ESPE

UNIVERSIDAD DE LAS FUERZAS ARMADAS

INNOVACIÓN PARA LA EXCELENCIA

DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA

CARRERA DE INGENIERÍA EN ELECTRÓNICA E INSTRUMENTACIÓN

TRABAJO DE TITULACIÓN, PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN ELECTRÓNICA E INSTRUMENTACIÓN

TEMA: “DETECCIÓN DE OBJETOS EN TIEMPO REAL A TRAVÉS DE REDES NEURONALES ARTIFICIALES”

AUTORES:

PABLO SEBASTIÁN AULESTIA ARAUJO

JONATHAN SAUL TALAHUA REMACHE

DIRECTOR: ING. VÍCTOR H. ANDALUZ PhD.

LATACUNGA

2017

DEDICATORIA

Dedico este trabajo de investigación a mi amada familia y amigos, por darme su apoyo incondicional y creer en mí a cada instante.

A ti Madre querida, por ser mi fuente de motivación e inspiración para poder superarme siguiendo tu ejemplo de lucha día a día y quien me enseñó a perseverar en cada proyecto emprendido, saliendo de mi zona de confort y venciendo mis miedos.

Incluyo en este homenaje a mi Danny querida, quien fue la persona más importante en mi existencia y sobre todo mi Ángel de la Guarda durante toda mi carrera, a quien me encomendé siempre para lograr cada uno de mis objetivos.

A mi hermana Vicky por demostrarme su un apoyo incondicional al ser siempre mi motor de vida, confidente y mi mejor amiga.

Y a ti mi traviesa Nachita, quien llegó en el momento justo para ganarse mi corazón, darme el aliciente que yo más necesitaba en esta etapa de mi vida, tu amor y tu compañía.

-Sebastián Aulestia

DEDICATORIA

A mis padres, Angel y Rosa, gracias a ellos soy lo que soy, por su apoyo incondicional desde siempre, por la comprensión y consejos brindados cuando más los necesitaba, este triunfo es suyo, se los dedico de todo corazón.

-Jonathan Talahua

AGRADECIMIENTO

Agradezco a Dios, por darme su fortaleza y bendición, guiándome por el camino del bien, pero sobre todo por darme las fuerzas necesarias para no desmayar en cada uno de mis sueños.

Agradezco a mis padres por haberme impartido cada uno de sus valores humanos y morales que me permitirán ser un digno profesional.

Finalmente agradecer a la Universidad de las Fuerzas Armadas – ESPE y en especial a mi tutor el Ing. Víctor H. Andaluz y a mis mentores Ing. Marco Benalcázar e Ing. David Rivas quienes, con su inmensa sabiduría, paciencia y profesionalismo, han sido parte fundamental en la culminación de este proyecto.

-Sebastián Aulestia

AGRADECIMIENTO

A Dios, por haberme permitido llegar hasta este punto, por darme salud e iluminar mi mente para lograr mis objetivos.

A mi madre y padre, por darme la vida, por ser ejemplo de perseverancia y constancia. Gracias por creer en mí, por haberme apoyado en todo momento, por sus consejos, sus valores y por la motivación constante que me ha permitido ser una persona de bien y más que nada, por su amor.

A mis hermanos Maribel y Jordan, por ser un pilar fundamental en mi vida y estar conmigo en los momentos más difíciles

A mis amigos que nos apoyamos mutuamente en nuestra formación profesional.

Finalmente, a mi tutor Ing. Víctor H. Andaluz y mis mentores: Ing. Marco E. Benalcázar e Ing. David Rivas, quienes con su infinita sabiduría me asesoraron en la elaboración de este trabajo.

