; last three layers are fine tuned for face recognition task. Previous layers of network is also fixed that those layers contain more general features. However following layers are more powerful in terms of extracting precise to the details of problem. The last fully connected layer's options are specified according to our datasets such as setting same class number of our dataset. Training options are specified as following: epoch:20; validation frequency: 3 iterations; minibatchsize:32; initial learning rate: 1e-4; hardware resource: single GPU; learning rate schedule: constant; learning rate: 0.0001.
- After obtaining features by fine tuning, the softmax classifier is trained using back propagation algorithm with our datasets.

The predicted class is:

$$i = \underset{i}{argmax} a_i \tag{11}$$

- Softmax classifier is the binary Logistic Regression classifier's generalization to multiple classes. It takes a vector which consists real-valued scores in given class and normalizes the values between zero and one that sum to one.
- Softmax classifier is calculated by equation 12.

$$P(c_r|x) = \frac{P(x|c_r)\,P(c_r)}{\sum_{j=1}^{k} P(x|c_j)\,P(c_j)} = \frac{e^{a_r}}{\sum_{j=1}^{k} e^{a_j}} \tag{12}$$

where $a_r = \ln P(x|c_r)\,P(c_r), P(x|c_r)$ is the conditional probability of the sample given class r, and P(c_r) is the class prior probability.

4. Experimental configuration and results

In this section, data sets used in our experiments are introduced and the performances of deep and conventional face recognition methods are evaluated. All experiments are conducted on MATLAB 2018a and a desktop computer with the following specifications: Intel i7 7700K 4.20 Ghz CPU, Nvdia GeForce 1080 GPU, 16 GB RAM.

4.1 DATABASES

For the experimental section, 3 popular and comprehensive datasets, namely, AT&T [42], faces95 and faces96 [43] are employed to train and compare the performance of the conventional and deep learning based methods.

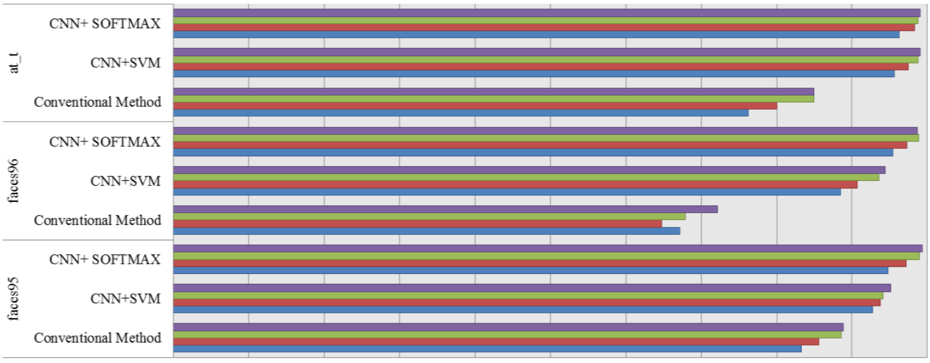
The AT&T face database contains 400 images of 40 different person. The size of each image is 92x112 pixels with 256 grey levels per pixel.

The faces95 dataset contains a total of 1440 colored images of 72 different individuals. The size of each image is 180x200 pixels.

The faces96 dataset contains a total of 3040 colored images of 152 different individuals. The size of each image is 196x196 pixels.

4.2 RESULTS

In this subsection, face recognition performance rate of conventional and deep models are examined and obtained results are illustrated. As it defines the efficiency of the algorithm, evaluating the performance of deep learning methods is as important as the algorithm itself [44].



	faces95			faces96			at_t		
	Conventional Method	CNN+SVM	CNN+ SOFTMAX	Conventional Method	CNN+SVM	CNN+ SOFTMAX	Conventional Method	CNN+SVM	CNN+ SOFTMAX
🟪 9/10 train set	88,89%	95,15%	99,31%	72,19%	94,37%	98,68%	85,00%	99,06%	99,06%
🟩 8/10 train set	88,54%	94,10%	98,96%	67,88%	93,54%	98,84%	85,00%	98,75%	98,75%
🟥 7/10 train set	85,64%	93,75%	97,22%	64,79%	90,73%	97,24%	80,00%	97,50%	98,33%
🟦 6/10 train set	83,30%	92,71%	94,79%	67,14%	88,49%	95,45%	76,25%	95,63%	96,25%

FIGURE 3. Conventional & Deep Learning based architectures comparison results.

Figure 5 illustrates the performance comparison of used architectures based on three different datasets with varying training dataset rate. It has been observed that the face recognition accuracy for all data sets increases correctly with the amount of data in the training data set.

