Counter-Measures to Photo Attacks in Face Recognition: a public database and a baseline

André Anjos and Sébastien Marcel
Idiap Research Institute
Centre du Parc - rue Marconi 19
CH-1920 Martigny, Suisse
{andre.anjos,sebastien.marcel}@idiap.ch

Abstract

A common technique to by-pass 2-D face recognition systems is to use photographs of spoofed identities. Unfortunately, research in counter-measures to this type of attack have not kept-up - even if such threats have been known for nearly a decade, there seems to exist no consensus on best practices, techniques or protocols for developing and testing spoofing-detectors for face recognition. We attribute the reason for this delay, partly, to the unavailability of public databases and protocols to study solutions and compare results. To this purpose we introduce the publicly available PRINT-ATTACK database and exemplify how to use its companion protocol with a motion-based algorithm that detects correlations between the person's head movements and the scene context. The results are to be used as basis for comparison to other counter-measure techniques. The PRINT-ATTACK database contains 200 videos of real-accesses and 200 videos of spoof attempts using printed photographs of 50 different identities.

1. Introduction

Identity theft is a concern that prevents the mainstream adoption of biometrics as *de facto* form of identification in commercial systems [1]. Contrary to password-protected systems, our biometric information is widely available and extremely easy to sample. It suffices a small search on the internet to unveil prelabelled samples from users at specialized websites such as Flickr or Facebook. Images can also be easily captured at distance without previous consent. Users cannot trust that these samples will not be dishonestly used to assume their identity before biometric recognition systems.

In this work we are particularly concerned with direct [2] print-attacks to unimodal 2-D (visual spectra) face-recognition systems[1]. These so-called *spoofing attempts* [3] are direct attacks to the input sensors of the biometric system. Attackers in this case are assumed not to have access to the internals of the recognition system and manage to penetrate by only displaying printed photos of the attacked identity to the input camera. This type of attack is therefore very easy to reproduce and has great potential to succeed [4].

Despite the fact that solutions exist for spoof prevention using multi-modal techniques [5, 6, 7, 8], it is our belief that research for counter-measures solely based on unimodal 2-D imagery has not yet reached a matured state. There seems to exist no consensus on best practices and techniques to be deployed on attack detection using non-intrusive methods. The number of publications on the subject is small. A missing key to this puzzle is the lack of standard databases to test counter-measures, followed by a set of protocols to evaluate performance and allow for objective comparison.

This work introduces a publicly available database, protocols and a baseline technique to evaluate counter-measures to spoofing attacks in face recognition systems. The remaining of this text is organized as follows: Section 2 discusses the current state-of-the-art in anti-spoofing for 2-D face recognition systems. Section 3 describes the PRINT-ATTACK anti-spoofing database and defines protocols for its usage. Section 4 defines a baseline technique that can be used for comparison with other algorithms. Section 5 reports on the experimental setup and results. Finally, Section 6 concludes and discusses possible extensions of this work.

[1] We will refer to such systems simply as face recognition systems from this point onwards.

2. Literature Survey

Face recognition systems are known to respond weakly to attacks for a long time [9, 4] and are easily spoofed using a simple photograph of the enrolled person's face, which may be displayed in hard-copy or on a screen. In this short survey, we focus on methods that present counter-measures to such kind of attacks.

Anti-spoofing for 2-D face recognition systems can be coarsely classified in 3 categories with respect to the clues used for attack detection: motion, texture analysis and liveness detection. In *motion analysis* one is interested in detecting clues generated when two-dimensional counterfeits are presented to the system input camera, for example photos or video clips. Planar objects will move significantly differently from real human faces which are 3-D objects, in many cases and such deformation patterns can be used for spoof detection. For example, [10] explores the Lambertian reflectance model to derive differences between the 2-D images of the face presented during an attack and a real (3-D) face, in real-access attempts. It does so by deriving an equation that estimates the latent reflectance information that exists on images captured in both scenarios using either a variational retinex-based method or a far simpler difference-of-gaussians [11] based approach similar to [12]. This is the first work on literature to propose a publicly available database specifically tailored towards the development of spoofing counter-measures. [13] present a technique to evaluate liveness based on a short sequence of images using a binary detector that evaluates the trajectories of selected parts of the face presented to the input sensor using a simplified optical flow analysis followed by an heuristic classifier. The same authors introduce in [14] a method for fusing scores from different experts systems that observe, concurrently, the 3-D face motion scheme introduced on the previous work and liveness properties such as eye-blinks or mouth movements. [15] the authors propose a method to detect attacks produced with planar media (such as paper or screens) using motion estimation by optical flow.