-Jonathan Talahua

Real-Time Face Detection Using Artificial Neural Networks

Pablo S. Aulestia[1], Jonathan S. Talahua[1], Víctor H. Andaluz[1], and Marco E. Benalcázar[2(✉)]

[1] Universidad de las Fuerzas Armadas ESPE, Sangolquí, Ecuador
{psaulestia, jstalahua, vhandaluzl}@espe.edu.ec
[2] Departamento de Informática y Ciencias de la Computación, Escuela Politécnica Nacional, Quito, Ecuador
marco.benalcazar@epn.edu.ec

Abstract. In this paper, we propose a model for face detection that works in both real-time and unstructured environments. For feature extraction, we applied the HOG (Histograms of Oriented Gradients) technique in a canonical window. For classification, we used a feed-forward neural network. We tested the performance of the proposed model at detecting faces in sequences of color images. For this task, we created a database containing color image patches of faces and background to train the neural network and color images of 320 × 240 to test the model. The database is available at http://electronica-el.espe.edu.ec/actividad-estudiantil/face-database/. To achieve real-time, we split the model into several modules that run in parallel. The proposed model exhibited an accuracy of 91.4% and demonstrated robustness to changes in illumination, pose and occlusion. For the tests, we used a 2-core-2.5 GHz PC with 6 GB of RAM memory, where input frames of 320 × 240 pixels were processed in an average time of 81 ms.

Keywords: Real-time face detection · Histograms of oriented gradients · Feed-forward neural networks

1 Introduction

Computer vision consists of a set of algorithms which allow us to analyze the content of digital images through mathematical models that emulate the human visual system. Object detection is one of the main problems of computer vision and is the base to implement algorithms to interpret and understand the dynamic world using color, grayscale or binary images. Computer vision represents a great challenge, especially when we try to interpret or understand an image or sequences of images (i.e., video) automatically. The main applications of computer vision include the identification and localization of objects in given space, search and tracking of objects for autonomous robots, and image restoration. Therefore, computer vision is an important field to do research and object detection is a key topic for developing new algorithms.

A. Lintas et al. (Eds.): ICANN 2017, Part II, LNCS 10614, pp. 590–599, 2017.
https://doi.org/10.1007/978-3-319-68612-7_67

An object detection system is composed of the following modules: image acquisition and preprocessing, feature extraction, classification and refinement of the detection. The feature extraction problem has been extensively addressed using different methods such as extraction of edges, local binary patterns, segmentation and blending of color spaces [1–3]. For example, in [4] authors propose the use of local binary patterns (LBP). Histograms of oriented gradients (HOG) [5] and discrete-time filters based on the HAAR wavelet transform [6] have also been used. The goal of these methods is to represent a digital image in a given n-dimensional space of characteristics. For a given application, we want the feature extractor module to be robust to changes of illumination, orientation or position [7].

The classification stage implements a decision boundary to separate the different object classes of a given image. The most common and classical methods for classification include the use of support vector machines (SVMs), k-nearest neighbors (kNN), cascade classifiers, and logistic regression. These classifiers combined with the feature extractors described above work well in images whose background is a structured or a partially structured environment [8–13]. On the other hand, when the conditions of the environment change, the performance of the detectors worsen. For example, in [8–11] the performance of the classifiers is less than 94% and the detection accuracy of the whole systems is lower than 90%. In [12, 13] the processing time per image is greater than 200 ms. Additionally, there are other contributions [14–18] that show a performance over 96% using advanced classifiers such as convolutional neural networks (CNN). However, for training and testing these detectors, GPUs have been used to accelerate the computations.

In the literature review presented in this paper, we can see that there exist object detection models with relatively low computational cost. These models with relatively simple structure perform well in structured or partially structured environments. However, when the conditions of the environment change, their performance worsen. On the other hand, there are complex models that exhibit good performance but demand of high computational resources to be trained and tested. Therefore, more research is needed to develop simple object detection models that exhibit both low computational cost and good performance simultaneously.

In this work, we propose a real-time face detection system for unstructured environments. The input to our system is a sequence of images (i.e., video). For feature extraction, we use the HOG descriptor. For classification, we use a feed-forward artificial neural network (ANN). We use a 2-core-2.5 GHz PC with 6 GB of RAM memory to test the proposed model. The algorithm was split into several modules that run in parallel to achieve real-time processing.

Following this introduction, in Sect. 2 we describe the materials and methods used for the feature extraction and classification stages. In Sect. 3, we present the experimental results of the complete system. Finally, in Sect. 4 we present the conclusions and future work.

