sustainability

MDPI

Article

Entropy-Based Face Recognition and Spoof Detection for Security Applications

Francisco A. Pujol [1,*], María José Pujol [2], Carlos Rizo-Maestre [3] and Mar Pujol [4]

[1] Department of Computer Technology, University of Alicante, 03690 San Vicente del Raspeig-Alicante, Spain
[2] Department of Applied Mathematics, University of Alicante, 03690 San Vicente del Raspeig-Alicante, Spain; mjose@ua.es
[3] Department of Architectural Constructions, University of Alicante,
03690 San Vicente del Raspeig-Alicante, Spain; carlosrm@ua.es
[4] Department of Computer Science and Artificial Intelligence, University of Alicante,
03690 San Vicente del Raspeig-Alicante, Spain; mar@dccia.ua.es
* Correspondence: fpujol@ua.es

Received: 24 October 2019; Accepted: 16 December 2019; Published: 20 December 2019

check for updates

Abstract: Nowadays, cyber attacks are becoming an extremely serious issue, which is particularly important to prevent in a smart city context. Among cyber attacks, spoofing is an action that is increasingly common in many areas, such as emails, geolocation services or social networks. Identity spoofing is defined as the action by which a person impersonates a third party to carry out a series of illegal activities such as committing fraud, cyberbullying, sextorsion, etc. In this work, a face recognition system is proposed, with an application to the spoofing prevention. The method is based on the Histogram of Oriented Gradients (HOG) descriptor. Since different face regions do not have the same information for the recognition process, introducing entropy would quantify the importance of each face region in the descriptor. Therefore, entropy is added to increase the robustness of the algorithm. Regarding face recognition, our approach has been tested on three well-known databases (ORL, FERET and LFW) and the experiments show that adding entropy information improves the recognition rate significantly, with an increase over 40% in some of the considered databases. Spoofing tests has been implemented on CASIA FASD and MIFS databases, having obtained again better results than similar texture descriptors approaches.

Keywords: face recognition; security; spoofing; histogram of oriented gradients; smart cities

1. Introduction

Biometrics relies on measuring different human characteristics and matching them to previously collected measurements in a database. Biometric features are “built-in” the body so they cannot be shared or subtracted. Even though these systems seem to be extremely reliable, it is always possible to capture a legitimate biometric trait from a user, copy it and replicate it later by someone else. The act of using an artifact to fool a biometric system, where someone pretends to be another person, is known as spoof attack [1].

Nowadays, fingerprint hardware systems represent more than 92% of the total biometric features market [2]. With the rise of facial identification in mobile systems, experts forecast that annual facial recognition devices and licenses will increase from $28.5 mn in 2015 to more than $122.8 mn worldwide by 2024. During that period, annual revenue for facial biometrics, including both visible light facial recognition and infrared-based facial thermography, will increase from $149.5 mn to $882.5 mn, at a compound annual growth rate (CAGR) of 22% [3].

In the last few years, technology regarding image capturing has evolved, allowing consumers to buy high resolution cameras at a very low cost, specially with the use of billions of smartphones that allow users to have a digital camera on their hand constantly. Taking advantage of this situation, it seems straightforward to prevent spoofing by authenticating using biometric sensors such as fingerprints, iris or facial features. The annual revenue from mobile biometrics systems and applications is expected to grow from $6.5 bn in 2016 to $50.6 bn in 2022, with a compound annual growth rate of 41% [4].

Biometric systems can be compromised and are vulnerable to a wide range of attacks. Among all these potential attacks, the one with the greatest practical relevance is the spoof attack. As mentioned before, it consists of submitting a stolen or copied biometric trait to the sensor in order to defeat the biometric system and access the system in an unauthorized way. These attacks don't need any knowledge about the security system itself because, if the authorized user is able to access, the attacker just needs to simulate the biometric trait of that user. Because of this, most security systems provide some kind of protection such as hashing, digital signature or encryption that are ineffective in spoof attacks [5]. In the last few years there has been an intensive research to provide reliable anti-spoofing systems for biometric traits, including fingerprints [6,7], face [8–10], and other biometric features [11–13].

Spoofing attacks have grown exponentially in the last few years [14–16]. Among other areas, social networks have recently reported serious privacy and security issues [17–19]. Therefore, to ensure a higher security level in social networks it would be convenient to implement some kind of spoofing detector. However, using a supervised control of all the information on a social network is unfeasible due to the huge amount of data that can be produced at any given time. One of the most common ways cyberbullying develops is through identity spoofing. In this case, false user profiles attributed to the victim can be created. It may also be possible to access the user's profile or personal account on different social networks in such a way that the identity is spoofed by contacting others or making comments on behalf of the victim of bullying.

Consequently, the main objective of this work is to propose and develop a face recognition and spoof detection method that can be applied on social media by means of a novel entropy-based system. Entropy has been used in face recognition in recent years [20,21]. Thus, as different areas of a face image contribute in a different way to the global recognition, entropy on each area is introduced to construct a new version of the Histogram of Oriented Gradients (HOG) descriptor. As a result, the main contribution of this paper is the introduction of this new HOG-based descriptor, which makes use of entropy to code each area in a face image. To do this, after the Entropy-Based Histogram of Oriented Gradients descriptor is computed, Support Vector Machines are used for the classification process. Our system has been tested with three face recognition databases (Olivetti Research Laboratory: ORL, FERET and Labeled Faces in the Wild: LFW) and two face spoof detection datasets (CASIA Face Anti-Spoofing Database:FADS and Makeup Induced Face Spoofing: MIFS dataset), obtaining reliable results and outperforming other recent works using texture descriptors on the same databases.

This paper is organized as follows: Section 2 summarizes some related works; Section 3 explains our proposal of face detection, recognition and spoofing detection, introducing the Entropy-Based Histogram of Oriented Gradients (EBHOG) descriptor; Section 4 describes the experimental setup and the set of experiments completed with different databases; and finally, conclusions and some future works are discussed in Section 5.

2. Related Work

On social networks, a great amount of pictures and videos are uploaded and shared every day, where users can post an image where someone else appears without his/her consent. A face detection and recognition system would act before the image is published, identifying the people appearing in the image and notifying those users to give consent, protecting their privacy and increasing security by reporting potential cases of spoofing.

In order to simulate the authorized user, some face recognition systems can be spoofed by showing a photograph, video or even a face model of the user. Spoofing attacks can be detected using several methods. When detection and recognition are required to work in real-time, they must be computationally inexpensive. Most of the recognition methods are not fast enough or use non-conventional images [22]. Moreover, due to social image sharing and social networking websites, personal facial photographs of many users are usually accessible to the public. For instance, an impostor can obtain the photographs of genuine users from a social network, and submit them to a biometric authentication system to fool it [23].

Over the last few years, a wide variety of feature representation methods have been proposed to help describe scenes, objects and biometric features in different images. The particularities of each of these methods describe different aspects of visual features, each being best suited to certain particular conditions. Some of these methods focus on local information, others on holistic descriptors. Among all local feature descriptors, the most commonly used are SIFT (Scale-Invariant Feature Transform) [24,25], HOG (Histogram of Oriented Gradients) [26,27], SURF (Speeded-up Robust Features) [28,29] and LBP (Local Binary Patterns) [30,31], which are used to address variability in the image caused by changes in perspective, occlusions and variation in brightness.

As in many other machine learning applications, deep learning methods have proven to be an effective way to detect spoofing attacks. Many related works have considered face spoofing as a binary classification problem, where the system classifies a face as belonging to either a legitimate user or a fake user [32,33]. Thus, in [34] authors use CaffeNet and GoogLeNet convolutional neural networks (CNNs) models and perform a texture analysis. Alotaibi and Mahmood [35] presented a nonlinear diffusion to distinguish a fake image from a real image, which is the applied to a CNN for face liveness detection. Finally, an LBP network for face spoofing detection is proposed in [36], where LBP features are combined with deep learning. In spite of the immense potential of deep learning methods, they are computationally expensive, they need extremely large datasets for training and their internal complexity makes it difficult in some applications to interpret the results or to understand the algorithm mechanism [37–39].

