

Allan da Silva Pinto

**“A Countermeasure Method for
Video-Based Face Spoofing Attacks”**

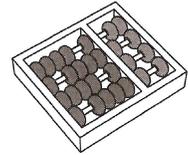
*“Detecção de Tentativas de Ataque com Vídeos
Digitais em Sistemas de Biometria de Face”*

**CAMPINAS
2013**



University of Campinas
Institute of Computing

*Universidade Estadual de Campinas
Instituto de Computação*



Allan da Silva Pinto

“A Countermeasure Method for Video-Based Face Spoofing Attacks”

Supervisor: Prof. Dr. Anderson de Rezende Rocha
Orientador(a):

“Detecção de Tentativas de Ataque com Vídeos Digitais em Sistemas de Biometria de Face”

MSc Dissertation presented to the Post Graduate Program of the Institute of Computing of the University of Campinas to obtain a Master degree in Computer Science.

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Ciência da Computação do Instituto de Computação da Universidade Estadual de Campinas para obtenção do título de Mestre em Ciência da Computação.

THIS VOLUME CORRESPONDS TO THE FINAL VERSION OF THE DISSERTATION DEFENDED BY ALLAN DA SILVA PINTO, UNDER THE SUPERVISION OF PROF. DR. ANDERSON DE REZENDE ROCHA.

ESTE EXEMPLAR CORRESPONDE À VERSÃO FINAL DA DISSERTAÇÃO DEFENDIDA POR ALLAN DA SILVA PINTO, SOB ORIENTAÇÃO DE PROF. DR. ANDERSON DE REZENDE ROCHA.

A handwritten signature in blue ink that reads "Anderson de Rezende Rocha".

Supervisor's signature / *Assinatura do Orientador(a)*

CAMPINAS

2013

Ficha catalográfica
Universidade Estadual de Campinas
Biblioteca do Instituto de Matemática, Estatística e Computação Científica
Ana Regina Machado - CRB 8/5467

P658c Pinto, Allan da Silva, 1984-
A countermeasure method for video-based face spoofing attacks / Allan da Silva Pinto. – Campinas, SP : [s.n.], 2013.

Orientador: Anderson de Rezende Rocha.
Dissertação (mestrado) – Universidade Estadual de Campinas, Instituto de Computação.

1. Identificação biométrica. 2. Ataques de falsificação de face. 3. Ataques de falsificação baseado em vídeo. 4. Ritmo visual. 5. Análise de Fourier - Processamento de dados. I. Rocha, Anderson de Rezende, 1980-. II. Universidade Estadual de Campinas. Instituto de Computação. III. Título.

Informações para Biblioteca Digital

Título em outro idioma: Detecção de tentativas de ataque com vídeos digitais em sistemas de biometria de face

Palavras-chave em inglês:

Biometric identification

Face spoofing attacks

Video-based face spoofing

Visual rhythm

Fourier analysis - Data processing

Área de concentração: Ciência da Computação

Titulação: Mestre em Ciência da Computação

Banca examinadora:

Anderson de Rezende Rocha [Orientador]

Marina Atsumi Oikawa

Fábio Augusto Menocci Cappabianco

Data de defesa: 24-10-2013

Programa de Pós-Graduação: Ciência da Computação

TERMO DE APROVAÇÃO

Dissertação Defendida e Aprovada em 24 de outubro de 2013, pela Banca examinadora composta pelos Professores Doutores:

Fábio Cappabianco

Prof. Dr. Fábio Augusto Menocci Cappabianco
ICT / UNIFESP

Marina A. Oikawa

Prof^a. Dr^a. Marina Atsumi Oikawa
IC / UNICAMP

Anderson de Rezende Rocha

Prof. Dr. Anderson de Rezende Rocha
IC / UNICAMP

A Countermeasure Method for Video-Based Face Spoofing Attacks

Allan da Silva Pinto¹

October 24, 2013

Examiner Board/*Banca Examinadora*:

- Prof. Dr. Anderson de Rezende Rocha (Supervisor/*Orientador*)
- Dra. Marina Atsumi Oikawa
Institute of Computing - UNICAMP
- Prof. Dr. Fábio Augusto Menocci Cappabianco
Institute of Science and Technology - UNIFESP
- Prof. Dr. Ricardo da Silva Torres
Institute of Computing - UNICAMP (Substitute/*Suplente*)
- Prof. Dr. Joao Paulo Papa
Computer Science Department - UNESP (Substitute/*Suplente*)