2 Materials and Methods

2.1 Materials

As an application case of the proposed model, we considered the problem of face detection. To address this problem, we took 7117 photographs to create a database of color images. To train the ANN, we used 5750 color images and the remaining 1367 images were used to validate the system of classification and detection. Out of the 5750 images, 2750 images contain faces with a frontal pose and the remaining 3000 images are background (Fig. 1). The training images are patches with a size of 64 × 64 pixels. The images for testing have a size of 640 × 480 pixels. Both sets of images are represented in the RGB color space and are in the JPG format. These images were taken from students and staff of the Universidad de las Fuerzas Armadas ESPE-Ecuador. The age range of the people that were photographed is between 12 to 50 years. This data set is available at http://electronica-el.espe.edu.ec/actividad-estudiantil/face-database/.

Fig. 1. Examples of faces.

2.2 Face Detection

The first step of the proposed face detection model consists of extracting an image patch I from the original image. We assume this patch belongs to the class $Y \in \{0, 1\}$, where $Y = 1$ and $Y = 0$ represent the face and non-face classes, respectively. The image patch I is obtained by observing the original image through a canonical window W, whose upper left corner is located at the pixel $p = (x, y)$. Second, from the patch I, we extract a feature vector $\mathbf{X}$ using the HOG technique. Third, the vector $\mathbf{X}$ is fed to classifier $\psi : \mathbf{X} \rightarrow \{0, 1\}$ based on an ANN. Fourth, if the result of $\psi(\mathbf{X})$ is 1, then a bounding box of the same size as W is placed at $p = (x, y)$ (Fig. 2). Fifth, we shift W to the pixel $p + \Delta p$ in the original image and repeat the previous steps. The value of Δp controls the overlapping between adjacent windows. Finally, to deal with objects of different sizes, we generate a pyramid of images by iteratively resizing the original image until W contains the object of interest within its limits. In Fig. 3, we show a pyramid of six images by reducing the size of the original image by a factor of 1.1.

We chose a canonical window W of 64 × 64 pixels. This size is a tradeoff between the size of the objects that can be detected and the computational cost of the proposed

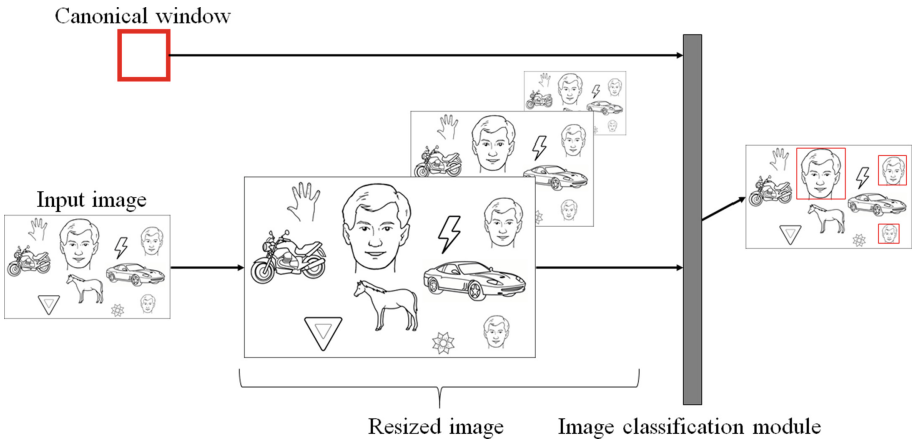

Fig. 2. Illustration of a face detection system, where the faces detected are enclosed in bounding boxes.

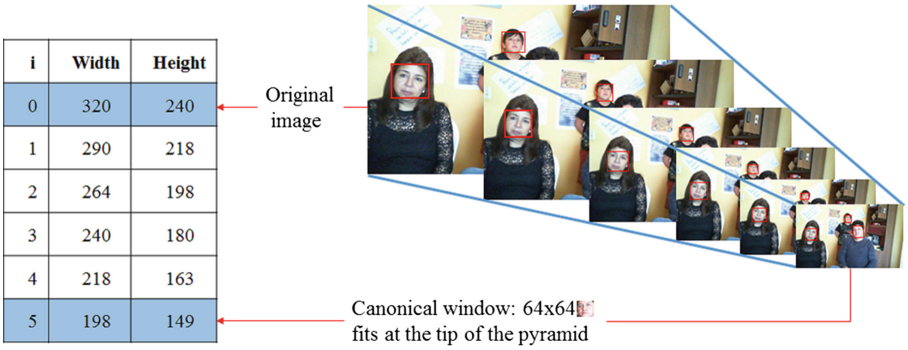

Fig. 3. Pyramid of images obtained by reducing the size of the original image by $s = 1.1$

system. We assume the proposed system will operate in scenarios where the objects of interest are located at a maximum distance of 2 m from the camera.