The HOG descriptor is one of the most popular approaches for object detection. It is invariant to illumination and geometric transformations and it has been successfully applied to many security applications, such as privacy in image feature extraction using homomorphic encryption [40], phishing detection [41], classification of sensitive information embedded within uploaded photos [42], handwritten digits recognition [43], facial expression recognition with CNNs [44] and, particularly, to face spoofing detection [11,45–48]. Due to its popularity in anti-spoofing detection, in this work a variant of the HOG descriptor will be presented and experimentally validated.

3. Materials and Methods

The proposed system will detect and extract faces in a set of images, recognize extracted faces by matching them against the ones stored in a database and perform experimental proofs of the proposed method to enhance security. In order to validate the system, images with only one face will be considered. A description of the developed system is in Figure 1.

The whole process consists of the following stages:

- Image acquisition: image retrieval from a still photo.
- Face detection: detection of some patterns in the image in order to locate a face.
- Image pre-processing: crop the image to remove irrelevant parts and, if needed, apply image processing (change color space, filtering, etc.) to enhance some parameters to be measured in the next stage.
- Identifiers extraction: calculation of coefficients or identifying characteristic values.
- Face recognition: comparison of coefficients of the faces in the database and a new input image to verify identity.

Let us explain next the key features of our proposal.

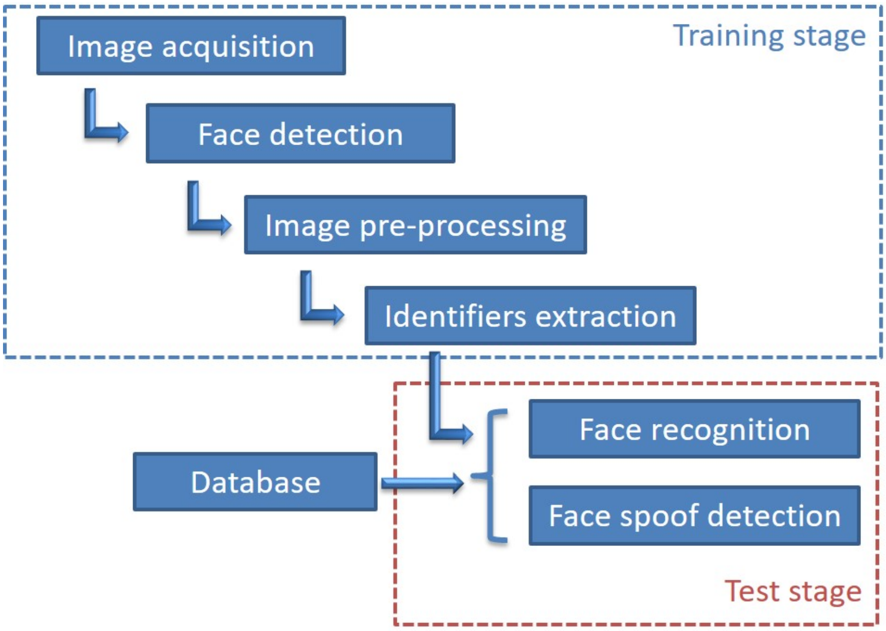


Figure 1. System overview.

3.1. Detection Framework

Regarding the detection framework, we have used a well-known object detection algorithm developed by Paul Viola and Michael Jones [49]. It is a robust algorithm with a very high detection rate, suitable for real time applications that can be used for face detection. The algorithm uses four stages to enable a fast and accurate detection: Haar feature selection, the integral image for feature computation, AdaBoost for feature selection and an attentional cascade for efficient computational resource allocation [50].

After this process, original images are cropped so that only face images are taken into account in the following stages of our proposal. Then images are normalized in size and color information is removed, since our system will work with grayscale images.

3.2. Recognition Framework

For the recognition process, the Histogram of Oriented Gradients (HOG) has been considered. HOG is a well-known image descriptor based on the image's gradient orientations. HOGs are rotationally-invariant image descriptors that have been used in optimization problems as well as in computer vision. The Histogram of Oriented Gradients (HOG) method has proven to be an effective descriptor, in general, for object recognition and for face recognition in particular [27].

The method is based on evaluating local histograms of image gradient orientations in a dense grid. The basic idea is that local object appearance and shape can often be characterized by the distribution of local intensity gradients or edge directions, even without precise knowledge of the corresponding gradient or edge positions [51]. HOG counts occurrences of edge orientations in a neighborhood of an image. In practice, this is implemented by dividing the image window into small spatial regions ("cells"), and each cell accumulates a local 1-D histogram of gradient directions (or edge orientations) over the pixels of the cell. The combined histogram entries form the representation [26].

Let G_x and G_y be the horizontal and vertical gradients of the image I. They can be computed for each pixel (x, y) using simple 1-D masks as follows:

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

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

Then, the magnitude and orientation of the gradient are calculated as:

$$M(x,y) = \sqrt{G_x^2 + G_y^2} \tag{3}$$

$$\theta(x,y) = \arctan\left(\frac{G_y}{G_x}\right) \tag{4}$$

Histograms are then constructed with the magnitude and orientation of each pixel, so that each cell will have one histogram and they will be concatenated to obtain the feature descriptor. The procedure for the implementation of the HOG descriptor algorithm is shown in Figure 2 and can be summarized as:

- Divide input image into small connected regions (cells).
- For each cell, a histogram of edge orientations is computed for all the pixels in the cell.
- Every cell is discretized into angular bins, according to the gradient orientation.
- Calculate histograms of oriented gradients over spatial cells.
- Group adjacent cells into overlapping blocks and normalize histograms.
- Block HOGs constitute the descriptor.
- Train a classifier (by using SVM -Support Vector Machines-, for instance).
- Classify using the trained SVM.

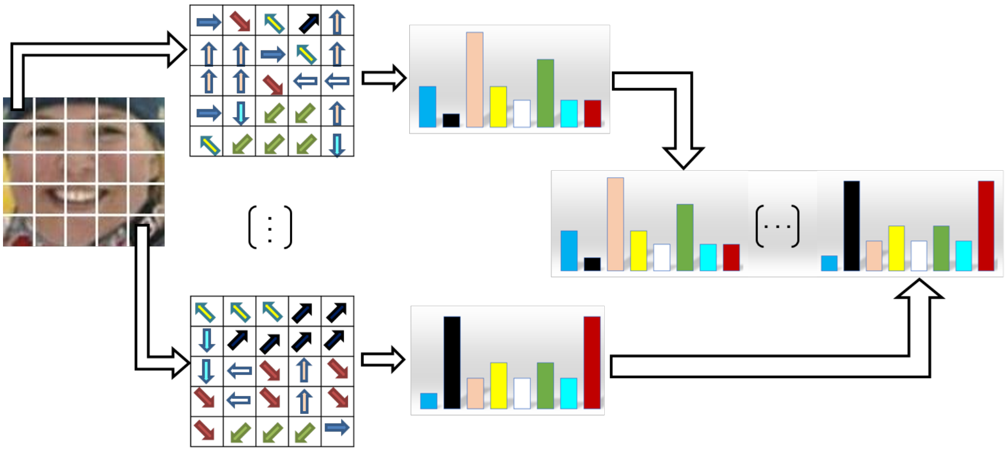


Figure 2. Histogram of Oriented Gradients (HOG) calculation. The orientation of gradients are calculated in each cell and all the histograms are concatenated to obtain the global HOG descriptor.

HOGs give the same importance (or weight) to each block in the image. However, some of these blocks contain information of the most remarkable features for face recognition, such as the eyes, the nose or the mouth [52,53], and many other blocks do not provide significant features for recognition. In other words, not all the blocks give the same information for a face recognition scheme. Shannon introduced entropy as a measure for measuring quantitatively the amount of information produced by a process [54]. Therefore, we consider that using entropy would quantify or weigh the importance of each block in the calculated HOGs, since different face regions will have different weights. Consequently, we introduce the Entropy-Based Histogram of Oriented Gradients (EBHOG) descriptor as follows:

- Divide input image into c small connected regions (cells).
- For each cell c, a histogram of edge orientations is computed for all the pixels in the cell.
- Every cell is discretized into b angular bins, according to the gradient orientation.
- Calculate histograms of oriented gradients over spatial cells.