¹Financial support: Capes scholarship 2011–2012

Abstract

Spoofing attacks can be easily accomplished in a facial biometric system wherein users without access privileges attempt to authenticate themselves as valid users, in which an impostor needs only a photograph or a video with facial information of a legitimate user. Even with recent advances in biometrics, information forensics and security, vulnerability of facial biometric systems against spoofing attack is still an open problem. Even though several methods have been proposed for photo-based spoofing attack detection, attacks performed with videos have been vastly overlooked, which hinders the use of facial biometric systems in modern applications. In this dissertation, we present an algorithm for video-based spoofing attack detection through the analysis of global information which is invariant to the video content, since we discard video contents and only analyze content-independent noise signatures present in the video related to the acquisition unique processes. Our approach takes advantage of noise signatures generated by the recaptured video to distinguish between fake and valid access videos. To capture noise properties and obtain a compact representation of them, we use the Fourier spectrum followed by the computation of video visual rhythms and the extraction of different characterization methods (e.g., histogram of oriented gradients, local binary patterns and gray-level co-occurrence matrices), used as feature descriptors. To evaluate the effectiveness of the proposed approach, we introduce the novel Unicamp Video-Attack Database (UVAD) which comprises 14,870 videos composed of real access and spoofing attack videos. In addition, we evaluate the proposed method using the Replay-Attack Database, which contain photo-based and video-based face spoofing attacks.

Resumo

Ataques de falsificação constituem um tipo de ataque que pode ser facilmente realizado em um sistema de biometria de face por usuários sem privilégios de acesso que tentam se autenticar como usuários válidos ou legítimos. Para isto, o usuário impostor necessita de apenas uma fotografia ou um vídeo com as informações faciais de um usuário legítimo, alvo do ataque, que pode ser obtido em redes sociais, páginas pessoais, entre outros. Mesmo com os recentes avanços nas áreas de biometria, forense e segurança da informação, a vulnerabilidade dos sistemas de biometria de face frente a ataques de falsificação de face é ainda um problema em aberto. Embora diversos métodos têm sido propostos para detectar ataques realizados com fotografias, o problema de detecção de ataques realizados com vídeos e modelos 3D tem sido desconsiderados, o que limita o poder de defesa e contramedidas dos sistemas de autenticação de aplicações modernas, principalmente em aplicações à web e dispositivos móveis. Nesta dissertação, nós apresentamos um método para detecção de ataque de falsificação de face realizado com vídeo que utiliza as informações globais presentes nos vídeos, sendo invariante ao conteúdo. Em nosso método, calculamos e analisamos a assinatura de ruído presente no vídeo, gerado pela sua recaptura, para distinguir vídeos de acessos válidos de vídeos falsos. Para capturar as propriedades de ruído e obter uma compacta representação, nós usamos o espectro de Fourier seguido do cálculo do ritmo visual do vídeo e da extração de características por meio de diferentes métodos de caracterização (e.g., histogramas de gradientes orientados, padrões binários locais e matrizes de co-ocorrência em tons de cinza). Para avaliar a efetividade da abordagem proposta, nós construímos a base de dados Unicamp Video-Attack Database (UVAD) que consiste de 14.870 vídeos de acesso válido e de tentativas de ataque. Além disso, nós avaliamos o método proposto usando o Replay-Attack Database, o qual contém tentativas de ataques de falsificação realizados com fotografias e vídeos.

Acknowledgements

I thank God for having given me health, strength and mood to get up every day and pursue my dream. In these two years and six months the life taught me once again that dedication, perseverance, courage and gratitude for everyone and everything we have around us are virtues essential to overcome all the difficulties and achieve our goals.

I thank my advisor Anderson Rocha for the excellent advising throughout this work. I am grateful for the advices, the comments and compliments, which contributed greatly to this work and surely will have great effects throughout my professional career.

I also thank professors Hélio Pedrini and William Robson Schwartz who closely worked with my advisor and me in this research. I'm grateful for their suggestions during the meetings of the group, in which I learned important aspects related to my research. I thank the group also for helping me in times of difficulties or doubts I had when conducting this work and also the various revisions made in the submitted articles. You are examples of great professionals to be followed.

I thank my wife Euridinéia for her words of confidence, comfort and motivation, which helped me in times of trouble. I thank her love, care, support and for believing that education will take us to a better future. Euridinéia, I love you so much.

I thank my parents for always teaching me to pursue a better future with dignity and with hard work and dedication. I thank them for always being concerned about my education and for supporting me in all my decisions, from of the classes before college, graduation and now in the master's program. I also thank my wife's parents and sister for having supported us in our life project and helping us in times of difficulties.

I thank all the volunteers who contributed with the construction of the huge video database used in this project and Unicamp by allowing the recordings made on Campus. In special, I thank the Faculty of Mechanical Engineering, the Board of Directors of Logistics and Infrastructure for Education (Basic Cycle), the institutes of Computing, Economy, Biology, and Mathematics, Statistics and Scientific Computing for having granted appropriated places to carry out the shooting. Finally, I thank the professor Rodolfo Azevedo by providing the tablets used in the experiments.

I thank all the professors with whom I took classes during the master's program and

the friends and colleagues of the laboratory for support and reception upon my arrival at Unicamp. Our daily experience in the laboratory taught me many things that contributed to my professional and personal growth.