2.3 Histograms of Oriented Gradients

We used the HOG descriptor because it provides information about the orientations of the edges that dominate each position of the image. This method is also invariant to changes of illumination, pose and occlusion of the object to detect.

Gradient Calculation: The vertical and horizontal gradients of a pixel $p = (x, y)$ of an image I are $dx = I(x+1, y) - I(x-1, y)$ and $dy = I(x, y+1) - I(x, y-1)$, respectively. We calculate the orientation and magnitude of these gradients with $\theta(x, y) = arctan(dy/dx)$ and $g(x, y) = \sqrt{dx^2 + dy^2}$, respectively.

Histogram Calculation: We split the input image I into non-overlapping cells of 8×8 pixels each (Fig. 4a). Then, we split the orientation between 0° and 180° into 9 intervals (Fig. 4b). Next, we calculated the value of the histogram h at the interval k, $h(k)$, by accumulating the gradients of the cell C using the following equation:

$$h(k) = \sum_{(x,y)\in C} w_k(x,y)g(x,y), \tag{1}$$

where $k = 1, 2, \ldots, 9$, $w_k(x,y) = 1$ if $20 * (k-1) \leq \theta(x,y) < 20 * k$ and $w_k(x,y) = 0$ otherwise. Next, we concatenated the histograms of each cell inside a block of 2×2 cells obtaining thus the vector $v' = (h_1, \ldots, h_4)$, where h_i, $i = 1, 2, 3$, and 4, denotes the histogram of the i^{th} cell in a given block (Fig. 4c). Then we normalized the vector v' using the L2 norm obtaining the vector v. Finally, to obtain the one-dimensional HOG vector, we concatenated all the normalized vectors v into the new vector $\mathbf{X} = (v_1, v_2, \ldots, v_n)$, where n is the total number of blocks in a patch I. With these configurations, the length of a HOG vector for an image patch of 64×64 pixels is 1764.

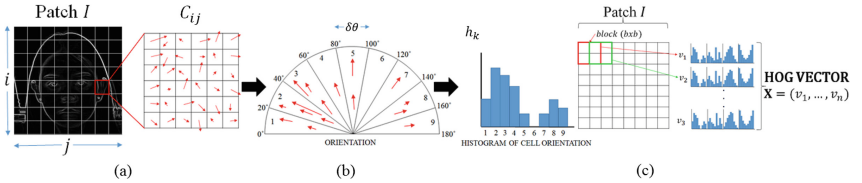

Fig. 4. Illustration of the feature extraction stage using the HOG technique: (a) division of the image patch into non-overlapping cells, (b) histogram of each cell, and (d) feature vector for an image patch.

2.4 Classification

For the classification stage, we used a feed-forward ANN because this model is a universal function approximator [19]. The ANN we used in this work has three layers: input $L^{(0)}$, hidden $L^{(1)}$ and output $L^{(2)}$. The hidden layer is composed of m neurons, and a sigmoid transfer function $f^{(1)}$. The output layer is composed of a single neuron with a sigmoid transfer function. This structure can be seen in Fig. 5. The response $\mathbb{P}(Y = 1|\mathbf{X})$ of the ANN, for an input $\mathbf{X} = (v_1, \ldots, v_n)$, is given by the following expression:

$$\mathbb{P}(Y = 1|\mathbf{X}) = f^{(2)}[W^{(2)}f^{(1)}(W^{(1)}\mathbf{X} + b^{(1)}) + b^{(2)}], \tag{2}$$

where $\mathbb{P}(Y = 1|\mathbf{X})$ denotes the conditional probability that $\mathbf{X}$ belongs to a face, $W^{(1)}$ and $W^{(2)}$ denote the matrices of weights of the layers $L^{(1)}$ and $L^{(2)}$, respectively, and $b^{(1)}$ denotes the bias vector for neurons of the layer $L^{(1)}$, and $b^{(2)}$ is the bias for the neuron of the layer $L^{(2)}$.

