

Turkish Journal of Electrical Engineering and Computer Sciences

Volume 28 | Number 2

Article 23

1-1-2020

An automated eye disease recognition system from visual content of facial imagesusing machine learning techniques

ASHRAFI AKRAM

RAMESWAR DEBNATH

Follow this and additional works at: <https://journals.tubitak.gov.tr/elektrik>

 Part of the Computer Engineering Commons, Computer Sciences Commons, and the Electrical and Computer Engineering Commons

Recommended Citation

AKRAM, ASHRAFI and DEBNATH, RAMESWAR (2020) "An automated eye disease recognition system from visual content of facial imagesusing machine learning techniques," *Turkish Journal of Electrical Engineering and Computer Sciences*: Vol. 28: No. 2, Article 23. <https://doi.org/10.3906/elk-1905-42>
Available at: <https://journals.tubitak.gov.tr/elektrik/vol28/iss2/23>

This Article is brought to you for free and open access by TÜBİTAK Academic Journals. It has been accepted for inclusion in Turkish Journal of Electrical Engineering and Computer Sciences by an authorized editor of TÜBİTAK Academic Journals. For more information, please contact academic.publications@tubitak.gov.tr.



An automated eye disease recognition system from visual content of facial images using machine learning techniques

Ashrafi AKRAM*[✉], Rameswar DEBNATH[✉]

Computer Science and Engineering Discipline, Science Engineering and Technology School, Khulna University, Khulna, Bangladesh

Received: 08.05.2019

Accepted/Published Online: 26.10.2019

Final Version: 28.03.2020

Abstract: Many eye diseases like cataracts, trachoma, or corneal ulcer can cause vision problems. Progression of these eye diseases can only be prevented if they are recognized accurately at the early stage. Visually observable symptoms differ a lot among these eye diseases. However, a wide variety of symptoms is necessary to be analyzed for the accurate detection of eye diseases. In this paper, we propose a novel approach to provide an automated eye disease recognition system using visually observable symptoms applying digital image processing techniques and machine learning techniques such as deep convolution neural network (DCNN) and support vector machine (SVM). We apply the principal component analysis and t-distributed stochastic neighbor embedding methods for better feature selection. The proposed system automatically divides the facial components from the frontal facial image and extracts the eye part. The proposed method analyzes and classifies seven eye diseases including cataracts, trachoma, conjunctivitis, corneal ulcer, ectropion, periorbital cellulitis, and Bitot's spot of vitamin A deficiency. From the experimental results, we see that the DCNN model outperforms SVM models. We also compare our method with some other existing methods. Our method shows improved accuracy compared to other methods. The average accuracy rate of our DCNN model is 98.79% with sensitivity of 97% and specificity of 99%.

Key words: Deep convolution neural network, support vector machine, principal component analysis, t-distributed stochastic neighbor embedding, automated eye disease recognition

1. Introduction

Eye diseases can lead to partial or even complete absence of vision if they are left unobserved in the initial period. Early detection of these eye diseases can prevent vision impairment [1]. In recent years, digital image processing and machine learning techniques are widely used for automatic disease detection, diagnosis, and clinical decision-making procedures to achieve the optimum and most accurate results [2, 3]. Machine learning algorithms have been used for several challenging tasks, such as brain tumor segmentation with magnetic resonance (MR) imaging [4], age-related eye diseases [5], eye tumor detection [6], skin disease detection [7], and automated diabetic retinopathy testing [8]. Digital image processing and machine learning techniques are also being applied in eye disease recognition [5, 6, 8–10]. It may potentially be able to detect abnormal changes in the cells, sclera, cornea, iris, pupil, and blood vessels. Many eye diseases such as cataracts, trachoma, corneal ulcer, conjunctivitis, and ectropion can be detected by observing visual symptoms. Most people delay to take the correct steps to restore their eye disease problems. Expert eye doctors are able to identify diseases by

*Correspondence: ashrafiku16@gmail.com

observing eyes' visual symptoms but they are not available in many remote areas worldwide. Thus, researchers are interested in developing an automated intelligent system that can provide detection and segmentation of the eye region and classification of eye diseases.

Several approaches for eye disease recognition are found in the literature. Among them, machine learning techniques outperform other methods. Dimililer et al. [6] proposed an eye tumor recognition system of different iris tumors' detection using backpropagation neural networks (BPNNs). For the experiments, they used only 100 images in a dataset (50 normal images and 50 abnormal images), with 30 images for training and 70 images for testing. The recognition rate was 95%.

Gargeya et al. [8] proposed a diagnostic tool using a deep learning algorithm for automated diabetic retinopathy detection. The proposed algorithm processed color fundus images and categorized them as no retinopathy or generating an abnormal subregion of diabetic retinopathy. They tested the model using 5-fold cross-validation on a local dataset of 75,137 color fundus images. The accuracy rate was 97% with sensitivity of 94% and specificity of 98%. Wang et al. [9] proposed a machine vision-based algorithm for visually observable symptoms on faces using semisupervised anomaly detection. The proposed system is focused on detecting and classifying ill faces into multiple categories where only two eye diseases can be recognized. The accuracy rate is 76.6% with sensitivity of 81% for ill face detection.

Salam et al. [10] proposed a system to recognize glaucoma from digital fundus images using a hybrid feature set. This proposed methodology is a combination of structural (cup to disc ratio) and nonstructural (texture and intensity) features to improve the accuracy of automated analysis of glaucoma. They evaluated their proposed system with two local datasets. One dataset consists of 50 fundus images (15 glaucoma and 35 healthy images) and the other comprises 100 fundus images (26 glaucoma and 74 healthy eye images). The accuracy rate is 97% with specificity 98% and sensitivity 92%. Gunay et al. [11] proposed an automated diagnosing system of adenoviral conjunctivitis using the facial picture of the ill face. They measured the vascularization and intensity of redness in pink eyes after segmenting the sclera regions of eye images to diagnose conjunctivitis with only 30 images (18 healthy and 12 adenoviral conjunctivitis eye images). The average accuracy rate is 96%.

Most of the existing works present recognition results on one or two eye diseases, but there is still a lack of research on recognition of some other important eye diseases. The main pitfall in this area are the lack of benchmarks and lack of publicly available visual content-based eye disease datasets for many diseases. In this paper, we aim to develop a standard dataset with many samples as well as many eye diseases. We develop our dataset with seven diseases and the dataset contains 1753 images. These datasets can be made available free of cost to researchers of other institutions. We develop our system for recognition of seven eye diseases using CNN and SVM because, in general, CNN and SVM outperform other machine learning techniques for many recognition tasks.