Results reveal that Deep learning based architecture outperform the conventional one for all datasets even if the training dataset decreases. For all used datasets, deep learning based architecture achieves more than %90 success rate. In this context, it has been proven that self-learning ability, which distinguishes deep learning from other methods in the process of feature extraction, is successful. As can be seen from the figure; face recognition performance of fine tuning pre trained AlexNet model is better than CNN+SVM deep model especially for faces95 and faces96 databases.

5. Conclusion

This paper proposes a deep learning based architectures for the face recognition problem. In order to reveal the efficiency and accuracy of the proposed system, it is compared with one of the most reliable conventional architecture. This architecture extracts feature using HOG algorithm whereas the classification is achieved by designing a one vs all multi-class SVM.

The proposed deep learning based architectures, on the other hand, utilizes a popular pre-trained Convolutional neural network architecture, called as AlexNet. This architecture is modified and adapted into the proposed face recognition system that an SVM classifier is integrated into the system for the classification phase.

As a second approach, the same architecture, utilizing fine tuning pre trained AlexNet model, integrated Softmax classifier instead of SVM for face recognition problem. Overall, these three architectures are compared by utilizing three standard datasets, namely, faces95, faces96 and at_t, designed for face recognition problem. Results verify the superiority of the deep learning based architecture over the conventional architecture. Also results indicate that fine-tuned model (CNN + Softmax) performs better than the pre-trained model relying on CNN and SVM classifier. As it is expected, results reveal that the performance of the deep learning architecture increases with respect to the size of the training data.

References

[1] Tolba, A. S., A. H. El-Baz, and A. A. El-Harby. "Face recognition: A literature review." International Journal of Signal Processing 2.2 (2006): 88-103.

[2] Sharif, Muhammad, Sajjad Mohsin, and Muhammad Younas Javed. "A survey: face recognition techniques." Research Journal of Applied Sciences, Engineering and Technology 4.23 (2012): 4979-4990.

[3] LeCun, Yann, Yoshua Bengio, and Geoffrey Hinton. "Deep learning." nature 521.7553 (2015): 436.

[4] Elgallad, Elaraby A., et al. "Human identity recognition using sparse auto encoder for texture information representation in palmprint images based on voting

technique." Computer Science and Information Technology (SCCSIT), 2017 Sudan Conference on. IEEE, 2017.

[5] Anar, Ali Canberk, Erkan Bostanci, and Mehmet Serdar Guzel. "Live Target Detection with Deep Learning Neural Network and Unmanned Aerial Vehicle on Android Mobile Device." arXiv preprint arXiv:1803.07015.2018.

[6] LeCun, Yann, Koray Kavukcuoglu, and Clément Farabet. "Convolutional networks and applications in vision." Circuits and Systems (ISCAS), Proceedings of 2010 IEEE International Symposium on. IEEE, 2010.

[7] Mikolov, Tomáš, et al. "Strategies for training large scale neural network language models." Automatic Speech Recognition and Understanding (ASRU), 2011 IEEE Workshop on. IEEE, 2011.

[8] Collobert, Ronan, et al. "Natural language processing (almost) from scratch." Journal of Machine Learning Research 12.Aug (2011): 2493-2537.

[9] Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks." Advances in neural information processing systems. 2012.

[10] Farabet, Clement, et al. "Learning hierarchical features for scene labeling." IEEE transactions on pattern analysis and machine intelligence 35.8 (2013): 1915-1929.

[11] Tompson, Jonathan J., et al. "Joint training of a convolutional network and a graphical model for human pose estimation." Advances in neural information processing systems. 2014.

[12] Bordes, Antoine, Sumit Chopra, and Jason Weston. "Question answering with subgraph embeddings." arXiv preprint arXiv:1406.3676 (2014).

[13] Sun, Yi, Xiaogang Wang, and Xiaoou Tang. "Deep learning face representation from predicting 10,000 classes." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2014.

[14] Taigman, Yaniv, et al. "Deepface: Closing the gap to human-level performance in face verification." Proceedings of the IEEE conference on computer vision and pattern recognition. 2014.

[15] Schroff, Florian, Dmitry Kalenichenko, and James Philbin. "Facenet: A unified embedding for face recognition and clustering." Proceedings of the IEEE conference on computer vision and pattern recognition. 2015.

[16] Setiowati, Sulis, Eka Legya Franita, and Igi Ardiyanto. "A review of optimization method in face recognition: Comparison deep learning and non-deep learning methods." Information Technology and Electrical Engineering (ICITEE), 2017 9th International Conference on. IEEE, 2017.

[17] Sun, Yi, et al. "Deep learning face representation by joint identification-verification." Advances in neural information processing systems. 2014.

[18] Yi, Dong, et al. "Learning face representation from scratch." arXiv preprint arXiv:1411.7923 (2014).

[19] Parkhi, Omkar M., Andrea Vedaldi, and Andrew Zisserman. "Deep Face Recognition." BMVC. Vol. 1. No. 3. 2015.