Texture analysis counter-measures take advantage of texture patterns that may look unnatural when exploring the input image data. Examples of detectable texture patterns are printing failures or overall image blur. [12] describes a method for print-attack detection by exploiting differences in the 2-D Fourier spectra comparing the hard-copies of client faces and real-accesses. The method will work well for down-sampled photos of the attacked identity, but is likely to fail for higher-quality samples. In [16] the author proposes a method to detect spoofing attacks using printed photos by analyzing the micro-textures present on the paper using a linear SVM classifier [17]. One limitation of this method is that the input image needs to be reasonably sharp.

Liveness detection tries to capture signs of life from the user images by analysing spontaneous movements that cannot be detected in photographs, such as eye-blinks. [18] and [19] bring a real-time liveness detection specifically against photo-spoofing using (spontaneous) eye-blinks which are supposed to occur once every 2-4 seconds in humans. The system developed uses an undirected conditional random field framework to model the eye-blinking that relaxes the independence assumption of generative modelling and state dependence limitations from hidden Markov modelling. A later work by the same authors [20] augment the number of counter-measures deployed to include a scene context matching that helps preventing video-spoofing in stationary face-recognition systems.

3. The PRINT-ATTACK Database

The PRINT-ATTACK biometric (face) database[2] consists of short video recordings of both real-access and attack attempts to 50 different identities. To create the dataset each person recorded a number of videos at 2 different stationary conditions:

- **controlled**: In this case the background of the scene is uniform and the light of a fluorescent lamp illuminates the scene;
- **adverse**: In this case the background of the scene is non-uniform and day-light illuminates the scene.

Under these two different conditions, people were asked to sit down in front of a custom acquisition system built on an Apple 13-inch MacBook laptop and capture two video sequences with a resolution of 320 by 240 pixels (QVGA), at 25 frames-per-second and of 15 seconds each (375 frames). Videos were recorded using Apple's Quicktime format (MOV files).

The laptop is positioned on the top of a short support (∼15 cm) so that faces are captured as they look up-front. The acquisition operator launches the capturing program and asks the person to look into the laptop camera as they would normally do waiting for a recognition system to do its task. The program shows a reproduction of the current image being captured and, overlaid, the output of a face-detector used to guide the person during the session. In this particular setup, faces are detected using a cascade of classifiers based on a variant of Local Binary Patterns (LBP) [21] referred as Modified Census Transform (MCT) [22]. The face-detector helps the user self-adjusting the distance from

[2] `http://www.idiap.ch/dataset/printattack`

Figure 1. Example hard-copies of client high-resolution pictures.

the laptop camera and making sure that a face can be detected at all times during the acquisition. After acquisition was finished, the operator would still verify the videos did not contain problems by visual inspection and proceed to acquire the next video.

3.1. Collecting samples and generating the attacks

Under the same illumination and background settings used for real-access video clips, the acquisition operator took two high-resolution pictures of each person using a 12.1 megapixel Canon PowerShot SX150 IS camera that would be used as basis for the spoofing attempts. People were asked to cooperate in with this part of the acquisition so as to maximize the chances of an attack to succeed. They were asked to look up-front such as in the acquisition of the real-access attempts.

To realize the attacks, hard copies of the digital photographs were printed on plain A4 paper using a Triumph-Adler DCC 2520 color laser printer. Figure 1 shows some examples of printed copies. The left column contains samples taken from the *controlled* scenario, while the right column shows samples from the *adverse* scenario.