The proposed method segments the facial components using a digital image processing technique. Then we detect the eye region automatically and these features are applied to DCNN and SVM models. We also apply PCA and t-SNE for feature selection and then classification is done using SVM based on the radial basis function (RBF) kernel. In existing system authors tested their methods with limited numbers of eye diseases. In this paper we choose seven eye diseases including cataracts, trachoma, corneal ulcer, conjunctivitis, ectropion, periorbital cellulitis, and Bitot's spot of vitamin A deficiency. From the experimental results, we obtain 98.79% average accuracy with sensitivity of 97% and specificity of 99% by the proposed DCNN model for seven diseases. When the SVM model is considered, the SVM, PCA-SVM, and t-SNE-SVM achieve average accuracy rates of

91.70%, 96.13%, and 80.45% with sensitivity of 90.5%, 93.5%, and 12.5% and specificity of 94.13%, 98.87%, and 87.37%, respectively. We also compare our method with some existing methods. We compare our method with the method in [9] for diseases like periorbital cellulitis and corneal ulcer. The method's average recognition rate is 76.6%. We also test our method for adenoviral conjunctivitis recognition with the method in [11]. The method's average accuracy rate is 96%. Overall our method achieves better accuracy than others. We get a recognition rate of 98.79% with sensitivity of 97% and specificity of 99%.

The rest of the paper is organized as follows. In Section 2, we describe our proposed methodology. Section 3 shows the experimental and quantitative analysis of the results obtained from the algorithms. Finally, Section 4 concludes the paper.

2. Proposed methodology

In this paper, an automated eye disease recognition system is designed using various machine learning techniques. In our eye disease recognition system, the human eye part is obtained from the facial image automatically. The first stage is the capturing of image. This procedure is acquired by digital camera. Initially, the original image has been loaded as an input image and then the method detects the face from the input image by our algorithm. Our method scales the facial images at 500×500 pixels. Then the method segments the various facial components. Once the eye part is segmented from the face part, we apply the eye parts of the image for learning. The phases of the proposed method are shown in Figure 1.

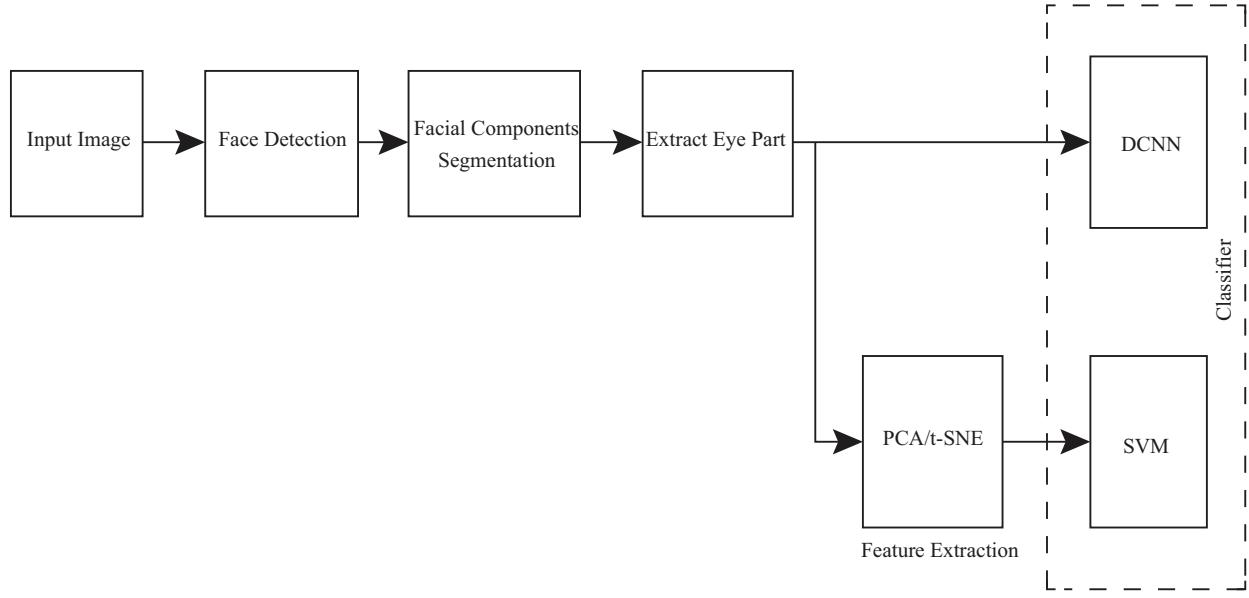


Figure 1. Proposed methodology.

2.1. Facial features detection

Facial feature points are generally obtained from facial components such as eyes, nose, jaw, mouth, etc. In our proposed system, the facial feature point detection involves three steps: step 1, localizing the face in the image; step 2, detecting facial feature points of each face component; and step 3, segmenting the facial component. We apply a method for face detection that uses the histogram of oriented gradient (HOG) to construct the feature

vector and linear SVM classifier for detection as proposed in [12]. We use the HOG-based feature selection technique because it outperforms other existing feature selection techniques for human detection [12]. The linear SVM is used as a baseline classifier for face or nonface classification for its simplicity and higher speed.

The HOG descriptor technique computes occurrences of histogram gradient orientation in a small spatial region of an image referred to as a “cell” [12, 13]. The image is partitioned into cells of size $N \times N$ pixels. Then it evaluates the vectors that represent histograms of orientated gradients of each cell in the detection window. The gradient vector for x and y directions is calculated by Equations (1) and (2), respectively. In these equations, L is a pixel intensity (gray-scale) function for the (x, y) direction in an image I . Then the gradients are used to calculate gradient magnitude $M_{x,y}$ and gradient orientation $\theta_{x,y}$ by Equations (3) and (4), respectively.

$$gx = \frac{\partial I}{\partial x} = L(x, y + 1) - L(x, y - 1) \quad (1)$$

$$gy = \frac{\partial I}{\partial y} = L(x + 1, y) - L(x - 1, y) \quad (2)$$

$$M_{x,y} = (gx^2 + gy^2)^{\frac{1}{2}} \quad (3)$$

$$\theta_{x,y} = \tan^{-1} \frac{gx}{gy} \quad (4)$$

The orientation of all pixels is computed and accumulated in an M-bins histogram of orientations over $N \times N$ spatial cells. Then all the achieved histograms are concatenated into unique histogram vectors in order to construct the final features vector. Finally, the feature vectors are given to a linear SVM to classify face or nonface. The SVM will be discussed in Section 2.2.2. After detecting a face, we apply the facial landmark extraction method to localize and label facial components. These landmarks are located around edges of facial components such as the eyes, nose, mouth, eyebrows, etc. This facial landmark extraction method is based on an ensemble of regression trees proposed in [14]. Thus, in this technique, each stage regressor in the cascaded shape regression framework is based on the ensemble of regression trees [14]. The ensemble of regression trees can be used to regress the position of facial landmarks. Each regressor in the cascade makes its predictions based on features such as pixel intensity values extracted from the face image [14]. The features used in each regressor in the cascade returns a shape vector, which is used to update the current shape of the estimate at each stage. The processes can be formulated as follows:

$$S_{t+1} = S_t + r_t(\phi_t(I, S_t)) \quad (5)$$

Here, r_t is a regressor at stage t , I is an input image, and S_t is the currently estimated shape vector. It will update the S_t stage by the stage where $\phi_t(I, S_t)$ is a function referred to as a shape-index feature, which depends on the current estimate of S_t . In [14], Kazemi et al. used 194 landmarks to detect facial components on face images. In our methodology, we use 68 landmarks to detect facial components because 68 landmarks are sufficient for detecting main components of faces.¹ By this method, facial components such as the jaw, right eye, left eye, mouth, nose, left eyebrow, and right eyebrow can be accessed through facial landmarks 0 to 17, 37 to 42, 43 to 48, 49 to 68, 28 to 36, 18 to 22, and 23 to 27, respectively, as shown in Figure 2. This technique

¹Official Dlib Library [online]. Website <http://dlib.net/> [accessed 10 September 2018].

helps to detect the facial components for determining the specific region of the face. Once the area of facial components is detected after the facial landmark extraction process, the system extracts the ROI from the face image shown in Figure 3.