I thank Institute of Computing and Recod Laboratory for providing all infrastructure needed for the execution of this work. I also thank staffs of the Commission Graduate and all the staff of the Institute of Computing that contributed directly or indirectly to this work.

I thank my students at Faculty Max Planck in Indaiatuba for their suggestions, comments and compliments regarding the classes I have been teaching there throughout the last one and a half year, which greatly contributed to my professional growth.

Finally, I thank CAPES for the financial support of this research during my first year of study in the master's program.

Contents

Abstract	ix
Resumo	xi
Acknowledgements	xiii
1 Introduction	1
1.1 Conceptualization and Motivation	1
1.2 Objective	3
1.3 Contributions	4
1.4 Related Publications with this Dissertation	5
1.5 Dissertation Outline	5
2 Related Work	6
2.1 Existing Databases	7
2.2 Motion Analysis and Clues of the Scene	8
2.3 Texture and Frequency Analysis	10
2.4 Other Approaches	12
2.5 Problems with the Existing Approaches	13
3 Proposed Method	15
3.1 Calculation of the Residual Noise Videos	15
3.2 Calculation of the Fourier Spectrum Videos	16
3.3 Calculation of the Visual Rhythms	17
3.4 Feature Extraction	18
3.4.1 Gray Level Co-occurrence Matrices (GLCM)	20
3.4.2 Local Binary Patterns (LBP)	21
3.4.3 Histogram of Oriented Gradient (HOG)	22
3.5 Supervised Learning Algorithms	22

4	Database Creation	25
5	Experimental Results	28
5.1	Protocols for the UVAD Database	28
5.2	Parameters for the Filtering Process, Visual Rhythm Analysis and Classification	29
5.3	Experiment I: Influence of the Display Devices	30
5.4	Experiment II: Influence of the Biometric Sensors	32
5.5	Experiment III: Attack with Tablets	35
5.6	Experiment IV: Influence of the Feature Descriptors	37
5.7	Experiment V: Comparison to a State-of-the-Art Method	40
5.8	Experiment VI: Evaluation of the Method in the Replay-Attack Database .	41
6	Conclusions and Future Work	42
	Bibliography	44
A	Learning Algorithms	50
A.1	SVM Algorithm	50
A.2	PLS Algorithm	54

List of Tables

4.1	Comparison of the UVAD proposed database and other available reference benchmarks in the literature.	27
5.1	Number of features (dimensions) using either the direct pixel intensities as features or the features extracted by image description methods.	30
5.2	Results showing Area Under the receiver operating characteristic Curve (AUC) of the experiment analyzing the influence of the display devices using a PLS Classifier and Median Filter.	31
5.3	Results (AUC) of the experiment analyzing the influence of the display devices using a PLS Classifier and Gaussian Filter.	32
5.4	Results (AUC) of the experiment analyzing the influence of the display devices using a SVM Classifier and Median Filter.	32
5.5	Results (AUC) of the experiment analyzing the influence of the display devices using a SVM Classifier and Gaussian Filter.	33
5.6	Results (AUC) of the experiment analyzing the influence of the biometric sensors using a PLS Classifier and Median Filter.	33
5.7	Results (AUC) of the experiment analyzing the influence of the biometric sensors using a PLS Classifier and Gaussian Filter.	34
5.8	Results (AUC) of the experiment analyzing the influence of the biometric sensors using a SVM Classifier and Median Filter.	34
5.9	Results (AUC) of the experiment analyzing the influence of the biometric sensors using a SVM Classifier and Gaussian Filter.	34
5.10	Results (AUC) of the experiment analyzing attacks with tablets using a PLS Classifier and Median Filter.	35
5.11	Results (AUC) of the experiment analyzing attacks with tablets using a PLS Classifier and Gaussian Filter.	35
5.12	Results (AUC) of the experiment analyzing attacks with tablets using a SVM Classifier and Median Filter.	36
5.13	Results (AUC) of the experiment analyzing attacks with tablets using a SVM Classifier and Gaussian Filter.	36

5.14	Results showing AUC using the PLS classification technique.	38
5.15	Results showing AUC using the SVM classification technique.	39
5.16	Comparison between the method presented in [51] and the method proposed in this work considering the use of horizontal visual rhythm, Median filter and SVM classification technique. The Results showing AUC. According to McNemar test, the methods are statistically different.	40
5.17	Results showing AUC for the test set.	41