Using such images, the operator generates the attacks by displaying the printouts of each client to the same acquisition setup used for sampling the real-client accesses. Video clips of about 10 seconds are captured for each spoof attempt, in two different attack modes:

- *hand-based attacks*: in this mode, the operator holds the prints using their own hands;
- *fixed-support attacks*: the operator glues the client prints to the wall so they don't move during the spoof attempt.

The first set of (hand-based) attacks show a *shaking* behavior that can be observed when people hold photographs of spoofed identities in front of cameras and that, sometimes, can trick eye-blinking detectors [19]. It differs from the second set that is completely static and should be easier for liveness-based counter-measures to spoofing.

3.2. Performance Figures

A spoofing detection system is subject to two types of errors, either the real access is rejected (false rejection) or an attack is accepted (false acceptance). In order to measure the performance of a spoofing detection system, we use the Half Total Error Rate (HTER), which combines the False Rejection Rate (FRR) and the False Acceptance Rate (FAR) and is defined as:

$$HTER(\tau, \mathcal{D}) = \frac{FAR(\tau, \mathcal{D}) + FRR(\tau, \mathcal{D})}{2} \quad [\%] \qquad (1)$$

where $\mathcal{D}$ denotes the used dataset. Since both the FAR and the FRR depends on the threshold τ, they are strongly related to each other: increasing the FAR will reduce the FRR and vice-versa. For this reason, results are often presented using either Receiver Operating Characteristic (ROC) or Detection-Error Tradeoff (DET) [23] curves, which basically plots the FAR versus the FRR for different values of the threshold. Another widely used measure to summarise the performance of a system is the Equal Error Rate (EER), defined as the point along the ROC or DET curve where the FAR equals the FRR.

3.3. Protocols

The set of 400 videos (200 real-accesses and 200 attacks) is decomposed into 3 subsets allowing for training, development and testing of binary classifiers. Identities for each subset were chosen randomly but do not overlap, i.e. people that are on one of the subsets do not appear in any other set. This choice guarantees that specific behavior (such as eye-blinking patterns or head-poses) are not picked up by detectors and final systems generalize well.

Moreover, each print-attack subset can be further sub-classified into two groups that split the attacking support used during the acquisition (hand-based or fixed-support). Counter-measures developed using this database should report error figures that consider both separated and aggregated grouping, from which it is possible to understand which types of attacks are better handled by the proposed method. Table 1 summarizes the number of videos taken for both real-access

Type	Train	Devel.	Test	Total
Real-access	60	60	80	200
Print-attack	30+30	30+30	40+40	100+100
Total	120	120	160	400

Table 1. Number of videos in each database subset. Numbers displayed as sums indicate the amount of hand-based and fixed-support attacks available in each subset when relevant.

and print-attack attempts and how they are split in the different subsets and groups.

It is recommended that training and development samples are used to *learn* classifiers how to discriminate. One trivial example is to use the training set for training the classifier itself and the development data to estimate when to stop training. A second possibility, which may generalize less well, is to merge both training and development sets, using the merged set as training data and to formulate a stop criteria. Finally, the test set should be *solely* used to report error rates and performance curves. If a single number is desired, a threshold τ should be chosen at the development set and the HTER reported using the test set data. As means of uniformizing reports, we recommend choosing the threshold τ on the EER at the development set.

We now define a baseline technique that can be used as comparison point for future work developed using this database, exemplifying how error should be reported.

4. The Proposed Counter-Measure

Motion-based algorithms for anti-spoofing typically use complex methods such as Optical Flow estimators to extract deformation patterns from the image being analyzed. Nevertheless, for stationary recognition systems, another far simpler clue can be effectively used to distinguish between real-accesses and attacks: the relative movement intensity between the face and the scene background. In the case of an attack, using a photograph or a video-clip, it should be possible to observe a high-correlation between the total amount of movement in these two regions of interest (RoI).