- Group adjacent cells into overlapping blocks and normalize histograms.
- Calculate Shannon's entropy for each computed HOG. This will give a weight w_k for each block, for $k = 1, 2, \cdots, N$.
- Normalize weighted histograms.
- The weighted HOGs using entropy constitute the descriptor.
- Train a classifier by using SVM.
- Classify using the trained SVM.

The entropy for block k, H_k, is defined as:

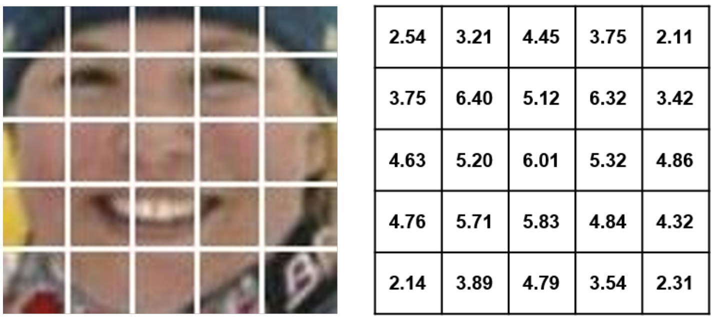
$$H_k = -\sum_{j_k=1}^{N} P_{j_k} \log_2 P_{j_k} \tag{5}$$

where N is the number of blocks in the image and P_{j_k} is:

$$P_{j_k} = \frac{HOG_k(j)}{\sum_{i=1}^{c \times b} HOG_k(i)}. \tag{6}$$

HOG_k indicates the HOG obtained for block k in the input image.

The entropies of different regions in an image are shown in Figure 3. In this figure, it can be noticed that blocks containing key features for recognition, such as eyes, nose or mouth, have significant higher entropy values than blocks with irrelevant information for face recognition.



2.54	3.21	4.45	3.75	2.11
3.75	6.40	5.12	6.32	3.42
4.63	5.20	6.01	5.32	4.86
4.76	5.71	5.83	4.84	4.32
2.14	3.89	4.79	3.54	2.31

Figure 3. Entropy for different blocks in an image.

The weight w_k will then be calculated as:

$$w_k = H_{\max} \cdot \left(\frac{H_k - H_{\min}}{H_{\max} - H_{\min}} \right), \tag{7}$$

where

$$H_{\max} = \max_{\forall k} H_k \tag{8}$$

$$H_{\min} = \min_{\forall k} H_k \tag{9}$$

Let $W = \{w_1, w_2, \cdots, w_N\}$ be the set of weights calculated from Equation (7) and $HOG = \{HOG_1, HOG_2, \cdots, HOG_N\}$ the histograms of oriented gradients for each block. The entropy-based HOG is then calculated by multiplying each HOG_k by its corresponding weight w_k:

$$EBHOG = \{w_1 \times HOG_1, w_2 \times HOG_2, \cdots, w_N \times HOG_N\} \tag{10}$$

After weighting each HOG, the histograms must be normalized again, since the sum of all the values in each entropy weighted histogram is not 1. Thus, if $EBHOG_k = w_k \times HOG_k$ is the

k-th weighted histogram, min_k and max_k are the minimum and maximum values in $EBHOG_k$, the normalized histogram $EBHOG_k^{Norm}$ is:

$$EBHOG_k^{Norm} = \frac{EBHOG_k}{max_k - min_k} \tag{11}$$

The whole training and testing process has been represented in Figures 4 and 5.

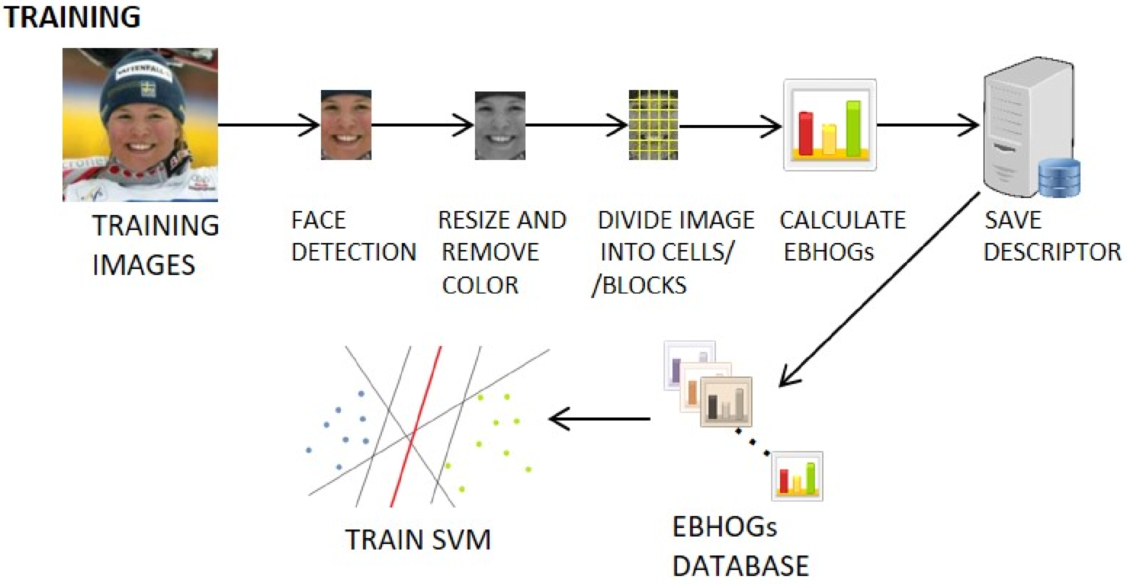


Figure 4. Diagram of the training stage for the face recognition process.

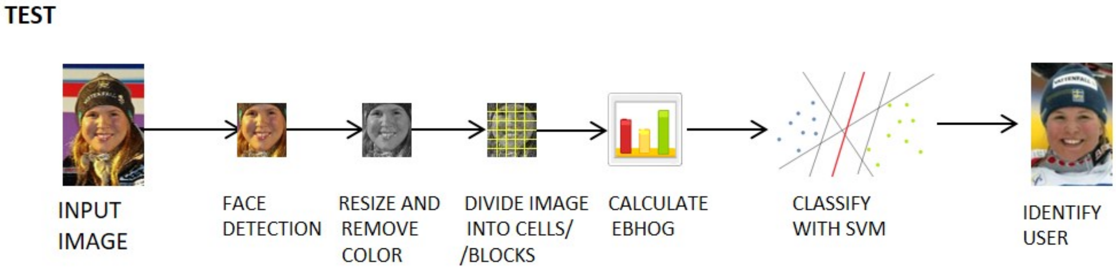


Figure 5. Diagram of the test stage for the face recognition process.

For classification, the Support Vector Machines (SVM) classifier has been chosen. Having several classes to be identified, the main idea of SVM is to select a hyperplane that is equidistant from the examples of each class to achieve the so-called maximum margin on each side of the hyperplane. To define this hyperplane, only the training data of each class next to these margins, which are called support vectors, are taken into account. The search for the separation hyperplane in these transformed spaces, usually of very high dimension, will be done implicitly using the so-called kernel functions. A kernel function $K(\mathbf{x}_i, \mathbf{x}_j)$ is a function that assigns to each pair of elements $\mathbf{x}_i, \mathbf{x}_j \in \mathbb{X}$ of an input space $\mathbb{X}$, a real value corresponding to the scalar product of the transformed version of that element in a new space. Among the most popular kernel functions, one can find:

- Linear kernel, whose expression is:

$$K(\mathbf{x}_i, \mathbf{x}_j) = \langle \mathbf{x}_i, \mathbf{x}_j \rangle \tag{12}$$

where $\langle \cdot, \cdot \rangle$ refers to the scalar product.

- Gaussian kernel, expressed as:

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(\frac{-\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\sigma^2}\right) \tag{13}$$

where σ is standard deviation.

The selection of the kernel function depends on the data to be classified and will be validated in Section 4.

3.3. Detecting Spoofing Using EBHOG

Face spoofing attacks can be performed in general terms by using still images, videos or real faces. In order to apply the EBHOG method, our proposal aims at training models with different textures to detect a real face from fake faces. It can be noticed that fake faces cause some reflections that depend on the surface from where the facial image is being projected, being non-existent if the image were real [55].