List of Figures

1.1	General biometric system and its vulnerability points. (a) a threat resulting from an attack on the biometric sensor, presenting a synthetic biometric data (fake); (b), (c) and (d) represent threats resulting from re-submission of a biometric latent signal previously stored in the communication channel; (e) attack on the matching algorithm in order to produce a higher or lower score; (f) an attack on the communication channel between the enrollment center and the database (the control of this channel allows an attacker to overwrite the template that is sent to the biometric database); (g) an attack on the actual database itself, which could result in corrupted models, denial of service to the person associated to the corrupted model, or fraudulent authorization of an individual; (h) an attack that consists in overwriting the output of the matching algorithm, bypassing the authentication process. Image adapted from Buhan et al. [10].	2
3.1	Proposed method. Given a training set consisting of videos of valid accesses, video-based spoofs and a test video, we first extract a noise signature of every video (training and testing) and calculate the Fourier Spectrum on logarithmic scale for each video frame and then summarize each video by means of its visual rhythm. Considering the training samples, we train a classifier using a summarized version of the visual rhythms obtained by the estimation of the gray level co-occurrence matrices, as features. With a trained classifier, we are able to test a visual rhythm for a given video under investigation and point out whether it is a valid access or a spoof.	16
3.2	Example of a video frame of the spectra generated from (a) a valid video and (b) an attack video.	17
3.3	Examples of visual rhythms constructed from (a)-(b) central horizontal lines and from (c)-(d) central vertical lines. Note that the visual rhythm obtained from horizontal lines has been rotated 90 degrees for visualization purposes.	19

3.4	Examples of spectra whose highest responses are not only at the abscissa and ordinates axes.	19
3.5	Examples of visual rhythms constructed by a traversal in zig-zag.	20
3.6	(a) Possibles angular relationship θ between the center pixel ‘•’ and its neighbors that are at distance $d = 1$ and (b) An example of extraction of textural patterns of image I with the GLCM descriptor.	21
3.7	A window of size 3×3 is thresholded by the value of the central pixel. The pixel values are then multiplied by binomial weights and summed to obtain an LBP number to this window. Thus, LBP can produce up to $2^8 = 256$ different texture patterns, and a histogram with 256 bins is then calculated and used as a texture descriptor.	22
3.8	Example of extraction of the HOG descriptor of an input image I . After the color and gamma normalization in the image I , the resultant image I_{NORM} is divided into cells of size 8×8 pixels. In sequence, for each cell is calculated a histogram of gradients directions with nine bins. Then a set of four cells is grouped into a block and a normalization of the histograms calculated from each cell that compose the block is performed.	23
4.1	Examples of valid access video frames and attempted attack video frames that comprise the UVAD.	26
A.1	Considering a simple binary classification problem that consist in separates balls from triangles. The optimal hyperplane is represented by solid line and there is a weight vector \mathbf{w} and a threshold b such that $y_i \cdot ((\mathbf{w} \cdot \mathbf{x}_i) > 0$. Rescaling \mathbf{w} and b such that the points closest to the hyperplane satisfy the equation $ (\mathbf{w} \cdot \mathbf{x}_i) + b = 1$, we obtain a form (\mathbf{w}, b) of the hyperplane with $y_i((\mathbf{w} \cdot \mathbf{x}_i) + b) \geq 1$. Image adapted from [11].	52

Chapter 1

Introduction

1.1 Conceptualization and Motivation

Biometric authentication or biometrics is a technology concerned with recognizing humans in an automatic and unique manner based on behavior, physical and chemical traits. Examples of physical traits include fingerprint, geometric and veins of the hand, face, iris and retina. Speech and handwriting are examples of behavior traits and skin odor and DNA (Deoxyribonucleic Acid) information are examples of chemical traits [24].

In the last decades, biometrics have emerged as an important mechanism for access control that has been used in many applications, in which the traditional methods including the ones based on knowledge (e.g., keywords, secret question) or based on tokens (e.g., smart cards) might be ineffective since they are easily shared, lost, stolen or manipulated. In contrast, the biometric access control has been shown as a natural and reliable authentication method [24].

Access control can be seen as a verification problem wherein the authentication of a user is performed by reading and comparing the input biometric data captured by an acquisition sensor (query) with the biometric data of the same user previously stored in a database (template). The comparison between the query and the template is performed by a matching algorithm which produces a similarity score used to decide whether or not the access should be granted to the user.

Although biometric authentication is considered a secure and reliable access control mechanism, it becomes an easy target for attacks if protective measures are not implemented. Figure 1.1 shows a general biometric authentication system without any protective measure and some points of vulnerabilities. Buhan et al. [10] provide more details about abuses in biometric systems.

Spoofing attack is a type of attack wherein an impostor presents a fake biometric data to the acquisition sensor with the goal of authenticating oneself as a legitimate user,