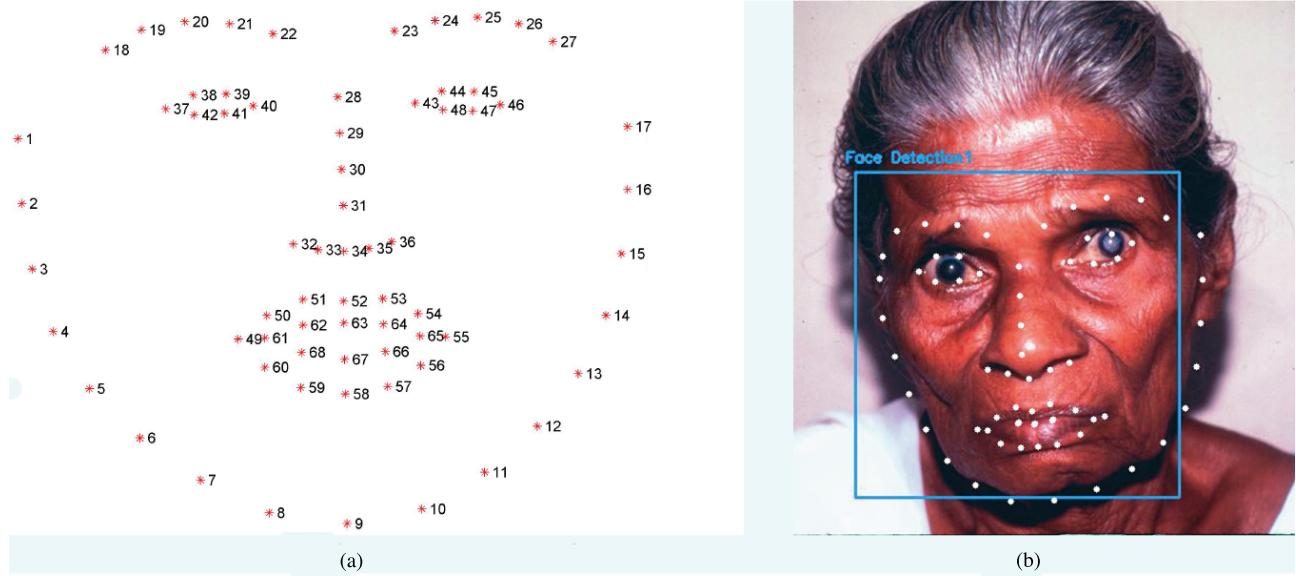


Figure 2. The 68 facial landmarks for the face region.



Figure 3. Region of interest from facial image.

2.1.1. Eye extraction

The main goal of facial feature detection is to segment the facial components, in particular for eye region extraction and segmentation. We detect and locate the position of eyes using a facial landmark detector [14].

Then it converts the facial landmarks' $(x_{(l,i)}, y_{(l,i)})$ -coordinates (where l is an index of the landmark), which specify where the 68 landmarks are. Each eye part contains six landmarks, starting at the left corner of the eye as shown in Figure 4. For each of the eye regions, we determine the starting and ending index values to extract (x, y) -coordinates. Using these indexes we extract eye regions and scale to resolution of 70×70 pixels.

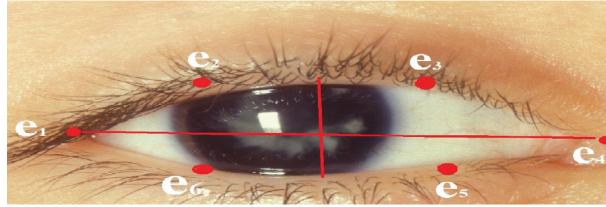


Figure 4. Eye extraction process using facial landmarks.

Then the segmented eye parts are used as training samples for the CNN and SVM, so that the system automatically recognizes eye diseases for specific categories.

2.2. Classifiers

Our automatic eye disease recognition consists of several steps: facial feature extraction, ROI (region of interest), and feature learning. In this section, we describe our proposed machine learning methods such as deep convolution neural network (DCNN) and support vector machine (SVM) to classify several eye diseases.

2.2.1. DCNN classifier

A deep convolution neural network (DCNN) is a multilayer neural network that is performed as a deep supervised learning method [15]. The DCNN has achieved excellent performances on image recognition tasks for the last few years [7, 8, 16, 17]. It can perform both feature extraction and image classification tasks [15]. The CNN architecture has two parts: first, a convolutional layer and a max-pool layer acting as a hierarchical feature extractor, where this feature extractor maps the input image intensities to a feature vector, and second, a fully connected layer that performs as a classifier where the extracted features are classified. It is followed by a soft-max activation function, as they have only one output neuron for every class. Due to the DCNN's excellent performances, we have proposed a method for feature extraction and disease recognition from eye images that uses DCNN. We have built a custom architecture for our DCNN model inspired by earlier works in [16, 17] for recognition tasks using the CNN. We mainly explore six convolution layers, six leaky ReLUs, three max-poolings, three fully connected layers, and five dropout layers. The first convolution layer receives $70 \times 70 \times 3$ (height \times width \times three input channels) = 14,700 input neurons associated with each pixel in the image. The values associated with each image's data matrix are normalized and fed to the hidden layer to get their classifications. The CNN-based architecture is presented in Figure 5. Table 1 shows the descriptions of the proposed CNN architecture.

As shown in Figure 5, each layer of convolution and max-pooling is composed of multiple 2D planes, which are called feature mappings. Each feature map consists of multiple independent neurons, which receive inputs from a small region in the previous layer. All the neurons of each feature map use the same kernel and connecting weights. Each neuron of the convolution layer extracts the features from input images. The higher level features can be obtained by using these extracted features from the subsequent layers. As the feature propagates to the highest layer, the dimension of each feature map layer reduces the feature size from

Table 1. Description of proposed DCNN architecture.