4.1. Feature Extraction

For this baseline technique we ignore the movement direction and focus on intensity only. The total motion in the RoI is calculated using simple gray-scaled frame-difference and an area-based normalization technique that removes differences in size so different face/background regions remain comparable as

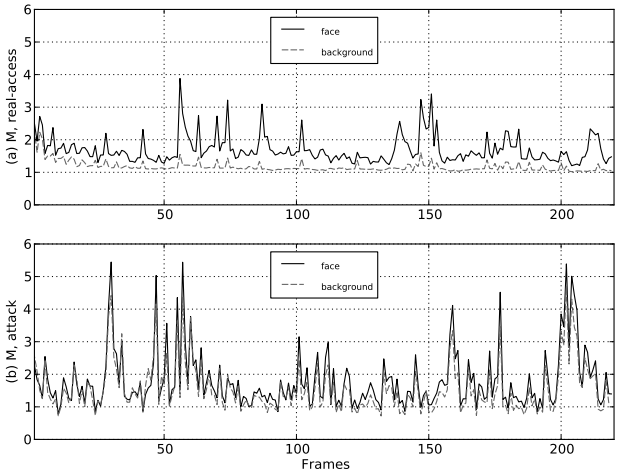

Figure 2. Motion $M_{\mathcal{D}}$ calculated as function of time in a typical real-access (a) and an attack (b).

shown in Equation 2. $M_{\mathcal{D}}$ represents the motion coefficient for the given RoI support $(x, y) \in \mathcal{D}$, with a given support area of $S_{\mathcal{D}}$. Effectively, $M_{\mathcal{D}}$ represents the average absolute gray-scale difference between two consecutive images (I_t and I_{t-1}) in the video stream. In the case of the face, the support region is provided by same face detector used during the acquisition. The background is computed by making $\mathcal{D}$ the whole image and subtracting the part relative to the face prior to averaging. Noise arriving from the face localisation is avoided by considering the face region not to move between two consecutive images.

$$M_{\mathcal{D}} = \frac{1}{S_{\mathcal{D}}} \sum_{(\mathrm{x},\mathrm{y})\in\mathcal{D}} |I_{\mathrm{t}}(\mathcal{D}) - I_{\mathrm{t}-1}(\mathcal{D})| \qquad (2)$$

The calculation of $M_{\mathcal{D}}$, even considering both RoIs, can be implemented in a very efficient manner allowing the variable to be computed for every two images in the sequence being observed. Figure 2 shows the evolution of $M_{\mathcal{D}}$ for both face and background in two scenarios: a real-access (a) and an attack (b). As it can be observed, the motion variations exhibit greater correlation in the case of an attack. Also note that the $M_{\mathcal{D}}$ signal for an attack seems to exhibit more variations in time, characterized by the amount of signal energy and higher-frequency components.

4.2. Classification

To input the motion coefficients into a classifier and avoid the variability in time, we extract 5 quantities that describe the signal pattern for windows of N non-overlapping images. The 5 quantities are the minimum of the signal in the window, the maximum, the average,

the standard deviation and the ratio R between the spectra sum for all non-DC components components and the DC component itself taking as basis the N-point Fourier transform of the signal at the window (see Equation 3).

$$R = \frac{\sum_{i=1}^{N} |\mathrm{FFT}_i|}{|\mathrm{FFT}_0|} \tag{3}$$

These quantities allow for a trained classifier to evaluate the degree of synchronized *shaking* within the scene, during the period of time defined by N. If there is no movement (fixed support attack) or too much movement (hand-based attack), the input data is likely to come from a spoof attempt. Normal accesses will exhibit decorrelated movement between the two RoIs as normal users move independently from the background.

4.3. Temporal Processing

In order to combine the time information with that of the window-based classifier, we accumulate the output over time for every block of N frames and apply a very simple binary decision scheme using a *majority-wins* approach. For every output the threshold τ defined at the EER on the development set is applied and, if the output is greater than τ we label it as a real-access, with a value of 1. Otherwise, we apply a label of 0. After a number M of decisions have been collected, we average the values attributed in each window and check if such a value exceeds 0.5. If that is the case, by majority of votes, we determine the video comes from a real-access, otherwise, a spoof attempt.