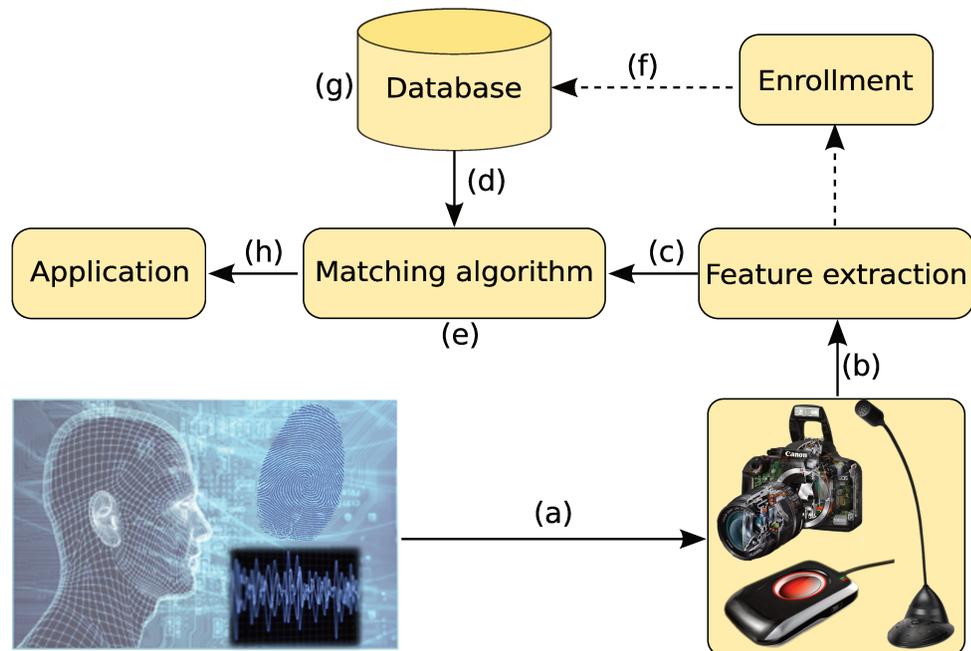


Figure 1.1: General biometric system and its vulnerability points. (a) a threat resulting from an attack on the biometric sensor, presenting a synthetic biometric data (fake); (b), (c) and (d) represent threats resulting from re-submission of a biometric latent signal previously stored in the communication channel; (e) attack on the matching algorithm in order to produce a higher or lower score; (f) an attack on the communication channel between the enrollment center and the database (the control of this channel allows an attacker to overwrite the template that is sent to the biometric database); (g) an attack on the actual database itself, which could result in corrupted models, denial of service to the person associated to the corrupted model, or fraudulent authorization of an individual; (h) an attack that consists in overwriting the output of the matching algorithm, bypassing the authentication process. Image adapted from Buhan et al. [10].

illustrated in Figure 1.1(a). Depending on the biometric trait used by the system, this mode of attack can be easily accomplished because some biometric data can be synthetically reproduced without much effort. Face biometric systems are highly vulnerable to such attacks since facial traits are widely available on the Internet, on personal websites and social networks such as Facebook¹, MySpace², YouTube³. In addition, we can easily collect facial samples of a person with a digital camera.

In the context of face biometrics, an attempt of spoofing attack can be performed by presenting to the acquisition sensor a photograph, a video or a 3D face model of a

¹<http://www.facebook.com>

²<http://www.myspace.com>

³<http://www.youtube.com>

legitimate user enrolled in the database. If an impostor succeeds in the attack using any of these approaches, the uniqueness premise of the biometric system is violated, making the system vulnerable [24].

1.2 Objective

Several methods have been proposed in the literature to detect spoofing attacks based on photographs, whereas attacks performed with videos and 3D models have been overlooked. We believe that attacks performed with videos and 3D models (rigid and realistic masks) is more difficult to be detected due to the a best quality of the fake biometric samples. Many methods aim at distinguishing real from fake biometric data based on the fact that artifacts are inserted into the printed samples due to printing process, therefore allowing one to explore attributes related to such artifacts including color, shape and texture [34, 51, 55]. Since photographs are static, another approach is to detect small movements in the face [30, 41, 61]. Recent works [4, 42] investigate context information of the scene (e.g., background information) to detect face liveness.

We believe that the aforementioned approaches are not suitable for detecting video-based attacks directly, especially in high resolution videos. The difficulty in detecting spoofing performed by video lies in the fact that it is easier to deceive an authentication system through a video since the dynamics of the video makes the biometric data more realistic. Furthermore, the content of a video is less affected by degradations in terms of color, shape or texture, unlike the printed images. Finally, we have less artifacts generated during quantization and discretization of the image captured by the imaging sensor in high resolution videos.

In this dissertation, we present a method for detecting video-based face spoofing attacks under the hypothesis that fake and real biometric data contain different acquisition-related noise signatures. To the best of our knowledge, this is the first attempt of dealing with video-based face spoofing based on the analysis of global information that is invariant to the video content. Our solution explores the artifacts added to the biometric samples during the viewing process of the videos in the display devices and noise signatures added during the recapture process performed by the acquisition sensor of the biometric system. Through the spectral analysis of the noise signature and the use of visual rhythms, we designed a feature characterization process able to incorporate temporal information of the behavior of the noise signal from the biometric samples.