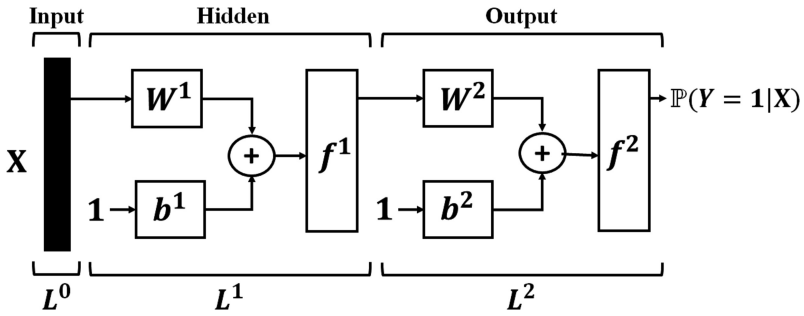

Fig. 5. Architecture of a feed-forward artificial neural network of three layers.

The activation functions that we used in the artificial neural network are $f^{(1)} = f^{(2)} = logsig(z) = 1/(1+e^{-z})$. To train the ANN, we created a database with patches of faces and backgrounds. From these patches, we extracted their corresponding HOG vectors. If a vector $\mathbf{X}$ belongs to a face, we labeled it with $Y = 1$, otherwise, we labeled it with $Y = 0$. In this way, we obtained a training set $\mathcal{D} = \{(\mathbf{X}_1, Y_1), \ldots, (\mathbf{X}_N, Y_N)\}$ composed of N examples. Then, we use the full-batch back-propagation and the gradient-descent algorithms to minimize the cost function $-\ln[\mathbb{P}(\mathcal{D}|\beta)]$, where $\mathbb{P}(\mathcal{D}|\beta)$ denotes the likelihood of the training set $\mathcal{D}$ given the parameters $\beta = \{W^{(1)}, W^{(2)}, b^{(1)}, b^{(2)}\}$ of the ANN [20].

3 Experimental Results

3.1 ANN Training

For training the neural network, we used a training set composed of 5750 image patches. The hidden layer of the ANN was composed of 4 neurons. With these configurations, we obtained a training error of 0.2% after 100 epochs.

3.2 Validation of the Classification Module

To evaluate the performance of the classifier of the proposed approach, we used 959 images divided in two cases: positive (faces) and negative (non-faces). The ROC (Receiver Operating Characteristic) curve was used to analyze the results. The AUC (area under curve) for the classification module has a value of 97.4% (Fig. 6a). The variation step of the threshold to obtain the ROC curve was of 0.001.

3.3 Validation of the Complete System

The system detects faces at six different scales. The input of the system is a video frame of 320 × 240 pixels (scale 1). To obtain the remaining 5 scales, we reduced the size of the original image by a factor of 1.1. For feature extraction, we used a canonical window of 64 × 64 pixels. The face detection at each scale was done in parallel to

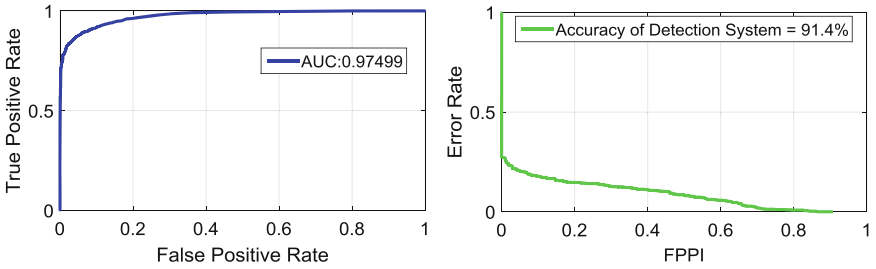

Fig. 6. (a) ROC curve of the classification module AUC = 0.97, (b) error rate versus false positives per image (FPPI) for the whole system.

reduce the computational time and achieve real-time. We tested the system in different environments to verify the robustness of the proposed approach to changes of illumination, pose and occlusion as shown Fig. 7. The first row shows that the system detects the faces in different light conditions. In the second row, the faces were detected even though the light conditions are very low and there are occlusions on the object of interest.