Layers	Layers (type)	Feature maps and neurons	Kernel	Number of parameters
0	Input image	3@70 × 70	—	—
1	Convolution	256@66 × 66	5 × 5	19456
2	LeakyReLU	LeakyReLU	—	0
3	Batch normalization	Batch normalization	—	1024
4	Convolution	256@62 × 62	5 × 5	1638656
5	LeakyReLU	LeakyReLU	—	0
6	Max-pooling	256@31 × 31	2 × 2	0
7	Batch normalization	Batch normalization	—	1024
8	Dropout	0.3	—	—
9	Convolution	128@27 × 27	5 × 5	819328
10	LeakyReLU	LeakyReLU	—	0
11	Batch normalization	Batch normalization	—	512
12	Convolution	128@23 × 23	5 × 5	409728
13	LeakyReLU	LeakyReLU	—	0
14	Max-pooling	128@11 × 11	2 × 2	0
15	Batch normalization	Batch normalization	—	512
16	Dropout	0.2	—	—
17	Convolution	256@7 × 7	—	819456
18	LeakyReLU	LeakyReLU	—	0
19	Batch normalization	Batch normalization	—	1024
20	Convolution	256@3 × 3	5 × 5	1638656
21	LeakyReLU	LeakyReLU	—	0
22	Max-pooling	256@1 × 1	2 × 2	0
23	Batch normalization	Batch normalization	—	1024
24	Dropout	0.3	—	—
25	Flatten	256	—	—
26	Dense	512	Sigmoid(activation)	131584
27	Batch normalization	Batch normalization	—	2048
28	Dropout	0.5	—	—
29	Dense	256	Sigmoid(activation)	—
30	Batch normalization	Batch normalization	—	1024
31	Dropout	0.5	—	—
32	Dense	8	—	2056
33	Softmax	8	—	—
34	Classification output	Categorical Cross entropy	—	—

the previous feature size that depends on the size of the convolutional and max-pooling layer. For better classification accuracy the number of features is usually increased to select or map suitable features.

Suppose we have an image $I_{p \times q}$, where p and q are the length and width of the image. Each image of the CNN is convolved with 2D kernel $W_{m,n}^s$, where m and n represent the kernel size and s represents a used

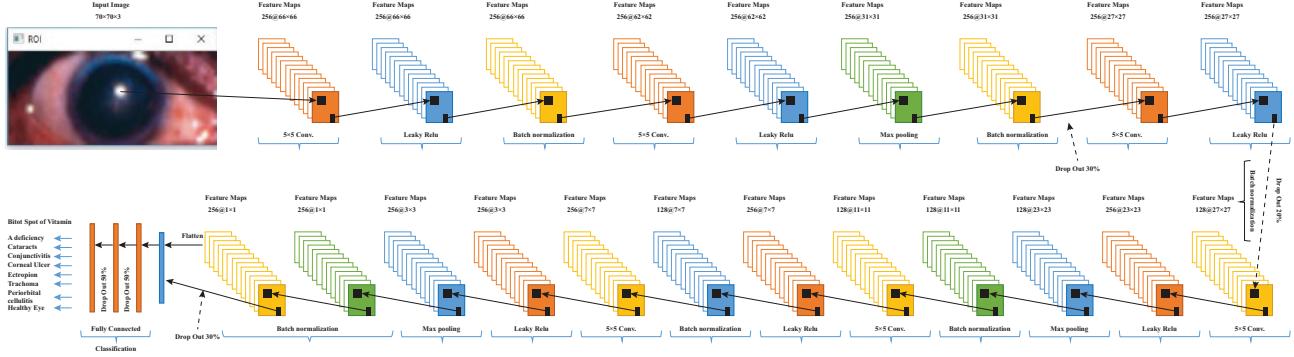


Figure 5. The architecture of the CNN.

kernel. The convolutional layer performs the mapping as follows:

$$C_{(x,y)}^s = I_{p \times q} * W_{m,n}^s \quad (6)$$

Each entry of $C_{(x,y)}^s$ can be defined as shown in Equation 7:

$$C_{(x,y)}^s(i,j) = \sigma \left(\sum_{u=i}^m \sum_{v=j}^n I_{p \times q}(u-i)(v-j) * W_{m,n}^s(u,v) + b^s \right) \quad (7)$$

Here, σ is a nonlinear function. The result passes through the leaky ReLU. Although in CNN architecture ReLU is the most used activation function, in our proposed architecture leaky ReLU activation is applied to fix the problem of dying neurons during backpropagation with gradient value of 0.1. The feature extractor with activation function leaky ReLU produced features vectors that consistently outperformed the feature extractors with other activation functions. We build several dropout layers to reduce model overfitting. The output of the convolutional layer is a 2D vector, which flattens into a single dimensional vector that is used as an input layer with three fully connected layers. The final layer, i.e. the fully connected layer, performs the classification followed by a softmax activation function [16]. For an input sample x , weight vector W , and K distinct linear functions, the softmax function can be defined for the i th class as follows:

$$P(y = i|x) = \frac{\exp(x^T w_i)}{\sum_{k=1}^K \exp(x^T w_k)} \quad (8)$$

2.2.2. SVM classifier

The support vector machine (SVM) is a linear binary classifier proposed by Vapnik in 1995 [18]. It performs intelligent machine techniques for the purpose of condition monitoring and medical diagnosis using its excellent ability in the classification process. In this paper we focus on the SVM using radial basis function (RBF) kernels for solving nonlinear separable classification problems [19]. Given a supervised soft-margin classification problem and a training set of N data points $\{y_i, x_i\}_{i=1}^m$, where $x_i \in \mathbb{R}^n$ is the i th input pattern and $y_i \in \mathbb{R}$ is the i th output pattern, the SVM method aims at constructing a classifier that is defined as follows:

$$y(x) = \text{sign} \left(\sum_{i=1}^m \alpha_i \gamma_i k(x_i, x) + b \right) \quad (9)$$

Here, α_i and b are obtained from a quadratic optimization problem. The quadratic optimization problem has a trade-off parameter C, which is defined by the experiment or user. In this paper we applied a Gaussian radial basis function (RBF) kernel that is defined as follows:

$$K(x_i, x) = e^{\gamma ||x_i - x||^2}, \quad (10)$$

where γ is a parameter as a Gaussian kernel function [18]. We also apply two feature selection methods, PCA and t-SNE. PCA is used to reduce dimensionality by eliminating redundant information of feature vectors [20]. PCA gives only as much variation as the dimensionality feature vector [20]. The t-SNE also reduces dimensions to a reasonable amount by minimizing the divergence between two distributions [21]. One distribution is the pairwise similarities of the input objects in original space (high-dimensional) and the other is the pairwise similarities of the corresponding low-dimensional points in the embedding. The model utilizes grid searching to select hyperparameters, which are a combination in the range of C and γ .

3. Experimental results

In this section, we describe the dataset used for eye disease experimentation, present the experimental setup and results, analyze different machine learning algorithm settings, and then compare our method to other eye disease recognition techniques.