To contemplate a more realistic scenario, this dissertation introduces the Unicamp Video-Based Attack Database (UVAD)⁴, specifically developed to evaluate video-based

⁴This database will be make public and freely available. Users present in the database formally authorized the release of their data for scientific purposes.

attacks in order to verify the following aspects/questions:

- The behavior of the method for attempted attacks with high resolution videos;
- The influence of the display devices in our method;
- Whether attacks with tablets are more difficult to be detected;
- The influence of the biometric sensor in our method;
- The best feature characterization to capture the video artifacts;
- Comparison with one of the best anti-spoofing methods for photo-based spoofing attack of notice.

1.3 Contributions

Such verifications can be accomplished due the diversity of the devices used to create the database which comprises valid access and attempted attack videos of 304 different people. Each user was filmed in two sections in different scenarios and lighting conditions. The attempted attack videos were produced using eight different display devices and three digital cameras from different manufacturers. The database has 608 valid access videos and 14,262 videos of video-based attempted spoofing attacks, all in full high definition quality.

In summary, the main contributions of this work are:

- (i) An efficient and effective method for video-based face spoofing attack detection able to recognize attempted attacks carried out with high resolution videos;
- (ii) The creation of a large and publicly available database to evaluate spoofing attacks specific methods performed with videos considering several display devices and different acquisition sensors;
- (iii) The Evaluation of the video characterization process considering different image features such as the Gray-Level Co-occurrence Matrices (GLCM), Histograms of Oriented Gradients (HOG) and Local Binary Patterns Histogram (LBPH) feature descriptors;
- (iv) A detailed study of the video-based spoofing attack problem that yielded important conclusions that certainly will be useful for the proposition of new anti-spoofing methods for video-based attacks.

1.4 Related Publications with this Dissertation

The preparation of scientific papers that reflected the progress of the project and contributions to the literature were done gradually during the second year of this work. The following are publications sorted by date:

1. **Allan Pinto**, William Robson Schwartz, Hélio Pedrini, and Anderson Rocha. A Countermeasure Method for Video-Based Face Spoofing Attacks. In *IEEE Trans. on Information Forensics and Security (T.IFS)*. (Submitted paper, 2013).
2. I. Chingovska, J. Yang, Z. Lei, D. Yi, S. Z. Li, O. Kähm, C. Glaser, N. Damer, A. Kuijper, A. Nouak, J. Komulainen, T. Pereira, S. Gupta, S. Khandelwal, S. Bansal, A. Rai, T. Krushna, D. Goyal, M.-A. Waris, H. Zhang, I. Ahmad, S. Kiranyaz, M. Gabbouj, R. Tronci, M. Pili, N. Sirena, F. Roli, J. Galbally, J. Fierrez, **A. Pinto**, H. Pedrini, W. S. Schwartz, A. Rocha, A. Anjos, S. Marcel. The 2nd Competition on Counter Measures to 2D Face Spoofing Attacks. In *Intl. Conference on Biometrics (ICB)*, 2013, Madri. The 6th IAPR Intl. Conference on Biometrics, 2013.
3. T. Carvalho, **A. Pinto**, E. Silva, F. O. Costa, G. R. Pinheiro, A. Rocha. Crime Scene Investigation (CSI): da Ficção à Realidade. In *Escola Regional de Informática de Minas Gerais (ERI-MG)*, 2012, Juiz de Fora. Simpósio Mineiro de Computação (SMC), 2012.
4. **Allan Pinto**, Hélio Pedrini, William Robson Schwartz, Anderson Rocha. Video-Based Face Spoofing Detection through Visual Rhythm Analysis. In *Conference on Graphics, Patterns and Images (SIBGRAPI)*, 2012, Ouro Preto. Proceedings of the XXV Conference on Graphics, Patterns and Images, 2012.

1.5 Dissertation Outline

We organize the remaining of this dissertation into five sections. Section 2 discusses state-of-the-art methods for detecting spoofing attacks to face biometrics. Section 3 presents the proposed method aiming at dealing with video-based spoofing attacks in face biometrics systems. Section 4 gives details regarding the proposed video-attack database while Section 5 shows and discusses the experimental results. Finally, Section 6 draws the conclusions obtained with this work and presents some possible future work directions.

Chapter 2

Related Work

According to Pan et al. [40], there are four major categories of anti-spoofing methods: data-driven characterization, user behavior modeling, user interaction need, and the presence of additional devices. Solutions that require extra devices are limited due to their high cost, which can prevent the use in large scale (e.g., deployment of an anti-spoofing solution on all ATMs of a banking network). The user cooperation during the biometric authentication can also be used to facilitate spoofing attack detection, however, this procedure lessens the transparency and inserts an additional time in the authentication process. Finally, the user behavior modeling approach (e.g., eye blinking, small face movements) has been considered in the literature for photo-based face spoofing detection, nevertheless, this approach might not work well to video-based spoofing attack detection due to the high dynamics present in video scenes. Solutions based on data-driven characterization explore biometric data by thoroughly searching for evidence and artifacts useful to detect attempted attacks.