Our proposal is based on the results of [9]. In their work, authors showed that introducing color information achieves reliable results to prevent face spoofing attacks. In particular, the YCbCr color system is used, since the texture information of the chrominance components show visible differences between real and fake faces. They used LBP to validate their proposal, among other texture descriptors. We propose here to use EBHOG instead of LBP as antispoofing scheme, as shown in Figure 6.

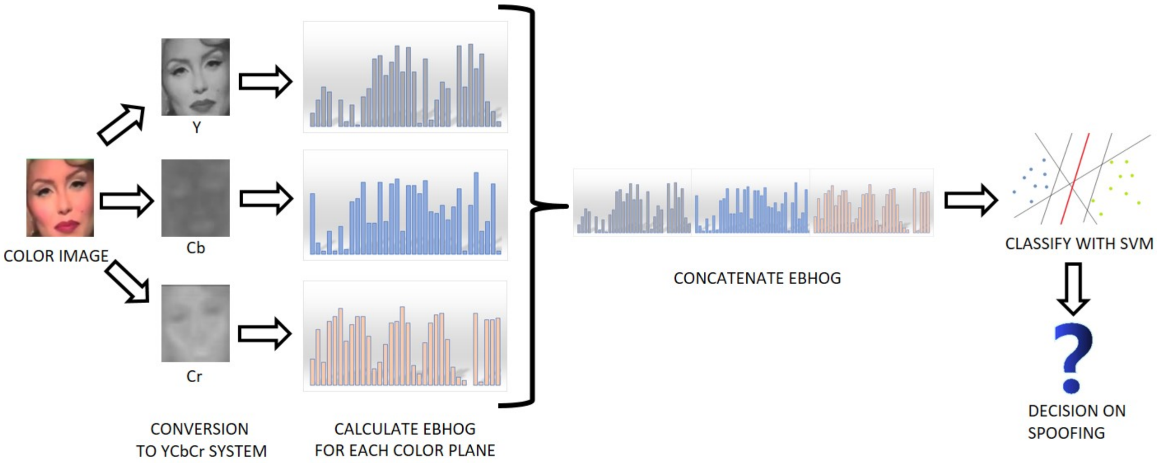


Figure 6. Proposal for face anti spoofing system based on Entropy-Based Histogram of Oriented Gradients (EBHOG).

4. Results

In this section, the datasets used to evaluate our model will be first introduced. Then, the parameters to be used in order to achieve reliable results in face recognition are calculated. After that, our approach will be compared with state-of-the-art methods on the selected databases. Finally, the results of some experiments completed to verify the suitability of our anti spoofing model are shown.

The tests for face recognition were performed using three face databases:

- The Olivetti Research Laboratory (ORL) database [56], which has 400 grayscale images of 40 persons. The images were taken at different times, with changing illumination conditions and different facial expressions. There are 10 images per person. 3 images were used for the training process in order to estimate the necessary EBHOG parameters. Then 2 images were used for the enrollment and, finally, the remaining 5 were used for the recognition stage.

- The Color FERET database [57]. It contains 11,338 pictures of 994 different individuals.The gallery set *fa* was used for the training process, with a subset of 200 users. Then, the tests were completed using gallery sets *fb*, *fc*, *dup1* and *dup2* of FERET database. Images stored in *fb* have changes in the expression from the images in subset *fa*. Images in *fc* have mainly differences in illumination. Then, *dup1* and *dup2* subsets are challenging, since images were taken on different dates from the ones in subset *fa*.
- The Labeled Faces in the Wild (LFW) dataset [58], composed of color images taken from the Internet. There are more than 13,000 photos of 5749 individuals, but for the recognition process only the users with at least two or more images per person have been considered. This reduces the bank of images to 1680 individuals. For the training process a subset of 200 users is again taken into account.

Figure 7 shows graphical examples of some images in these three databases.

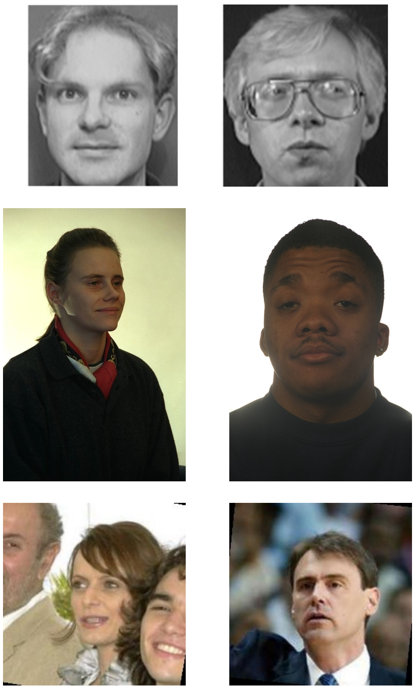


Figure 7. Face examples from the databases used in the face recognition databases: first row, images from the ORL database; second row, images from the Color FERET database; third row, images from the LFW database.

On the other hand, the datasets used for face spoof detection are:

- The CASIA Face Anti-Spoofing Database (FASD) [59], which contains videos of 50 subjects with their corresponding fake faces. There are different image qualities and the face attacks are warped photo, cut photo and digital display device attacks. 20 subjects were used for training the remaining 30 for testing.
- The MIFS (Makeup Induced Face Spoofing) dataset [60], composed of face images of 107 subjects obtained from YouTube video makeup tutorials and face images of associated target subjects from the Internet. There are 4 photos per subject (2 before makeup, 2 after makeup) and 2 photos per target subject, making a total of 642 still images. This database focus specially on impersonating a target person by using makeup.

Figure 8 presents some examples of images in both CASIA FASD and MIFS datasets.

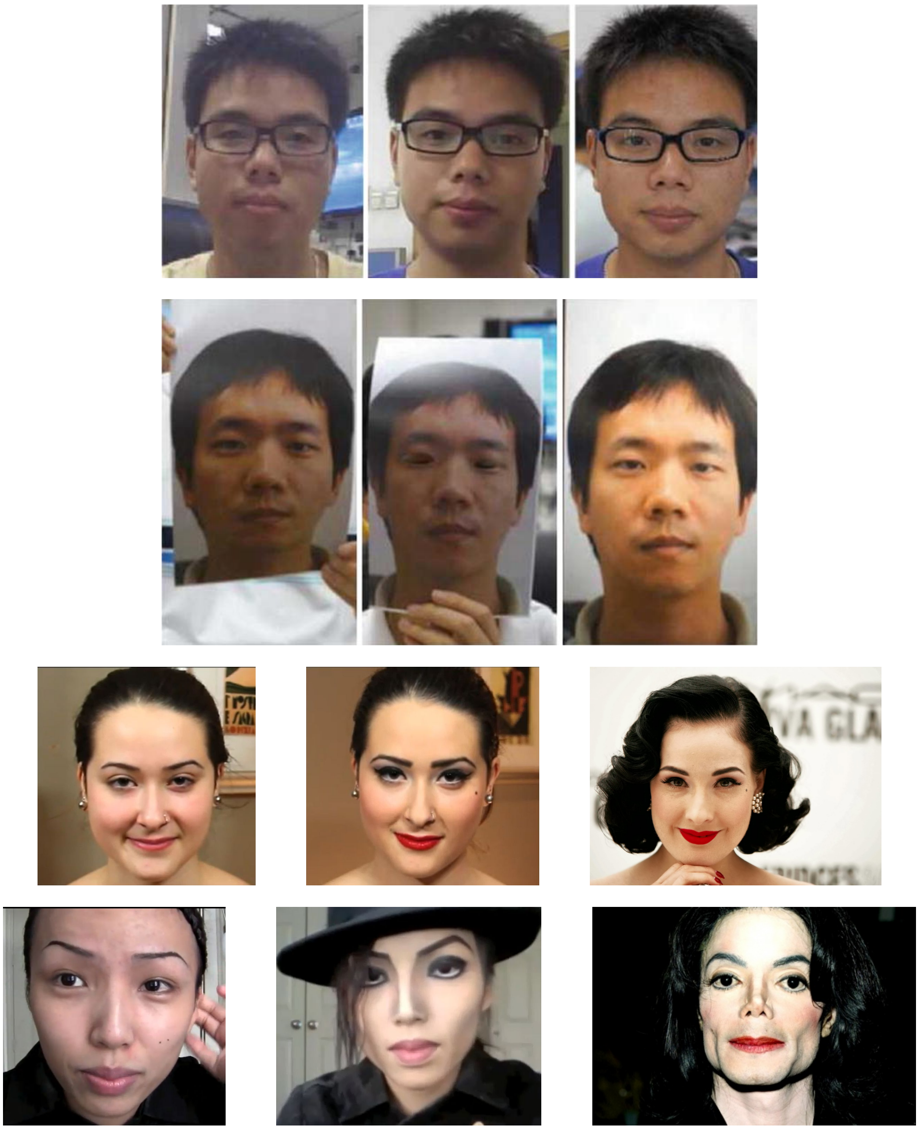


Figure 8. Some examples from the datasets used in the spoof detection: first row, images from the CASIA FASD database with different resolutions; second row, images from the CASIA FASD database with different spoofing attacks; third and fourth rows, images from the MIFS dataset, where the first image in each row is the subject before makeup, the second image is the subject after makeup and the third image is the target subject.