3.1. Datasets

First we develop a dataset for seven eye diseases. Images are independently collected from International Centre for Eye Health,² clinical images for symptoms on faces from the University of Rochester [9], UCSD School of Medicine and VA Medical Center [9], the Primary Care Dermatology Society [9], and other different resources [22, 23]. Some sources contain full face images with diseases. We crop the eye parts and resize the images. All images are resized to 256×256 . The dataset contains 1753 images. Table 2 shows the statistics of the image dataset. Figure 6 shows some image samples from our dataset. The symptoms of selected eye diseases include several visual abnormalities in the eye region, particularly blurred or clouded or yellowing lens, gray or white spots on the cornea, red or bloodshot eyes, yellow or greenish-yellow coatings on eyes, foamy white spots in sclera, swollen eyes, eyelid deformity such as the length of the lower eyelid being turned out from the eye, or reddish bumps on the edge of an inner eyelid depending on specific diseases, and symptoms are different for each disease.

3.2. Experimental setup and results

The entire experiment was conducted on a system with an Intel Core i7-7700HQ, an additional GPU (NVIDIA GeForce GTX 1060, 6 GB GDDR5), 512 GB SSD memory, and Keras with Tensorflow on the back end. All the model's results were implemented using OpenCV and Python. The facial landmark detector with pretrained models and the dlib library¹ are used to evaluate the 68 landmarks of facial structures on the face image [23]. The facial landmark extraction method uses the iBUG 300-W dataset for training [23]. This dataset comprises 135 images with 68 landmarks including different poses, lighting conditions, subjects, etc.

We conduct comparative experiments for eye disease recognition with DCNN and SVM models on our dataset. Each sample in the original dataset is an image of size 256×256 pixels. The original eye images are

²International Centre for Eye Health, London School of Hygiene & Tropical Medicine [online]. Website www.flickr.com/communityeyehealth [accessed 10 August 2018].

Table 2. Statistics of image dataset.

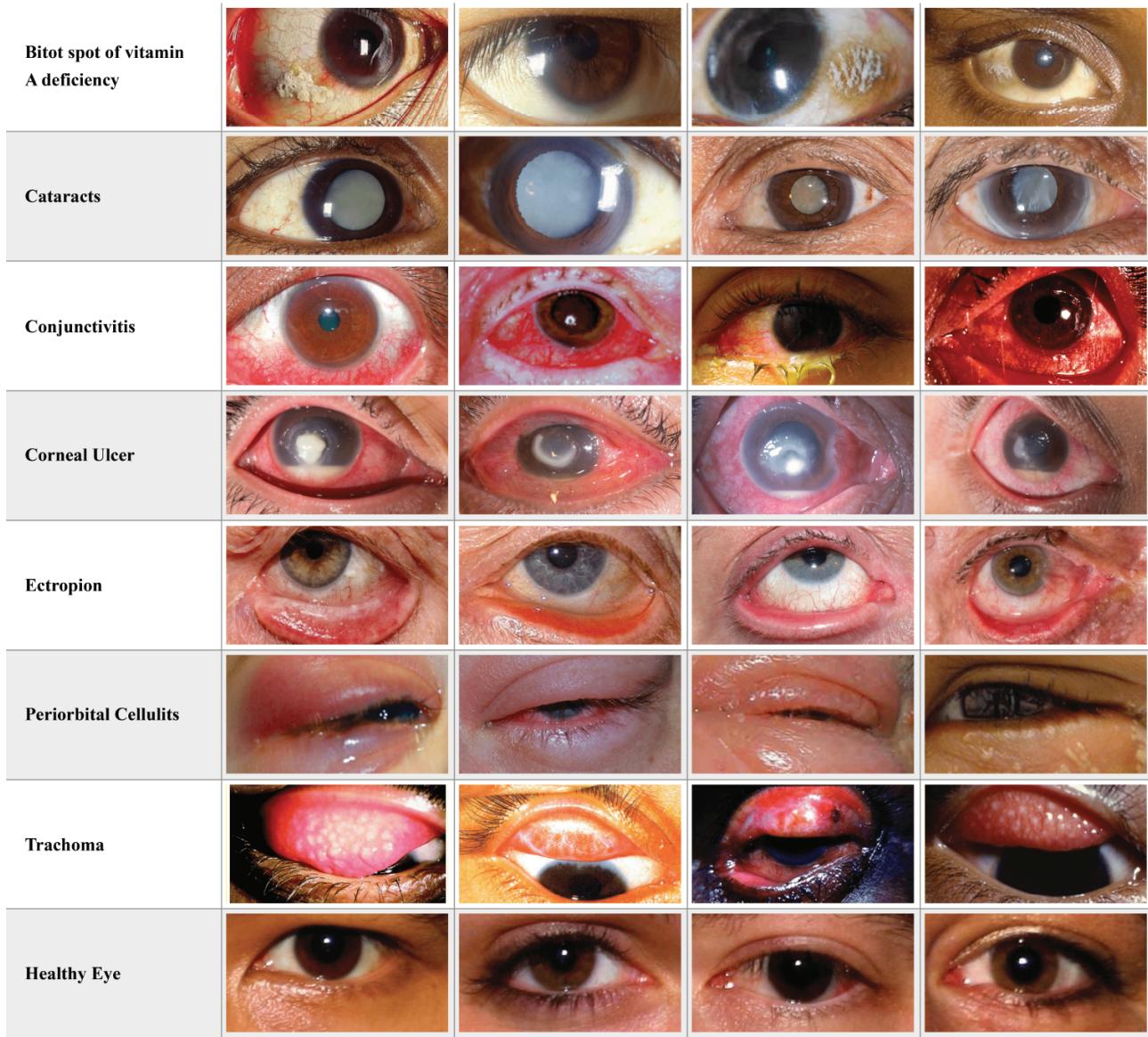
Images	Number of images
Corneal ulcer	362
Ectropion	209
Cataracts	187
Conjunctivitis	249
Trachoma	122
Periorbital cellulitis	168
Bitot's spot of vitamin A deficiency	187
Healthy	269
Total images	1753

too large for preprocessing efficiently. We have resized all the eye disease images to 70×70 pixels to reduce the time of training that was automatically evaluated by written script in Python, using Keras and the OpenCV framework. To reduce biases in the feature selection of the validation set, we perform a 10-fold cross-validation technique on the dataset. In 10-fold cross-validation, the original dataset is randomly divided into ten equally sized subsets. Then, each time, a single subset is used as the validation set and nine other subsets are used for training with 40 epochs each. Here, 20% of the original images are reserved as a test set, and 1402 images are randomly selected from the remaining dataset for training our models at the starting point. The DCNN model identifies 334 images out of 351 eye images correctly. From the experimental results of the DCNN model we see that the average accuracy rate is 98.79% with specificity of 97% and sensitivity of 99%. To evaluate the CNN model's performance, several performance metrics are used such as accuracy, precision, recall, and F-score as shown in Table 3. Figure 7 depicts the CNN model's accuracy and loss with the number of epochs, resulting in a mean error rate of the proposed system of 3%.