[20] Ahmed, Ejaz, Michael Jones, and Tim K. Marks. "An improved deep learning architecture for person re-identification." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2015.

[21] Sun, Yi, et al. "Deepid3: Face recognition with very deep neural networks." arXiv preprint arXiv:1502.00873 (2015).

[22] Szegedy, Christian, et al. "Going deeper with convolutions." Cvpr, 2015.

[23] Chen, Yanbei, Xiatian Zhu, and Shaogang Gong. "Person re-identification by deep learning multi-scale representations." (2018).

[24] Xiao, Tong, et al. "Learning deep feature representations with domain guided dropout for person re-identification." Computer Vision and Pattern Recognition (CVPR), 2016 IEEE Conference on. IEEE, 2016.

[25] Viola, Paul, and Michael J. Jones. "Robust real-time face detection." International journal of computer vision 57.2 (2004): 137-154.

[26] Dalal, Navneet, and Bill Triggs. "Histograms of oriented gradients for human detection." Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on. Vol. 1. IEEE, 2005.

[27] Cortes, Corinna, and Vladimir Vapnik. "Support-vector networks." Machine learning 20.3 (1995): 273-297.

[28] Li, Xiang-Yu, and Zhen-Xian Lin. "Face Recognition Based on HOG and Fast PCA Algorithm." The Euro-China Conference on Intelligent Data Analysis and Applications. Springer, Cham, 2017.

[29] Albiol, Alberto, et al. "Face recognition using HOG–EBGM." Pattern Recognition Letters 29.10 (2008): 1537-1543.

[30] Déniz, Oscar, et al. "Face recognition using histograms of oriented gradients." Pattern Recognition Letters 32.12 (2011): 1598-1603.

[31] Peker, Murat, Halis Altun, and Fuat Karakaya. "HOG Temelli Bir Yöntem ile Ölçek ve Yönden Bağımsız Gerçek Zamanlı Nesne Tanıma."

[32] Phillips, P. Jonathon. "Support vector machines applied to face recognition." Advances in Neural Information Processing Systems. 1999.

[33] Ayhan, Sevgi, and Şenol Erdoğmuş. "Destek vektör makineleriyle sınıflandırma problemlerinin çözümü için çekirdek fonksiyonu seçimi." Eskişehir Osmangazi Üniversitesi İktisadi ve İdari Bilimler Dergisi 9.1 (2014).

[34] Heisele, Bernd, Purdy Ho, and Tomaso Poggio. "Face recognition with support vector machines: Global versus component-based approach." Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on. Vol. 2. IEEE, 2001.

[35] Wang, Zhe, and Xiangyang Xue. "Multi-class support vector machine." Support Vector Machines Applications. Springer, Cham, 2014. 23-48.

[36] Wu, Jianxin. "Introduction to convolutional neural networks." National Key Lab for Novel Software Technology. Nanjing University. China (2017).

[37] O'Shea, Keiron, and Ryan Nash. "An introduction to convolutional neural networks." arXiv preprint arXiv:1511.08458 (2015).

[38] Grm, Klemen, et al. "Strengths and weaknesses of deep learning models for face recognition against image degradations." IET Biometrics 7.1 (2017): 81-89.

[39] Pilla Jr, Valfredo, et al. "Facial Expression Classification Using Convolutional Neural Network and Support Vector Machine."

[40] Ghazi, Mostafa Mehdipour, and Hazim Kemal Ekenel. "A comprehensive analysis of deep learning based representation for face recognition." arXiv preprint arXiv:1606.02894 (2016).

[41] Lu, Ze, Xudong Jiang, and Alex Kot. "Enhance deep learning performance in face recognition." Image, Vision and Computing (ICIVC), 2017 2nd International Conference on. IEEE, 2017.

[42] Samaria, Ferdinando S., and Andy C. Harter. "Parameterisation of a stochastic model for human face identification." Applications of Computer Vision, 1994., Proceedings of the Second IEEE Workshop on. IEEE, 1994.

[43] Hond, Darryl, and Libor Spacek. "Distinctive Descriptions for Face Processing." BMVC. No. 0.2. 1997.

[44] Bostanci, Betul, and Erkan Bostanci. "An evaluation of classification algorithms using Mc Nemar’s test." Proceedings of Seventh International Conference on Bio-Inspired Computing: Theories and Applications (BIC-TA 2012). Springer, India, 2013.

Current Address: FATIMA ZEHRA UNAL: Ankara University, Department of Computer Engineering, Ankara TURKEY
E-mail: fzkilic@ankara.edu.tr
ORCID: https://orcid.org/0000-0002-1789-0893
Current Address: MEHMET SERDAR GUZEL: Ankara University, Department of Computer Engineering, Ankara TURKEY
E-mail: mguzel@ankara.edu.tr
ORCID: http://orcid.org/0000-0002-3408-0083