All the experiments have been completed in a computer using MATLAB® in Windows 10, with an Intel® Core i7-7500U processor @2.70 GHz and 8 GB of RAM.

4.1. Proposed System Settings and Recognition Results

As mentioned before (see Section 3.1 and Figure 4), the well-known Viola and Jones detector has been used to detect faces, and then, color information is removed from images. Finally, all the images have been normalized to 130 × 150 pixels.

Then, the parameters for the calculation of the Entropy-Based Histogram of Oriented Gradients (EBHOG) descriptor must be set. In order to choose the best cell size and the number of cells per block for our application, several tests changing the cell size of EBHOG have been performed. These calculations were performed by using 3 images per user in the ORL database. The number of orientation histogram bins (9 bins) has been the same as in [26], as well as the number of overlapping cells between adjacent blocks: half the block size. Finally, a Support Vector Machine (SVM) with a linear kernel has been chosen to classify faces [61], since it is usually suggested to use linear kernels if the number of features is much larger than the number of samples, which would happen in the original data set. The results using Receiver Operating Characteristic (ROC) curves are shown in Figure 9, where CS stands for Cell Size and BS stands for Block Size.

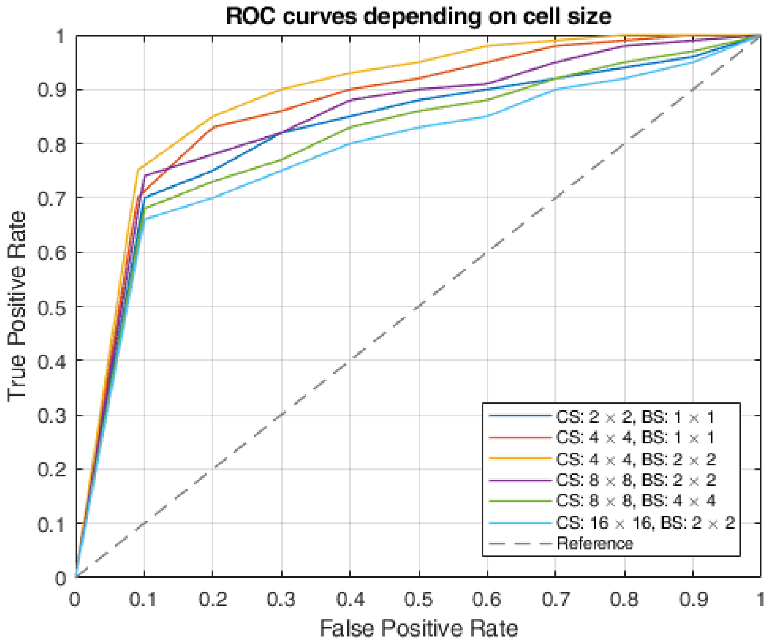


Figure 9. ROC curves depending on EBHOG cell sizes. CS stands for Cell Size, BS stands for Block Size.

To analyze these curves, the Area Under the ROC Curve (AUC) will be computed. The bigger the area, the better the classifier performs. Therefore, the results obtained regarding cell size/block size are presented in Table 1 in terms of area under the ROC curve (AUC).

Table 1. Area Under the ROC Curve (AUCs) of different cell sizes/block sizes for the considered database. The best results are highlighted in bold font.

Cell size (CS)/Block size (BS)	AUC
CS : 2 × 2, BS : 1 × 1	0.8220
CS : 4 × 4, BS : 1 × 1	0.8671
CS : 4 × 4, BS : 2 × 2	**0.8892**
CS : 8 × 8, BS : 2 × 2	0.8450
CS : 8 × 8, BS : 4 × 4	0.8090
CS : 16 × 16, BS : 2 × 2	0.7860

From these results, it becomes clear that the biggest area in Figure 9 is the one obtained with a EBHOG cell size of 4 × 4 and a block size of 2 × 2, so these are the parameters chosen to compute the histograms. To sum up, the parameters to obtain the Entropy-Based Histogram of Oriented Gradients are:

- Size of EBHOG cell: 4 × 4 pixels.
- Number of cells in block: 2 × 2 cells.
- Number of overlapping cells between adjacent blocks: 1 × 1 cells. It is calculated as half the block size.
- Number of orientation histogram bins: 9 bins.

All the experiments considered the same number of samples of a true null hypothesis and a false null hypothesis. With this, the False Rejection Rate (FRR) is defined as the probability that the system will not recognize the identity of an already enrolled person, and the False Acceptance Rate (FAR) is the probability that the system will not reject an impostor (person who is not in the database). False Rejection and False Acceptance errors happen when genuine users are denied access while impostors are accepted to the system, respectively. The experiments completed with the testing set in the ORL database give a False Rejection Rate (Number of False Rejections/Number of Enrollee Recognition Attempts) of 7% and a False Acceptance Rate (Number of False Acceptances/Number of Impostor Recognition Attempts) of just 2%.

Finally, the True Positive Rate (TPR) describes the performance of our system, since it calculates the ratio between the number of True Positives and the number of correct identification cases. In our case, with the parameters considered before we obtained a TPR of 94.4% for the ORL database. Table 2 shows the recognition rate results for the three databases considered in the experiments.

Table 2. Recognition results for the considered databases.

	ORL	FERET				LFW
		fb	*fc*	*dup1*	*dup2*	
TPR (%)	94.4	97.5	89.2	84.5	88.4	78.2

Note that the LFW database is commonly used for benchmarking face verification. However, in this work we consider the closed set identification protocol defined in [62–64]. The TPR is measured by the rank-1 identification accuracy, i.e., by a correct identification.

The mean computational time to extract EBHOG with the parameters set before was 12 ms for one subject on average. When using standard HOG this time was 8 ms on average.

4.2. Comparison with Other Methods and Discussion

Let us compare now the performance of our EBHOG descriptor and some other state-of-the-art descriptors for face recognition. In particular, the original Histogram of Oriented Gradients (HOG) approach has been taken into account [26], as well as other texture descriptors, such as Local Binary Patterns (LBP) [30], Patterns of Oriented Edge Magnitudes (POEM) [65], Scale-Invariant Feature Transform (SIFT) [51], Local Directional Patterns (LDP) [66], Weber Local Descriptors (WLD) [67] and recent Local Diagonal Extrema Number Patterns (LDENP) [68].

On the other hand, given their popularity and accuracy in recognition tasks, CNNs and Deep Learning represent a very successful model and they are used in many applications, particularly in computer vision tasks. One of the strategies that can be followed to apply Deep Learning is transfer learning. It consists of taking a pre-trained network and using it as a starting point to learn a new task. The advantage of this approach is that the pre-established network has already learned a broad set of features that can be applied for similar purposes. To do this, AlexNet has been selected [69]. The AlexNet architecture has eight layers with their respective learning parameters, five of which are convolutional layers and the remaining are fully connected. AlexNet was originally designed to support 1000 classes. However, in our classification problem the number of classes will be equal to the number of different users in each considered database. Thus, AlexNet was adapted to a smaller number of outputs, this being possible due to the flexibility to modify the last layer of the network.

The results the comparison between all these methods can be found in Table 3.

From these results, it becomes clear that adding entropy information to the original Histogram of Oriented Gradients descriptor improves the recognition rate significantly, with an increase over 40% in some of the databases considered for the experiments.