In this section, we review the literature on user behavior modeling and data-driven characterization methods, since such methods are preferable in practice because they are non-intrusive and do not require extra devices nor human interaction. Therefore, they are easily integrable to existing face recognition systems. In this category, there are several methods for photo-based spoofing attack detection that explore clues such as motion and frequency analysis, scene information, and texture. Before going any further, however, we first present some available face-related spoofing databases in the literature since most of the methods use one or some of such reference benchmarks.

2.1 Existing Databases

NUAA Database

The NUAA Photograph impostor database [55] comprises 5,105 valid access images and 7,509 fake images collected with a generic webcam, for which was reported an area under the receiver operating characteristic curve (AUC) of 94% classification rate. The images of valid access were collected of 15 identities in three sections in different places and illumination conditions, all with 640×480 pixel resolution. The production of the fake samples were done by taking high resolution photographs of 15 identities with a Canon digital camera. The authors simulated two attack modes: (1) printing photographs in the photo paper in the sizes $6.8\text{cm} \times 10.2\text{cm}$ and $8.9\text{cm} \times 12.7\text{cm}$; and (2) printing the photographs in the A4 70g paper using an HP color printer.

Print-Attack Database

The Print-Attack database [4] contains short videos of valid access and photo-based spoofing attacks of 50 identities. The valid access videos were generated in two different conditions: (1) in a controlled environment with a uniform background illuminated with fluorescent lamp; and (2) an uncontrolled environment with an irregular background illuminated with daylight. Two video sequences were collected for each user using an Apple MacBook webcam, all videos with 320×240 pixel resolution, 25 frames per second (fps) and 15 seconds of duration. The attempted attack videos were generated by taking two high resolution photographs with a Canon PowerShot digital camera of the 50 identities. Then, the photographs were printed on common A4 papers using a Triumph-Adler DCC 2520 color laser printer. The attempted attack videos were produced showing the photographs to the same webcam used in the generation of the valid access videos, considering two attack modes: (1) hand-based attacks wherein the impostor user presents the photographs using her own hands; and (2) fixed-support attacks in which the photographs were glued on a wall so that they do not move during the attempted attacks. In total, 200 access valid videos and 200 attempted attack videos were generated.

CASIA Database

The CASIA database [63] comprises 600 video clips of 50 identities. The videos were filmed in a natural scene with three cameras: a new and an old USB camera both with 640×480 pixel resolution and a Sony NEX-5 digital camera with $1,920 \times 1,080$ pixels. The database contains three attack modes: (1) warped photo attack, (150 640×480 -attempted attack videos); (2) cut photo attack (150 640×480 -attempted attack videos); and (3) video playback using an iPad (150 $1,280 \times 720$ -attempted attack videos). Although this

database has a variety of attacks, some factors hamper the evaluation of other methods with this database. For instance, the authors failed to prevent the downsizing of the videos shown during the simulation of the video-based spoofing attacks which severely damage the videos since such downsizing adds artifacts to the attempted attack videos that are not present in the valid access videos, creating an artificial data separability. Furthermore, the small amount of data and the use of only one device in the creation of the video-based spoofing attacks prevent more refined investigations.

Replay-Attack Database

The Replay-Attack database [13] contains short video recordings of valid access and attempted attacks of 50 identities. Similarly to the Print-Attack [4], the videos were generated with a low resolution webcam with 320×240 pixel resolution, 25 fps and 15 seconds of duration and the video capture process is the same as described in [4]. However, different from [4], two other attempted attack modes are considered: (1) mobile attacks where the impostor user displays photographs and videos in an iPhone screen produced with the same iPhone; and (2) high-definition attacks where the impostor user shows high resolution photographs and videos produced with a Canon PowerShot digital camera using the screen of a 1024×768 -pixel resolution iPad.

2.2 Motion Analysis and Clues of the Scene

Motion analysis of the face region was an early approach used to detect the liveness of biometric samples. In [41], Pan et al. investigated the action of eye blinking to detect attacks performed with photographs. The authors proposed the use of the undirected conditional random field framework to model the action of opening and closing eyes. Tests were performed in a database with 80 videos and 20 identities using a webcam. The solution obtained results with a false alarm rate smaller than 1%.

Li et al. [30] proposed a method for detecting a person's eye blink based on the fact that, for liveness detection, edges vary homo-responsively to the behavior of eye blink over some scales and orientations. Analyzing the trends of Gabor response waves in multi-scale and multi-orientation, the authors choose the five most homo-responsive Gabor response waves to the behavior of eye blink. The authors collected a database with 10 videos of attempted spoofing attacks performed with photographs and 10 videos of valid access, which were correctly classified.

In [61], Xu et al. proposed a method for detecting the eye states formulated as a binary classification problem in which the closed state represents the positive class and the open state the negative class. In order to form the feature vectors to be classified, the

region of the eyes is scanned with N blocks of different sizes for each biometric sample. For each block, three different feature vectors were extracted by using variants of the Local Binary Pattern Histogram method, generating three sets with N feature vectors. Finally, the vectors that form each set were concatenated, producing three feature vectors for each image. The authors collected 11,165 images from which 5,786 were used in the training stage. The best reported detection rate was 98.3%.