5. Experiments

For this work, the window size N has been arbitrarily fixed at 20. This value represents roughly a second of activity and allows the counter measure to be applied in discrete moments when integrated into a face recognition framework. After the calculation of $M_{\mathcal{D}}$, the input signal is broken into 20-point non-overlapping windows and fed to a multi-layer perceptron (MLP) classifier [24] with 5 hidden neurons, matching the number of inputs, and a single output node. Tries to increase the number of neurons on the hidden layer did not show better generalization and increases the probability of over-fitting. Reducing the number of neurons in such a layer showed performance degradation.

The network is trained using a resilient back-propagation algorithm [25] and exclusively using the training set video sequences. To avoid over-fitting and improve generalization, the development set is used to stop the training procedure as soon as the squared-output error on such a set reaches its first minimum. After training, a threshold is chosen on the equal-error rate (EER) using the development set and, based on such a value, the test set is used to evaluate the final performance of the classifier.

	Development		Test		
Support	FAR	FRR	FAR	FRR	HTER
Hand	10.91%	10.93%	6.82%	7.71%	7.27%
Fixed	10.30%	10.28%	14.77%	7.29%	11.03%
All	10.61%	10.65%	10.45%	7.50%	8.98%

Table 2. Summary of results by analyzing shaking behavior on print attacks.

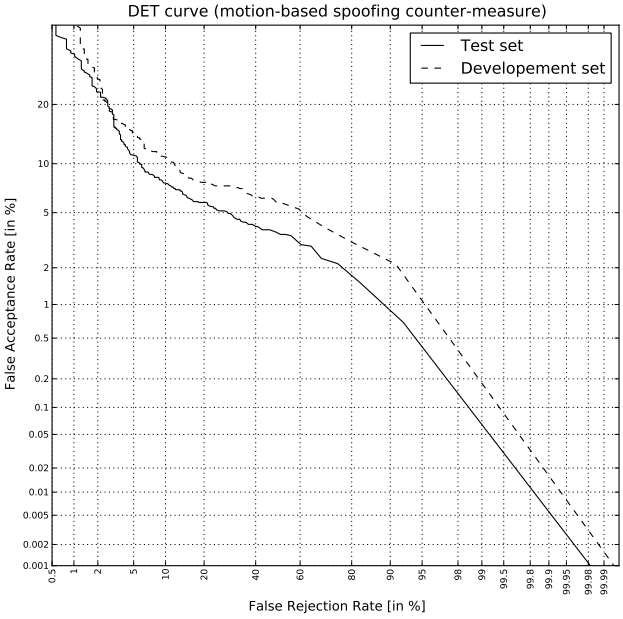

Figure 3. DET curves for the classifier leading to results in Table 2. The curves are traced using all data available in the respective sets (hand + fixed-support).

5.1. Results

Table 2 summarizes the best results for the print-attack development and test set classification. Figure 3 shows the DET curve for both the test and development sets taking as base the same classifier.

Naturally, for the training procedure, the MLP weights are initialized randomly. To assure stable convergence we repeated the training procedure several times (> 10), verifying equivalent minima is reached for the squared error and similar generalization is achieved by the MLP network. Other MLP's, trained using the same parameters, achieve similar results, with difference of only 1 percent on the test HTER.

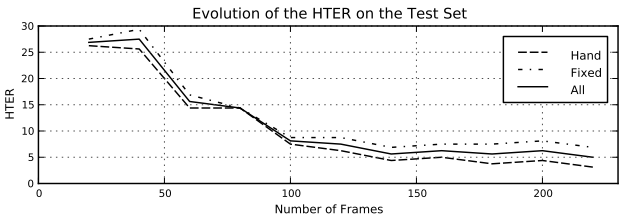

Figure 4. Half-total error rate on the test set with thresholds chosen *a priori* on the EER at the development set, as time passes.

Results shown so far take into consideration overall classification accuracy for windows of 20 images and no attention has been given to the order in which such windows happen. In other words, during the training procedure, no special attention is given to the time variable associated with each of 20-frame window extracted from the motion signal $M_{\mathcal{D}}$. It is natural to assume though one cannot reach the level of accuracy at Figure 3 by only looking at the first 20 images of a scene.