When working with ORL database and fb and fc sets from the FERET database, our proposal does not get the best results, although the performance is rather similar to other state-of-the-art descriptors (our recognition rate is less than 2% lower than the best method in Table 3). In particular, both the Local Diagonal Extrema Number Patterns (LDENP) method and using AlexNet with Transfer Learning achieve the highest recognition rates for these datasets, which are characterized by having different light conditions (ORL and subset fc in FERET) and different expressions (ORL and subset fb in FERET).

Table 3. Recognition rate results (%) of our method in comparison with state-of-the-art algorithms. The best results for each database are highlighted in bold font.

Method	**ORL**	**FERET**				**LFW**
		fb	***fc***	***dup1***	***dup2***	
LBP [30]	87.8	81.0	84.7	64.9	48.6	44.9
POEM [65]	94.4	97.6	86.0	77.8	76.5	74.0
SIFT [51]	92.7	95.9	66.1	65.2	55.4	69.8
LDP [66]	88.5	97.0	82.0	72.0	69.0	71.3
WLD [67]	90.0	93.0	51.0	61.0	50.0	38.0
LDENP [68]	94.6	97.7	**89.9**	82.9	86.8	58.5
AlexNet [69]	**95.5**	**99.7**	88.7	84.3	87.8	75.4
HOG [26]	93.5	90.0	74.0	54.0	46.6	48.2
EBHOG	94.4	97.5	89.2	**84.5**	**88.4**	**78.2**

On the other hand, the EBHOG descriptor has the better recognition rates for both datasets $dup1$ and $dup2$ from the FERET database and for the challenging LFW database. These datasets are characterized by including photos with temporal/age changes (subsets $dup1$ and $dup2$ in FERET) and large variations in pose, expression and illuminations (LFW). Our method achieves high recognition results for these difficult datasets, which shows again that entropy plays a major role to achieve reliable recognition rates in difficult, demanding situations.

4.3. Experiments on Detecting Spoofing and Discussion

Let us show now the results of the completed tests in order to detect face spoofing attacks. Experiments on the CASIA FASD database are strictly done with the original protocol defined by the authors. Thus, 30 face images in each of the training videos are selected randomly. Then, 30 face images from each of the testing videos are also selected randomly. The video is then classified as 'real' or 'fake' by averaging the 30 images scores. In order to compare the classification results, the Equal Error Rate (EER) is used, as suggested by the authors. The results are shown in Table 4, where a comparison with the results from [9] and from using HOG instead of EBHOG are displayed.

Table 4. Performance using Equal Error Rate (EER) (%) in the CASIA FASD database and YCbCr color system. The best results are highlighted in bold font.

Method	**EER**
LBP [9]	12.4
HOG [26]	21.6
EBHOG	**9.5**

As it can be seen, the proposed method improves on the original developed in [9], and it can be considered as an alternative to face spoofing detection.

In [60] authors defined two spoofing indexes. SI_1 evaluates the similarity between the after-makeup images of subject p, A_i^p, and the target images, T_j^p, for $i, j \in \{1,2\}$, with respect to two samples of the target identity, T_1^p, T_2^p. SI_2 computes the similarity between the after-makeup images A_i^p and the target images T_j^p with respect to two samples of the after-makeup images A_1^p, A_2^p:

$$SI_1 = 1 - \min_{i,j} \left| \phi\left(A_i^p, T_j^p\right) - \phi\left(T_1^p, T_2^p\right) \right| \tag{14}$$

$$SI_2 = 1 - \min_{i,j} \left| \phi\left(A_i^p, T_j^p\right) - \phi\left(A_1^p, A_2^p\right) \right| \tag{15}$$

Here $\phi(x,y)$ is the similarity match score between images x and y. Since our descriptor is based on histogram calculation, we propose to calculate the similarity between two images x and y by using the histogram intersection kernel:

$$\phi(x,y) = \sum_{i=1}^{N} \min\left(EBHOG_i^{Norm}(x), EBHOG_i^{Norm}(y)\right) \quad (16)$$

$\phi(x,y)$ is finally normalized in the $[0,1]$ interval.

Input images are then again classified as 'real' or 'fake' and, as in the CASIA FASD database, the Equal Error Rate (EER) is calculated to compare the results. The results are shown in Table 5, where a comparison with the same methods as in Table 4 are considered. Notice that the threshold in the similarity score to consider that an image is genuine or not has been introduced, as well.

Table 5. Performance using EER (%) in the MIFS database and YCbCr color system. The best results for each database are highlighted in bold font.

Method	**Similarity Threshold**				
	0.75	**0.8**	**0.85**	**0.9**	**0.95**
LBP [9]	15.5	7.6	3.4	4.5	9.8
HOG [26]	22.4	16.5	9.5	12.6	18.3
EBHOG	13.8	6.8	**3.1**	3.8	7.5

The threshold for the histogram intersection similarity score that achieves better results from Table 5 is 0.85. Again, the best performance corresponds to the method proposed. As a conclusion, introducing entropy improves the results in face spoofing detection compared with similar approaches.

To sum up, the results from the experiments show that our proposal is consistently among the best local descriptors for face recognition, outperforming most of the recent approaches results in Tables 3–5. The great amount of tests implemented on several face databases have effectively shown the potential of the EBHOG approach.

5. Conclusions

In the last few years, with the increasing popularity of mobile technologies, almost all mobile phone applications have access to private data in some way. This fact is particularly vulnerable in a smart city context. Cyberattacks on social networks have become common to get profiles and hackers often use them to steal personal data or even to discredit their real user. One way to prevent spoofing is by authenticating users using biometric traits such as fingerprints, iris or facial features.

In this work, a new face recognition and spoofing detection approach using an entropy-based HOG descriptor has been presented. The results show that our method provides a reliable descriptor for different databases and, as a result, we consider that our proposal may be applied to detect possible face spoofing attacks using pictures uploaded to social media. Future works aim at applying the proposed algorithm to real situations in social networks. We are currently adapting the method to work with GPUs and parallelizing the most time-consuming steps in the algorithm.

Author Contributions: Conceptualization, F.A.P. and M.J.P.; methodology, C.R.-M.; validation, C.R.-M. and M.P.; formal analysis, M.J.P.; writing—review and editing, F.A.P. and M.P. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Ministerio de Economía y Competitividad (Spain), project TIN2013-40982-R and by the Programa Redes-I^{3}CE de Investigación en Docencia Universitaria 2016–2017, Red ICE 3701, of the Instituto de Ciencias de la Educación of the University of Alicante.