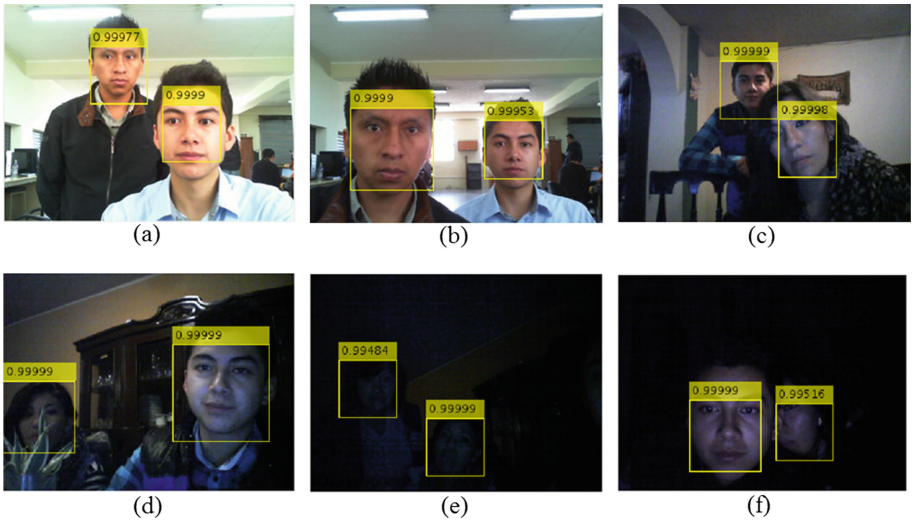

Fig. 7. Face detection results for different lights conditions: (a) high, (b) moderate (c) medium with overlapping of faces, (d) medium with occlusion, (e) low, and (f) low, with overlapping faces.

The performance of the whole system is 91.4% (Fig. 6b). This performance is lower than the performance of the classification module because the complete system is composed of an additional module to refine the detections. This module eliminates all the detections that overlap in more than 10%, except for the one with the highest

probability. Additionally, the performance of the whole system also decreases because of an increase in the rate of false negatives. This increase of false negatives occurs because there are faces in the test set that do not fit at any scale within the canonical window of 64 × 64 pixels.

In Table 1, we show the averages of the time that it takes for each module of the system to process a video frame. We can see that the average detection time per video frame is 81 ms. This value shows a relatively high speed compared to other detectors [14, 18].

Table 1. Averages of the time of the modules that compose the proposed system.

Step	Average time
Search of potential faces at different scales	31 ms
Feature extraction	16 ms
Classification	5 ms
Refinement of the detection	29 ms
Total	81 ms

These results obtained in this work evidence that our model is robust to changes of light conditions. Although the detection of the proposed system is limited by the maximum distance to which the object of interest can be located from the camera, the system can correctly detect a face, even when there is occlusion. There are several works that use traditional methods to detect faces, obtaining recognition accuracies higher than 90% [14–18]. However, most of these works evaluate their systems with still images and not with video frames. Additionally, some of these works are tested in highly controlled environments, where the light conditions are roughly the same among all the test images. Therefore, variations of brightness, which are not a problem for our system, are a limiting factor for these models.

4 Conclusions and Future Work

In this work, we have presented a real-time object detection system. We used a feature extractor based on the histograms of oriented gradients. The classifier is a feed-forward neural network with 4 neurons in the hidden layer and 1 neuron in the output layer. We tested this system at detecting faces, showing a detection rate of 91.4%. Even though we used a shallow neural network with only 4 neurons in the hidden layer, we obtained high performance in unstructured environments that included variations in brightness, pose and occlusion. We tested the proposed model using not sophisticated computational resources. The average processing time of the complete algorithm is 81 ms for each video frame of 320 × 240 pixels. To achieve this speed, we ran the different scales of detection in parallel, combining high and low level languages (MATLAB and C++). We also make publicly available the training and testing sets we used for this work at http://electronica-el.espe.edu.ec/actividad-estudiantil/face-database/. Future

work includes testing other classifiers different from neural networks. Additionally, we will also test the proposed model at detecting other type of objects different from faces.