Table 3. Statistical results of CNN model.

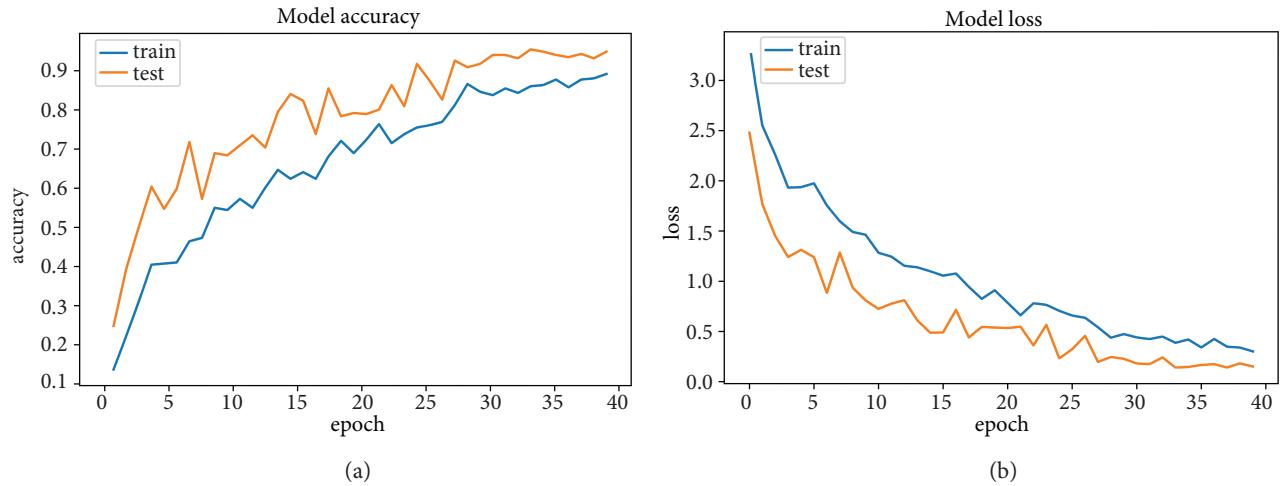
Model		Precision	Recall	F-score	Accuracy % (cross-validation [k=10])	Average model accuracy % (cross-validation [k=10])
CNN model	Bitot's spot of vitamin A deficiency	1.00	0.95	0.97	99.43	98.79
	Cataracts	1.00	0.95	0.97	99.43	
	Conjunctivitis	1.00	0.94	0.96	98.86	
	Corneal ulcer	0.95	1.00	0.97	98.29	
	Ectropion	0.97	0.98	0.94	98.58	
	Periorbital cellulitis	1.00	0.82	0.90	98.30	
	Trachoma	1.00	0.88	0.93	99.15	
	Healthy eye	0.90	1.00	0.95	98.29	

In order to get the other three estimations, the proposed system uses hybrid features of SVM for recognition of seven eye diseases. We propose the RBF kernel-based support vector machine [18] with three different models such as the SVM model, PCA-SVM model, and t-SNE-SVM model. We test our dataset with

**Figure 6.** Sample images of eye diseases and healthy eyes.

SVM, PCA-SVM, and t-SNE-SVM models. The PCA and t-SNE are used for feature selection of segmented eyes from images and decreasing the feature matrix size by selecting the most important features. We compare among these models with performance metrics such as accuracy, precision, recall, and F-score as shown in Table 4. We select the range of regularization parameter C and the value of γ and apply a 10-fold cross-validation technique on our dataset using the SVM with RBF kernel.

The grid searching range of the SVM in each parameter is $C = [2^{-2} \dots 2^7]$ and $\gamma = [2^{-7} \dots 2^2]$. In all combinations of SVM models tried we have $10 \times 10 = 100$ different combinations. We have achieved the best accuracy rate for $C = 14.444$ and $\gamma = 0.008$ in the SVM, PCA-SVM, and t-SNE-SVM models. These parameters are then used to train and test these three models. The SVM model without PCA is able to achieve an average accuracy rate of 91.70% with sensitivity of 91% and specificity of 94%. The PCA-SVM model

**Figure 7.** The training and testing of CNN model accuracy and loss, respectively.**Table 4.** Comparison of hybrid feature classification with SVM.

	SVM without PCA				Average accuracy(%)	SVM with PCA				Average accuracy(%)	SVM with t-SNE				Average accuracy(%)
	Precision	Recall	F-Score	Accuracy (%)		Precision	Recall	F-Score	Accuracy (%)		Precision	Recall	F-Score	Accuracy(%)	
Bitot's spot of vitamin A deficiency	1.00	0.95	0.97	92.43	91.70	1.00	1.00	1.00	99.13	96.13	0.15	0.17	0.18	89.74	80.45
Cataracts	0.66	1.00	0.79	76.77		0.73	1.00	0.84	96.01		0.32	0.34	0.32	89.45	
Conjunctivitis	0.94	0.96	0.98	93.67		1.00	1.00	1.00	97.00		0.24	0.23	0.24	85.75	
Corneal ulcer	0.97	0.94	1.00	92.72		1.00	1.00	1.00	96.15		0.37	0.45	0.41	21.93	
Ectropion	0.95	0.92	0.97	93.17		1.00	1.00	1.00	97.07		0.11	0.15	0.11	88.03	
Periorbital cellulitis	1.00	0.76	0.86	92.45		1.00	0.82	0.90	98.29		0.18	0.11	0.15	90.31	
Trachoma	1.00	0.71	0.83	95.67		1.00	0.75	0.86	95.15		0.27	0.18	0.23	93.16	
Healthy eye	1.00	0.96	0.98	96.72		1.00	0.96	0.98	97.21		0.17	0.13	0.16	84.04	

shows an accuracy rate of 96.13% with sensitivity of 94% and specificity of 99%. The t-SNE-SVM model shows accuracy rate of 80.78% with sensitivity of 13% and specificity of 87%. The CNN model shows the highest accuracy rate among all the models based on some statistical results such as model accuracy rate, precision, recall, and F-score. Specificity and sensitivity are mainly used to detect the suspected disease or disease-free state in the autonomous medical diagnostic process. We also compute specificity and sensitivity, defined as follows:

$$\text{Specificity} = \frac{TN}{TN + FP} \quad (11)$$

$$Sensitivity = \frac{TP}{TP + FN} \quad (12)$$

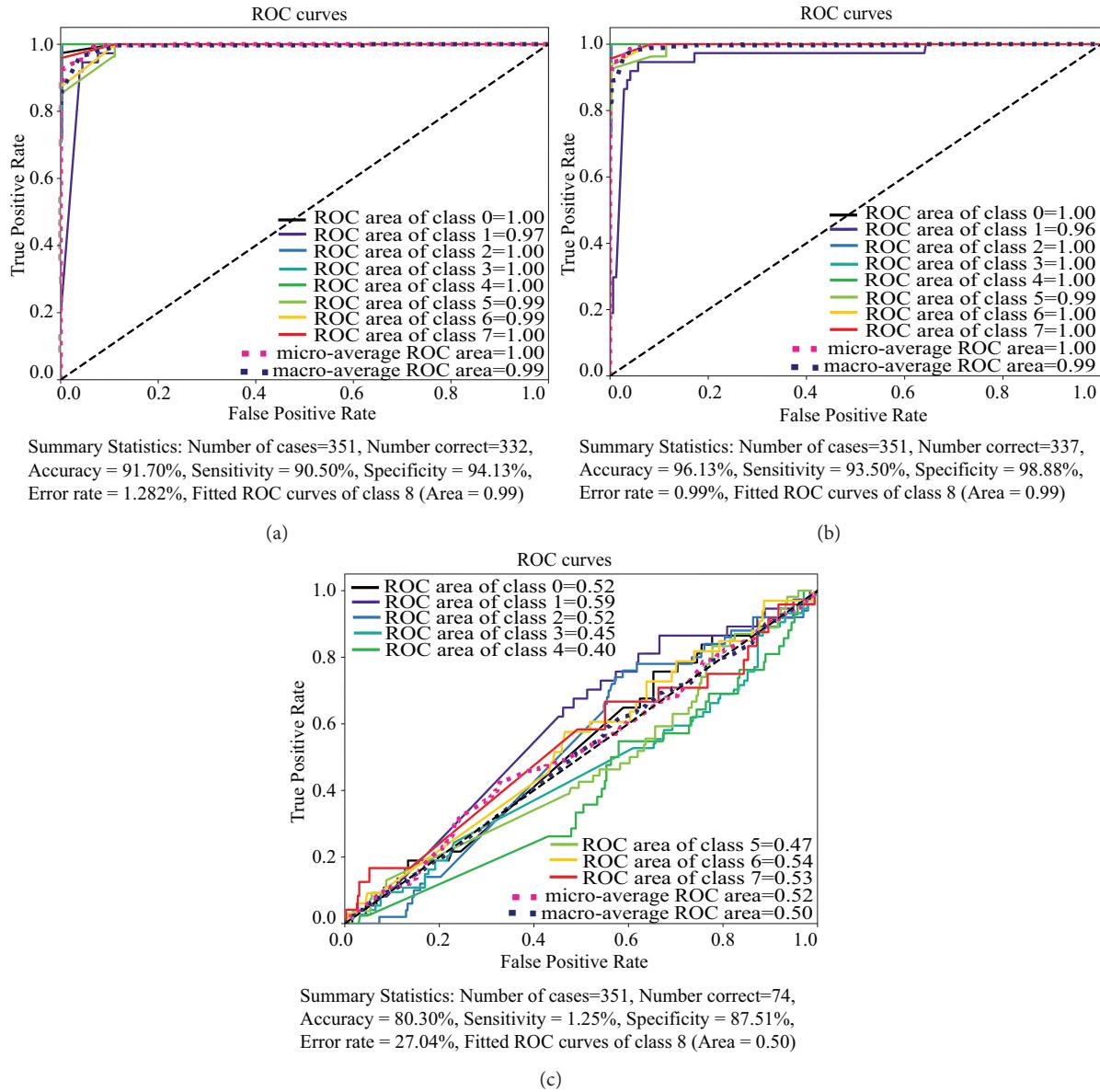
True positives (TP), true negatives (TN), false positives (FP), and false negatives (FN) are vital to understanding the evaluation of disease diagnosis. A true positive (TP) is defined as the patient having an eye disease that was successfully detected as the selected eye disease image. A false positive (FP) is defined as a patient not having the eye disease but being detected as having the eye disease symptoms, true negative (TN) is defined as the patient not having the eye disease and being detected as a healthy eye image, and a false negative (FN) is defined as the patient having the eye disease but the eye disease not be recognized. Table 5 shows the specificity and sensitivity of the proposed model. The proposed CNN model obtained the highest sensitivity rate of 97% and specificity of 99%. The t-SNE-SVM-based model shows the lowest sensitivity rate of 13% and specificity of 88%. From the results, sensitivity and specificity are improved by the proposed system. Moreover, we see that the results of our methods are better than those of other eye disease recognition systems.

Table 5. Comparison of statistical results of proposed machine learning techniques.

	CNN				SVM				PCA-SVM				T-SNE-SVM			
	TP	FP	Sensitivity	Specificity	TP	FP	Sensitivity	Specificity	TP	FP	Sensitivity	Specificity	TP	FP	Sensitivity	Specificity
Bitot's spot of vitamin A deficiency	35	0	0.95	1.00	35	0	0.95	1.00	35	0	0.95	1.00	0	0	0	1.00
Cataracts	35	0	0.99	1.00	37	19	1.00	0.94	37	14	1.00	0.96	0	0	0	1.00
Conjunctivitis	47	1	1.00	0.99	50	0	0.98	1.00	50	0	1.00	0.98	0	0	0	1.00
Corneal ulcer	74	6	1.00	0.98	74	0	1.00	1.00	74	0	1.00	0.97	74	274	1.00	0.01
Ectropion	41	4	0.98	0.99	42	0	0.88	0.91	42	0	1.00	1.00	0	0	0	1.00
Periorbital cellulitis	27	0	0.94	1.00	25	0	0.96	0.86	27	0	0.96	1.00	0	1	0	0.99
Trachoma	21	0	0.89	1.00	17	0	0.76	0.82	18	0	0.82	1.00	0	0	0	0.99
Healthy eye	54	6	1.00	0.98	52	0	0.71	1.00	52	0	0.75	1.00	0	2	0	1.00

Figure 8 shows the ROC curve calculated with the true positive and false positive statistical features from Table 5. The fitted ROC areas for the proposed systems are 0.99, 0.99, and 0.50 for the SVM, PCA-SVM, and t-SNE-SVM models, respectively. Here class 0, class 1, class 2, class 3, class 4, class 5, class 6, and class 7 are Bitot's spot of vitamin A deficiency, cataracts, conjunctivitis, corneal ulcer, ectropion, healthy, periorbital cellulitis, and trachoma, respectively. The ROC area estimates the overall performance of the algorithm. Results show that the image of an eye is classified as diseased or healthy.

In Table 6, we show the comparison of our method with some existing methods. We get the results of other methods from [9, 11]. We apply a 10-fold cross-validation technique on 1753 eye images for computing our results. Although in the datasets the learning task and experimental setup are not the same as in our proposed method, it is inconvenient to compare the performance directly. However, we get an assumption of improved results by our method compared to the other methods. From the experimental results in Table 6, we see that our method shows better results than the existing methods.

**Figure 8.** The ROC curves: (a) SVM, (b) PCA-SVM, (c) t-SNE-SVM, calculated by the statistical data from Table 5.**Table 6.** Comparison of proposed method with some existing methods.