Figure 4 shows the evolution of the half-total error rate on the test set, with a threshold chosen *a priori* on the EER at the development set as time passes using the *majority-wins* approach as defined on Section 4.

5.2. Discussion

Results in Table 2 show one can achieve about 9% HTER on the test set with a threshold chosen *a priori* on the development set using the proposed motion-based technique as a counter-measure to spoofing. It is also possible to observe that the trained classifier performs better to discriminate attacks executed with the attacker's own hands (∼7.3% HTER on the test set). This effect is related to the total energy of movement on those attacks and the way such energy is distributed across the spectra of motion M. In fixed-support attacks there is no movement in the scene and all captured information concerning the motion M can only be originated from encoding noise. In such case, the proposed classification scheme is still able to perform well (∼11% HTER on the test set). Real-accesses sit in between these two types of attacks when analyzed with respect to their motion pattern M. A person trying to be recognized will hold still as much as possible and observed motion is only originated by involuntary movements of eyes, natural head swings or voluntary (less important) lip movements.

Figure 4 summarizes the evolution in time of the accuracy of the proposed detection scheme. As it can be seen, only after 60 frames (or 3 decisions have been taken) the first drop in the HTER for the test set is observed. This effect is due to the *majority-wins* approach as doubtful cases in which the decision has flipped in the previous slots would only be confirmed after the arrival of the third decision. Advancing on time will only reveal improvement again when the fifth decision is taken, resolving the draws in the first four rounds. After this point, the HTER converges smoothly so it is about 5% after 220 frames which represents about 9 seconds of time. We again confirm our expectations that the system would work better for hand-based attacks by observing that the HTER on that subset is always smaller or equal to the values on the fixed-support column.

6. Conclusion

One of the easiest ways to spoof a 2-D face recognition system is by the use of photographs of attacked identities. This problem has been understood for nearly a decade now and, yet, no consensus seems to exist on techniques or best-practices to avoid this. Literature is scarce and results are difficult to generalize. To remedy this aspect, we made public a PRINT-ATTACK dataset and exemplified how to use its companion protocol with a motion-based algorithm that detects correlations between the client head movements and the scene context. The database is sufficiently large and contains a diverse set of spoofing attacks under different conditions.

Paper-based print attacks represent only one of the many ways to attack a 2-D face recognition system. With the decreasing prices of mobile phones and high-resolution portable devices, one can expect these supports to replace paper also for spoofing. To cover more ground, we should consider the recording of new attacks using these display media as well as different paper type such as those used for customer photographic prints.

Other possible types of attack that need to be considered are those using videos and three-dimensional masks. They represent respectively the second and third most probable attack strategies after photographs. We can also expect interest would grow in this direction.

One variable often disregarded in research is the motion pattern introduced by the attacker, while displaying the device with the photograph of the client face being attacked. A natural extension to this dataset and to this work is therefore to explore different attackers and lighting conditions. Such variables will likely impact motion-based counter-measures.

7. Acknowledgments

The authors would like to thank the Swiss Innovation Agency (CTI Project Replay) and the FP7 European TABULA RASA Project (257289) for their financial support. The authors would also like to thank Christine Marcel and Flavio Tarsetti for their valuable contributions to the creation of the PRINT-ATTACK dataset.

References

[1] S. A. C. Schuckers, "Spoofing and anti-spoofing measures," *Security*, vol. 7, no. 4, pp. 56–62, 2002.

[2] J. Galbally, C. McCool, J. Fierrez, S. Marcel, and J. Ortega-Garcia, "On the vulnerability of face verification systems to hill-climbing attacks," vol. 43(3). Pattern Recognition, 2010, pp. 1027–1038.

[3] A. K. Jain, P. Flynn, and A. A. Ross, Eds., *Handbook of Biometrics*. Springer-Verlag, 2008.

[4] N. M. Duc and B. Q. Minh, "Your face is not your password." Black Hat Conference, 2009.