Acknowledgment. The authors thank the Consorcio Ecuatoriano para el Desarrollo de Internet Avanzado -CEDIA-, and the Universidad de las Fuerzas Armadas -ESPE- for supporting the development of this work.

References

1. Gil, P., Torres, F., Ortiz, F.: Detección de objetos por segmentación multinivel combinada de espacios de color. Federación Internacional de Automatización, Real (2004)
2. Canny, J.: A computational approach to edge detection. Trans. Pattern Anal. Mach. Intell. **8**, 679–698 (1986). IEEE
3. Zhao, G., Pietikaen, M.: Dynamic texture recognition using local binary patterns with an application to facial expressions. Trans. Pattern Anal. Mach. Intell. **29**, 915–928 (2007). IEEE
4. Wolf, L., Hassner, T., Taigman, Y.: Descriptor based methods in the wild. In: Workshop on Faces in 'Real-Life' Images: Detection, Alignment, and Recognition, pp. 1–14, France (2008)
5. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: Computer Vision and Pattern Recognition, pp. 886–893. IEEE, Francia (2005)
6. Tang, J., Gongjian, W.: Object recognition via classifier interaction with multiple features. In: 2016 8th International Conference on Intelligent Human-Machine Systems and Cybernetics, pp. 337–340. IEEE, China (2016)
7. Nixon, M., Aguado, A.: Feature Extraction and Image Processing for Computer Vision, pp. 218–220. Academic Press, London (2012)
8. Viola, P., Jones, M.: Robust real-time face detection. Int. J. Comput. Vis. **57**, 137–154 (2004). Springer, The Netherlands
9. Schroff, F., Kalenichenko, D., Philbin, J.: Facenet: a unified embedding for face recognition and clustering. In: The IEEE Conference on Computer Vision and Pattern Recognition, pp. 815–823. IEEE, Boston (2015)
10. Guillaumin, M., Verbeek, J., Schmid, C.: Is that you? Metric learning approaches for face Identification. In: 12th International Conference on Computer Vision, pp. 498–505. IEEE, Kyoto (2009)
11. Cheng, W., Hüllermeier, E.: Combining instance-based learning and logistic regression for multilabel classification. Mach. Learn. **76**, 211–225 (2009). Springer
12. Fasel, I., Fortenberry, B., Movellan, J.: A generative framework for real time object detection and classification. Comput. Vis. Image Underst. **98**, 182–210 (2005). Elsevier
13. Ren, S., He, K., Girshick, R. and Sun, J.: Faster R-CNN: towards real-time object detection. In: Neural Information Processing Systems Conference, pp. 91–99 (2015)
14. Liu, Y., Cao, Y., Li, Y.: Facial expression recognition with PCA and LBP features extracting from active facial patches. In: IEEE International Conference on Real-time Computing and Robotics, pp. 1–6. IEEE, Angkor Wat (2016)
15. Jia, J., Xu, Y., Zhang, S., Xue, X.: The facial expression recognition method of random forest based on improved PCA extracting feature. In: 2016 IEEE International Conference on Signal Processing, Communications and Computing, pp. 1–5. IEEE, Hong Kong (2016)

16. Abdulrahman, M., Gwadabe, T., Abdu, F., Eleyan, A.: Gabor wavelet transform based facial expression recognition using PCA and LBP. In: Signal Processing and Communications Applications Conference, pp. 1–4. IEEE, Trabzon (2014)
17. Taigman, Y., Yang, M., Ranzato, M., Wolf, L.: DeepFace: closing the gap to human-level performance in face verification. In: The IEEE Conference on Computer Vision and Pattern Recognition, pp. 1701–1708. IEEE (2014)
18. Lagerwall, B., Viriri, S.: Robust real-time face recognition. In: Proceedings of the South African Institute for Computer Scientists and Information Technologists Conference, pp. 194–199. ACM New York, East London (2013)
19. Hornik, K., Stinchcombe, M., White, H.: Multilayer feedforward networks are universal approximators. Neural Netw. **2**, 359–366 (1989)
20. Hagan, M., Menhaj, M.: Training feedforward networks with the marquardt algorithm. IEEE Trans. Neural Netw. **5**, 989–993 (1994)