Acknowledgments: This work was partially supported by the Ministerio de Economía y Competitividad (Spain), project TIN2013-40982-R, the FEDER funds, and the "Red de Investigación en el uso del aprendizaje colaborativo para la adquisición de competencias básicas. El caso Erasmus+ EUROBOTIQUE", Red ICE 3701 curso 2016–2017.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Jain, A.K.; Flynn, P.; Ross, A.A. *Handbook of Biometrics*; Springer: Boston, MA, USA, 2008. doi:10.1007/978-0-387-71041-9.
2. Girardin, G. Consumers rule: Why the biometrics market is facing major disruption. *Biom. Technol. Today* **2017**, *2017*, 10–11. doi:10.1016/S0969-4765(17)30116-9.
3. Tractica. Global Biometrics Market Revenue to Reach $15.1 Billion by 2025 | Tractica. Available online: https://www.tractica.com/newsroom/press-releases/global-biometrics-market-revenue-to-reach-15-1-billion-by-2025/ (accessed on 8 January 2019).
4. Mobile biometrics revenues predicted to boom. *Biom. Technol. Today* **2017**, *2017*, 3–12. doi:https://doi.org/10.1016/S0969-4765(17)30161-3.
5. Nita, S.L.; Mihailescu, M.I.; Pau, V.C. Security and Cryptographic Challenges for Authentication Based on Biometrics Data. *Cryptography* **2018**, *2*, 39. doi:10.3390/cryptography2040039.
6. Chugh, T.; Cao, K.; Jain, A.K. Fingerprint Spoof Buster: Use of Minutiae-Centered Patches. *IEEE Trans. Inform. Forensics Secur.* **2018**, *13*, 2190–2202. doi:10.1109/TIFS.2018.2812193.
7. Shaju, S.; Davis, D. Haar wavelet transform based histogram concatenation model for finger print spoofing detection. In Proceedings of the IEEE 2017 International Conference on Communication and Signal Processing (ICCSP), Chennai, India, 6–8 April 2017; pp. 1352–1356.
8. Fernandez, A.; Carus, J.L.; Usamentiaga, R.; Casado, R. Face Recognition and Spoofing Detection System Adapted To Visually-Impaired People. *IEEE Lat. Am. Trans.* **2016**, *14*, 913–921.
9. Boulkenafet, Z.; Komulainen, J.; Hadid, A. Face spoofing detection using colour texture analysis. *IEEE Trans. Inform. Forensics Secur.* **2016**, *11*, 1818–1830.
10. Li, H.; Li, W.; Cao, H.; Wang, S.; Huang, F.; Kot, A.C. Unsupervised domain adaptation for face anti-spoofing. *IEEE Trans. Inform. Forensics Secur.* **2018**, *13*, 1794–1809.
11. Farmanbar, M.; Toygar, Ö. Spoof detection on face and palmprint biometrics. *Signal Image Video Process.* **2017**, *11*, 1253–1260.
12. Hsieh, S.H.; Li, Y.H.; Wang, W.; Tien, C.H. A Novel Anti-Spoofing Solution for Iris Recognition Toward Cosmetic Contact Lens Attack Using Spectral ICA Analysis. *Sensors* **2018**, *18*, 795.
13. Yu, S.; Ai, Y.; Xu, B.; Zhou, Y.; Li, W.; Liao, Q.; Poh, N. Two strategies to optimize the decisions in signature verification with the presence of spoofing attacks. *Inform. Sci.* **2016**, *352*, 188–202.
14. Dawson, M.; Omar, M.; Abramson, J. Understanding the methods behind cyber terrorism. In *Encyclopedia of Information Science and Technology*, 3rd ed.; IGI Global: Hershey PA, USA, 2015; pp. 1539–1549.
15. Patel, K.; Han, H.; Jain, A.K. Secure face unlock: Spoof detection on smartphones. *IEEE Trans. Inform. Forensics Secur.* **2016**, *11*, 2268–2283.
16. Kamble, A.; Malemath, V.S.; Patil, D. Security attacks and secure routing protocols in RPL-based Internet of Things: Survey. In Proceedings of the IEEE 2017 International Conference on Emerging Trends & Innovation in ICT (ICEI), Pune, India, 3–5 February 2017; pp. 33–39.
17. Liu, F.; Zhu, X.; Hu, Y.; Ren, L.; Johnson, H. A cloud theory-based trust computing model in social networks. *Entropy* **2016**, *19*, 11.
18. Rathore, S.; Sharma, P.K.; Loia, V.; Jeong, Y.S.; Park, J.H. Social network security: Issues, challenges, threats, and solutions. *Inform. Sci.* **2017**, *421*, 43–69.
19. van Schaik, P.; Jansen, J.; Onibokun, J.; Camp, J.; Kusev, P. Security and privacy in online social networking: Risk perceptions and precautionary behaviour. *Comput. Hum. Behav.* **2018**, *78*, 283–297.
20. Karczmarek, P.; Pedrycz, W.; Kiersztyn, A.; Rutka, P. A study in facial features saliency in face recognition: An analytic hierarchy process approach. *Soft Comput.* **2017**, *21*, 7503–7517.
21. Lu, Y.; Wang, S.; Zhao, W. Facial Expression Recognition Based on Discrete Separable Shearlet Transform and Feature Selection. *Algorithms* **2019**, *12*, 11.
22. Maatta, J.; Hadid, A.; Pietikainen, M. Face spoofing detection from single images using texture and local shape analysis. *IET Biom.* **2012**, *1*, 3–10. doi:10.1049/iet-bmt.2011.0009.
23. Chakka, M.M.; Anjos, A.; Marcel, S.; Tronci, R.; Muntoni, D.; Fadda, G. Competition on counter measures to 2-D facial spoofing attacks. In Proceedings of the International Joint Conference on Biometrics (IJCB 2011), Washington, DC, USA, 11–13 December 2011.

24. Lowe, D.G. Object recognition from local scale-invariant features. In Proceedings of the Seventh IEEE International Conference on Computer Vision, Kerkyra, Greece, 20–27 September 1999; Volume 2, pp. 1150–1157.
25. Luo, J.; Ma, Y.; Takikawa, E.; Lao, S.; Kawade, M.; Lu, B.L. Person-specific SIFT features for face recognition. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2007), Honolulu, HI, USA, 15–20 April 2007; Volume 2, p. 593.
26. Dalal, N.; Triggs, B. Histograms of oriented gradients for human detection. In Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), San Diego, CA, USA, 20–25 June 2005; pp. 886–893.
27. Déniz, O.; Bueno, G.; Salido, J.; De la Torre, F. Face recognition using histograms of oriented gradients. *Pattern Recognit. Lett.* **2011**, *32*, 1598–1603.
28. Bay, H.; Tuytelaars, T.; Van Gool, L. Surf: Speeded up robust features. In *European Conference on Computer Vision*; Springer: Berlin/Heidelberg, Germany, 2006; pp. 404–417.
29. Boulkenafet, Z.; Komulainen, J.; Hadid, A. Face antispoofing using speeded-up robust features and fisher vector encoding. *IEEE Signal Process. Lett.* **2017**, *24*, 141–145.
30. Ahonen, T.; Hadid, A.; Pietikäinen, M. Face Recognition with Local Binary Patterns. In *Computer Vision—ECCV 2004*; Pajdla, T., Matas, J., Eds.; Springer: Berlin/Heidelberg, Germany, 2004; pp. 469–481.
31. Ma, Z.; Ding, Y.; Li, B.; Yuan, X. Deep CNNs with Robust LBP Guiding Pooling for Face Recognition. *Sensors* **2018**, *18*, 3876.
32. Yang, J.; Lei, Z.; Li, S.Z. Learn Convolutional Neural Network for Face Anti-Spoofing. *arXiv* **2014**, arXiv:1408.5601.
33. Rehman, Y.A.U.; Po, L.M.; Liu, M. LiveNet: Improving features generalization for face liveness detection using convolution neural networks. *Expert Syst. Appl.* **2018**, *108*, 159–169. doi:10.1016/j.eswa.2018.05.004.
34. Patel, K.; Han, H.; Jain, A.K. Cross-Database Face Antispoofing with Robust Feature Representation. In *Biometric Recognition*; You, Z., Zhou, J., Wang, Y., Sun, Z., Shan, S., Zheng, W., Feng, J., Zhao, Q., Eds.; Springer International Publishing: Cham, Switzerland, 2016; Volume 9967, pp. 611–619. doi:10.1007/978-3-319-46654-5_67.
35. Alotaibi, A.; Mahmood, A. Deep face liveness detection based on nonlinear diffusion using convolution neural network. *Signal Image Video Process.* **2017**, *11*, 713–720. doi:10.1007/s11760-016-1014-2.
36. Li, L.; Feng, X.; Xia, Z.; Jiang, X.; Hadid, A. Face spoofing detection with local binary pattern network. *J. Vis. Commun. Image Represent.* **2018**, *54*, 182–192. doi:10.1016/j.jvcir.2018.05.009.
37. Livni, R.; Shalev-Shwartz, S.; Shamir, O. On the Computational Efficiency of Training Neural Networks. In *Advances in Neural Information Processing Systems 27*; Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N.D., Weinberger, K.Q., Eds.; Curran Associates Inc.: Dutchess County, NY, USA, 2014; pp. 855–863. Available online: http://papers.nips.cc/paper/5267-on-the-computational-efficiency-of-training-neural-networks (accessed on 10 December 2019).
38. Miralles-Pechuán, L.; Rosso, D.; Jiménez, F.; García, J.M. A methodology based on Deep Learning for advert value calculation in CPM, CPC and CPA networks. *Soft Comput.* **2017**, *21*, 651–665.
39. Mahmud, M.; Kaiser, M.S.; Hussain, A.; Vassanelli, S. Applications of deep learning and reinforcement learning to biological data. *IEEE Trans. Neural Netw. Learn. Syst.* **2018**, *29*, 2063–2079.
40. Yang, H.; Huang, Y.; Yu, Y.; Yao, M.; Zhang, X. Privacy-Preserving Extraction of HOG Features Based on Integer Vector Homomorphic Encryption. In Proceedings of the International Conference on Information Security Practice and Experience, Melbourne, Australia, 13–15 December 2017; Springer: Berlin/Heidelberg, Germany, 2017; pp. 102–117. Available online: https://www.semanticscholar.org/paper/Privacy-Preserving-Extraction-of-HOG-Features-Based-Yang-Huang/e286f4fb60fd819dd36db44d0f56dc76932aaee4 (accessed on 10 December 2019).
41. Bozkir, A.S.; Sezer, E.A. Use of HOG descriptors in phishing detection. In Proceedings of the 2016 IEEE 4th International Symposium on Digital Forensic and Security (ISDFS), Little Rock, AR, USA, 25–27 April 2016; pp. 148–153.
42. Chandra, D.K.; Chowgule, W.; Fu, Y.; Lin, D. RIPA: Real-Time Image Privacy Alert System. In Proceedings of the 2018 IEEE 4th International Conference on Collaboration and Internet Computing (CIC), Philadelphia, PA, USA, 18–20 October 2018; pp. 136–145.