[5] R. W. Frischholz and U. Dieckmann, "Bioid: A multimodal biometric identification system," *Computer*, vol. 33 issue 2, pp. 64–68, February 2000.

[6] I. Pavlidis and P. Symosek, "The imaging issue in an automatic face/disguise detection system," in *IEEE Workshop on Computer Vision Beyond the Visible Spectrum: Methods and Applications*, 2000.

[7] N. Eveno and L. Besacier, "Co-inertia analysis for "liveness" test in audio-visual biometrics," in *Image and Signal Processing and Analysis, 2005. ISPA 2005. Proceedings of the 4th International Symposium on*, September 2005, pp. 257–261.

[8] K. Kollreider, H. Fronthaler, and J. Bigun, "Verifying liveness by multiple experts in face biometrics," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, June 2008, pp. 1–6.

[9] L. Thalheim, J. Krissler, and P.-M. Ziegler, "Body check: Biometric access protection devices and their programs put to the test," *Heise Online*, 2002.

[10] X. Tan, Y. Li, J. Liu, and L. Jiang, "Face liveness detection from a single image with sparse low rank bilinear discriminative model," *Computer Vision ECCV 2010*, vol. 6316, pp. 504–517, 2010.

[11] Y. Li and X. Tan, "An anti-photo spoof method in face recognition based on the analysis of fourier spectra with sparse logistic regression," in *Chinese Conference on Pattern Recognition*, 2009.

[12] J. Li, Y. Wang, T. Tan, and A. K. Jain, "Live face detection based on the analysis of fourier spectra," in *In Biometric Technology for Human Identification*, 2004, pp. 296–303.

[13] K. Kollreider, H. Fronthaler, and J. Bigun, "Non-intrusive liveness detection by face images," *Image and Vision Computing*, vol. 27, no. 3, pp. 233–244, 2009.

[14] ——, "Verifying liveness by multiple experts in face biometrics," in *Computer Society Conference on Computer Vision and Pattern Recognition Workshops*. IEEE, 2008, pp. 1–6.

[15] W. Bao, H. Li, N. Li, and W. Jiang, "A liveness detection method for face recognition based on optical flow field," in *2009 International Conference on Image Analysis and Signal Processing*. IEEE, 2009, pp. 233–236.

[16] J. Bai, T. Ng, X. Gao, and Y. Shi, "Is physics-based liveness detection truly possible with a single image?" in *International Symposium on Circuits and Systems*. IEEE, 2010, p. 3425–3428.

[17] V. N. Vapnik, *The nature of statistical learning theory*. Springer, 1995.

[18] G. Pan, L. Sun, Z. Wu, and S. Lao, "Eyeblink-based anti-spoofing in face recognition from a generic webcamera," *IEEE 11th International Conference on Computer Vision (2007)*, pp. 1–8, 2007.

[19] G. Pan, Z. Wu, and L. Sun, "Liveness detection for face recognition," *Recent Advances in Face Recognition*, pp. 109–124, December 2008.

[20] G. Pan, L. Sun, Z. Wu, and Y. Wang, "Monocular camera-based face liveness detection by combining eyeblink and scene context," *Journal of Telecommunication Systems*, 2009.

[21] T. Ahonen, A. Hadid, and M. Pietikainen, "Face Recognition with Local Binary Patterns," *Lecture Notes in Computer Science*, pp. 469–481, 2004.

[22] B. Froba and A. Ernst, "Face detection with the modified census transform," *IEEE International Conference on Automatic Face and Gesture Recognition*, pp. 91–96, 2004.

[23] A. Martin, G. Doddington, T. Kamm, M. Ordowski, and M. Przybocki, "The det curve in assessment of detection task performance," in *Fifth European Conference on Speech Communication and Technology*, 1997, pp. 1895–1898.

[24] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*, 1st ed. Springer, October 2007.

[25] M. Riedmiller and H. Braun, "A direct adaptive method for faster backpropagation learning: the rprop algorithm," in *IEEE International Conference on Neural Networks*, vol. 1, no. 3. IEEE, 1993, pp. 586–591.