Method	Eye disease	Sample size (image type)	Accuracy (%)	Proposed CNN method accuracy (%)
Wang et al. [9]	Periorbital cellulitis and corneal ulcer	237 (more than 20 diseases)	76.6	98.30
Gunay et al. [11]	Adenoviral conjunctivitis	30 (18 healthy and 12 adenoviral conjunctivitis)	96	98.86

4. Conclusion

In this paper, we propose a visual content-based eye disease recognition system from facial images using image processing and machine learning techniques. In this paper we have developed a benchmark visual content-based image dataset for seven eye diseases containing data of 1753 images. Our proposed system is based on an algorithm that automatically crops the eye part from a frontal facial image. These eye parts are used for learning. We apply two learning methods for classification: DCNN and SVM. We also apply PCA and t-SNE for feature selection and then classify by SVM. We test our system with seven eye diseases that are visually observable. From the experimental results we see that DCNN outperforms SVM models. Of the SVM models, the PCA-SVM model shows better results than the other two SVM models. We also compare the recognition accuracy of DCNN methods with some other existing methods for eye disease recognition. It is observed that our method achieves better accuracy than others. We get a recognition rate of our method of 98.79% with sensitivity of 97% and specificity of 99%.

Acknowledgment

This research was funded by the Information and Communication Technology (ICT) Division, Ministry of Posts, Telecommunications, and Information Technology, Government of the People's Republic of Bangladesh.

References

- [1] Ackland P, Resnikoff S, Bourne R. World blindness and visual impairment: despite many successes, the problem is growing. *Community Eye Health Journal* 2017; 30 (100): 71-73.
- [2] Wiemer J, Schubert F, Granzow M, Ragg T, Fieres J et al. Informatics united: exemplary studies combining medical informatics, neuro informatics and bioinformatics. *Methods of Information Medical Journal* 2003; 42 (2): 126-133. doi: 10.1055/s-0038-1634323
- [3] Wan J, Wang D, Hoi SCH, Pengcheng W, Jianke ZHU et al. Deep learning for content-based image retrieval: a comprehensive study. In: MM '14: Proceedings of the 22nd ACM International Conference on Multimedia; Orlando, FL, USA; 2014. pp. 157-166. doi: 10.1145/2647868.2654948
- [4] Bauer S, Wiest R, Nolte LP, Reyes M. A survey of MRI-based medical image analysis for brain tumor studies. *Physical Medical Biology* 2013; 58 (13): R97–R129. doi: 10.1088/0031-9155/58/13/R97
- [5] Gao X, Lin S, Wong TY. Automatic feature learning to grade nuclear cataracts based on deep learning. *IEEE Transactions on Biomedical Engineering* 2015; 62: 2693-2701. doi: 10.1109/TBME.2015.2444389
- [6] Dimililer K, Ever YK, Ratemi H. Intelligent eye tumour detection system. *Procedia Computer Science* 2016; 102 (C): 325–332. doi: 10.1016/j.procs.2016.09.408
- [7] Esteva A, Kuprel B, Novoa RA, Ko J, Swetter SM et al. Dermatologist-level classification of skin cancer with deep neural networks. *Nature* 2017; 542 (7639): 115–118. doi: 10.1038/nature21056
- [8] Gargyea R, Leng T. Automated identification of diabetic retinopathy using deep learning. *Ophthalmology* 2017; 124 (7): 962-969. doi: 10.1016/j.ophtha.2017.02.008
- [9] Wang K, Luo J. Detecting visually observable disease symptoms from faces. *EURASIP Journal on Bioinformatics and Systems Biology* 2016; 2016: 13. doi: 10.1186/s13637-016-0048-7
- [10] Salam AA, Khalil T, Akram MU, Ameel A, Basit I. Automated detection of glaucoma using structural and non-structural features. *SpringerPlus* 2016; 5 (1): 1519. doi: 10.1186/s40064-016-3175-4
- [11] Gunay M, Goceri E, Danisman T. Automated detection of adenoviral conjunctivitis disease from facial images using machine learning. In: 2015 IEEE 14th International Conference on Machine Learning and Applications; Miami, FL, USA; 2015. pp. 1204-1209. doi: 10.1109/ICMLA.2015.232

- [12] Dalal N, Triggs B. Histograms of oriented gradients for human detection. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition; San Diego, CA, USA; 2005. pp. 1-20. doi: 10.1109/CVPR.2005.177
- [13] Carcagni P, Coco MD, Leo M, Distante C. Facial expression recognition and histograms of oriented gradients: a comprehensive study. SpringerPlus 2015; 4 (1): 645. doi: 10.1186/s40064-015-1427-3
- [14] Kazemi V, Sullivan J. One millisecond face alignment with an ensemble of regression trees. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition; Columbus, OH, USA; 2014. pp. 1867-1874. doi: 10.1109/CVPR.2014.241
- [15] Hinton GE, Salakhutdinov RR. Reducing the dimensionality of data with neural networks. Science 2006; 313 (5786): 504-7. doi: 10.1126/science.1127647
- [16] Alom MZ, Sidike P, Hasan M, Taha TM, Asari VK. Handwritten Bangla character recognition using the state-of-art deep convolution neural network. Computational Intelligence and Neuroscience 2018; 2018: 13. doi: 10.1155/2018/6747098
- [17] Jeon, WS, Rhee SY. Plant leaf recognition using a convolution neural network. International Journal of Fuzzy Logic and Intelligent Systems 2017; 17 (1): 26-34. doi:10.5391/IJFIS.2017.17.1.26
- [18] Cortes C, Vapnik V. Support-vector networks. Machine Learning 1995; 20 (3): 273-297. doi: 10.1023/A:1022627411411
- [19] Rajkumar N, Jaganathan P. A new RBF kernel based learning method applied to multiclass dermatology diseases classification. In: 2013 IEEE Conference on Information & Communication Technologies; Thuckalay, India; 2013. pp. 551-556. doi: 10.1109/CICT.2013.6558156
- [20] Moore B. Principal component analysis in linear systems: controllability, observability, and model reduction. In: IEEE Transactions on Automatic Control 1981; 26 (1): 17-32. doi: 10.1109/TAC.1981.1102568
- [21] Maaten LVD, Hinton G. Visualizing data using t-SNE. Journal of Machine Learning Research 2008; 9: 2579-2605.
- [22] Sagonas C, Tzimiropoulos G, Zafeiriou S, Pantic M. A semi-automatic methodology for facial landmark annotation. In: CVPRW '13 Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition Workshops; Washington, DC, USA; 2013. pp. 896-903. doi: 10.1109/CVPRW.2013.132
- [23] Sagonasa C, Antonakosa E, Tzimiropoulous G, Zafeiriou S, Pantic M. 300 faces in-the-wild challenge: database and results. Image and Vision Computing 2016; 47: 3-18. doi: 10.1016/j.imavis.2016.01.002