43. Lu, W.S. Handwritten digits recognition using PCA of histogram of oriented gradient. In Proceedings of the 2017 IEEE Pacific Rim Conference on Communications, Computers and Signal Processing (PACRIM), Victoria, BC, Canada, 21–23 August 2017; pp. 1–5.
44. Alizadeh, S.; Fazel, A. Convolutional neural networks for facial expression recognition. *arXiv* **2017**, arXiv:1704.06756.
45. Schwartz, W.R.; Rocha, A.; Pedrini, H. Face spoofing detection through partial least squares and low-level descriptors. In Proceedings of the 2011 IEEE International Joint Conference on Biometrics (IJCB), Washington, DC, USA, 11–13 October 2011; pp. 1–8.
46. Komulainen, J.; Hadid, A.; Pietikainen, M. Context based face anti-spoofing. In Proceedings of the 2013 IEEE Sixth International Conference on Biometrics: Theory, Applications and Systems (BTAS), Arlington, VA, USA, 29 September–2 October 2013; pp. 1–8.
47. Kaur, S.; Sharma, R. An Intelligent Approach for Anti-Spoofing in a Multimodal Biometric System. *Int. J. Comput. Sci. Eng.* **2017**, *9*, 522–529.
48. Galdi, C.; Nappi, M.; Dugelay, J.L. Secure User Authentication on Smartphones via Sensor and Face Recognition on Short Video Clips. In Proceedings of the International Conference on Green, Pervasive, and Cloud Computing, Cetara, Italy, 11–14 May 2017; Springer: Berlin/Heidelberg, Germany, 2017, pp. 15–22. Available online: https://www.semanticscholar.org/paper/Secure-User-Authentication-on-Smartphones-via-and-Galdi-Nappi/d7936ad5e71703d3c7b686ff13eacb88f3b7dbf9 (accessed on 10 December 2019).
49. Viola, P.; Jones, M. Robust real-time face detection. *Int. J. Comput. Vis.* **2004**, *57*, 137–154. doi:{10.1023/B:VISI.0000013087.49260.fb}.
50. Wang, Y.Q. An Analysis of the Viola-Jones Face Detection Algorithm. *Image Process. Line* **2014**, *4*, 128–148. doi:10.5201/ipol.2014.104.
51. Lowe, D. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vis.* **2004**, *60*, 91–110. doi:10.1023/B:VISI.0000029664.99615.94.
52. Karmakar, D.; Murthy, C.A. Face Recognition using Face-Autocropping and Facial Feature Points Extraction. In Proceedings of the 2nd International Conference on Perception and Machine Intelligence (PerMIn '15), Kolkata, West Bengal, India, 26–27 February 2015; ACM Press: New York, NY, USA, 2015; pp. 116–122. doi:10.1145/2708463.2709056.
53. Leung, H.Y.; Cheng, L.M.; Li, X.Y. A FPGA implementation of facial feature extraction. *J. Real-Time Image Process.* **2015**, *10*, 135–149. doi:10.1007/s11554-012-0263-8.
54. Portes de Albuquerque, M.; Esquef, I.A.; Gesualdi Mello, A.R.; Portes de Albuquerque, M. Image thresholding using Tsallis entropy. *Pattern Recognit. Lett.* **2004**, *25*, 1059–1065. doi:10.1016/j.patrec.2004.03.003.
55. Pan, G.; Wu, Z.; Sun, L. Liveness detection for face recognition. In *Recent Advances in Face Recognition*; IntechOpen: London, UK, 2008.
56. Samaria, F.S.; Harter, A.C. Parameterisation of a stochastic model for human face identification. In Proceedings of the Second IEEE Workshop on Applications of Computer Vision, Sarasota, FL, USA, 5–7 December 1994; pp. 138–142.
57. Phillips, P.J.; Moon, H.; Rizvi, S.A.; Rauss, P.J. The FERET evaluation methodology for face recognition algorithms. *IEEE Trans. Pattern Anal. Mach. Intell.* **2000**, *22*, 1090–1104.
58. Huang, G.B.; Ramesh, M.; Berg, T.; Learned-Miller, E. *Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments*; Technical Report 07-49; University of Massachusetts: Amherst, MA, USA, 2007.
59. Zhang, Z.; Yan, J.; Liu, S.; Lei, Z.; Yi, D.; Li, S.Z. A face antispoofing database with diverse attacks. In Proceedings of the 2012 5th IAPR International Conference on Biometrics (ICB), New Delhi, India, 29 March–1 April 2012; pp. 26–31.
60. Chen, C.; Dantcheva, A.; Swearingen, T.; Ross, A. Spoofing faces using makeup: An investigative study. In Proceedings of the 2017 IEEE International Conference on Identity, Security and Behavior Analysis (ISBA), New Delhi, India, 22–24 February 2017; pp. 1–8.
61. Chang, C.C.; Lin, C.J. LIBSVM: a library for support vector machines. *ACM Trans. Intell. Syst. Technol.* **2011**, *2*, 27.
62. Best-Rowden, L.; Han, H.; Otto, C.; Klare, B.F.; Jain, A.K. Unconstrained face recognition: Identifying a person of interest from a media collection. *IEEE Trans. Inform. Forensics Secur.* **2014**, *9*, 2144–2157.

63. Taigman, Y.; Yang, M.; Ranzato, M.; Wolf, L. Web-scale training for face identification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2015), Boston, MA, USA, 7–12 June 2015; pp. 2746–2754.
64. Pujol, F.A.; Mora, H.; Girona-Selva, J.A. A connectionist computational method for face recognition. *Int. J. Appl. Math. Comput. Sci.* **2016**, *26*, 451–465.
65. Vu, N.S.; Caplier, A. Face recognition with patterns of oriented edge magnitudes. In *European Conference on Computer Vision*; Springer: Berlin/Heidelberg, Germany, 2010; pp. 313–326.
66. Jabid, T.; Kabir, M.H.; Chae, O. Local directional pattern (LDP) for face recognition. In Proceedings of the 2010 Digest of Technical Papers International Conference on Consumer Electronics (ICCE), Las Vegas, NV, USA, 9–13 January 2010; pp. 329–330.
67. Chen, J.; Shan, S.; He, C.; Zhao, G.; Pietikainen, M.; Chen, X.; Gao, W. WLD: A robust local image descriptor. *IEEE Trans. Pattern Anal. Mach. Intell.* **2010**, *32*, 1705–1720.
68. Pillai, A.; Soundrapandiyan, R.; Satapathy, S.; Satapathy, S.C.; Jung, K.H.; Krishnan, R. Local diagonal extrema number pattern: A new feature descriptor for face recognition. *Future Gener. Comput. Syst.* **2018**, *81*, 297–306. doi:10.1016/j.future.2017.09.055.
69. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet Classification with Deep Convolutional Neural Networks. In Proceedings of the 25th International Conference on Neural Information Processing Systems (NIPS'12), Lake Tahoe, NV, USA, 3–6 December 2012; Curran Associates Inc.: Red Hook, NY, USA, 2012; Volume 1, pp. 1097–1105.