Tronci et al. [56] proposed an anti-spoofing method using the motion information and clues that are extracted from the scene considering static and video-based analyses. A static analysis consists of capturing spatial information of the still images using different visual features as color and edge directivity descriptor, fuzzy color and texture histogram, MPEG-7 descriptors, Gabor texture, Tamura texture, RGB and HSV histograms, and JPEG histogram. These analyses are motivated by the loss of quality and by the addition of noise in the biometric samples during the manufacturing process of the photographs. Video-based analysis is performed as a combination of simple measures of motion such as eye blink, mouth movement, facial expression change among others. In the end, a classifier is trained for each feature and a fusion scheme is then performed between the classifiers to decide whether a biometric sample is a fake or not.

Pan et al. [42] extended the method described in [41] including context information of the scene assuming a static facial recognition system whose background is previously known, denoted as reference scene. Similarly to [41], the authors analyzed clues such as eye blink in the face region. Considering a region of the face, within a certain neighborhood, the authors extracted a set of key points and, for each point, they calculated a Local Binary Pattern Histogram. Then, the χ^2 distance function is used to compare these histograms with other previously calculated key points of the reference scene. The validation was performed using a private database created by the authors in which are reported excellent results.

In [4], Anjos et al. proposed a database and a method for photo-based spoofing attack detection assuming a stationary facial recognition system which produced videos of the biometric samples. In this case, the intensity of the relative motion between the region of the face and the background can be used as a clue to distinguish valid access of attempted attacks. The authors calculate a measure of motion for each video frame obtaining a one-dimensional signal, which is described by the extraction of five measures to form a feature vector. The authors validated the method using the Print-Attack database (c.f., Sec. 2.1).

Yan et al. [62] proposed a method to liveness detection based on three scene clues in both spatial and temporal spaces. According to the authors, the non-rigid facial motion and the face-background consistency incorporate temporal information that can help the decision-making process regarding the face liveness. In the non-rigid facial motion analysis, the authors seek a pattern of non-rigid motion in the region of the face using the batch

image alignment method. The face-background consistency is based on the fact that if the face is real, its motion must be totally independent of the background and is performed separating the region of the face from background and analyzing the motion. Finally, the authors perform a banding artifact analysis, which are treated as additive noise. For that, the authors calculated the first order wavelet decomposition of the image. The authors validated the method using the Print-Attack database (c.f., Sec. 2.1) as well as others created by them. Good results were reported.

2.3 Texture and Frequency Analysis

Li et al. [31] proposed an anti-spoofing method for photo-based attempted attacks under the assumption that the faces present in photographs are smaller than the real faces and that the expressions and poses of the faces in the photographs are invariant. According to the authors, these facts are reflected in the image frequency domain whose high frequency components are less expressive in the photographs. Thus, the detection of an attack through photographs is performed by analyzing the 2-D Fourier spectrum of the samples and calculating the energy rate of the high frequency components, which is used as a threshold to decide whether the biometric sample came from a fake face or not.

In [55], Tan et al. proposed an anti-spoofing solution to attempted attacks performed with printed photographs based on Lambertian reflectance to distinguish real from fake biometric samples, assuming that the surface roughness of both classes are different. The authors proposed the use of the Variational Retinex-based and Logarithmic Total Variation methods for estimating the luminance and reflectance of an input image, respectively. Moreover, the calculation of the Fourier spectrum of the filtered image with the Difference of Gaussian is used to capture artifacts inserted into the samples during the printing process of the attack photographs. The authors modeled the detection problem as a binary classification problem and evaluated the use of the Sparse Logistic Regression and Sparse Low Rank Bilinear Logistic Regression methods for classifying the luminance, reflectance, and Fourier spectrum images previously estimated. The authors validated the method using the NUAA Photograph impostor database (c.f., Sec. 2.1).

Peixoto et al. [43] extended the technique proposed in [55] to detect attempted spoofing attacks performed in an environment with poor illumination. This extension is based on the fact that the brightness of the LCD screens affect the images in the recapturing process by allowing that the edges of the images become more susceptible to the blurring effect. Thus, the authors proposed an intermediate step before the reflectance feature extraction by applying an adaptive histogram equalization in the images. The evaluation of the extended algorithm was performed in the NUAA and Yale Face Databases [20]. The achieved results showed that the proposed extension reduced the misclassification

in more than 50% to attempted attacks with high resolution photographs of the NUAA database.

Määttä et al. [34] proposed to solve the photo-based spoofing problem based on the fact that real and fake biometric facial samples differ: (1) in how these objects reflect light, since human faces are 3D objects and faces printed are planar objects; (2) in the pigmentation; and (3) in the quality due to printing defects contained in the photographs. Based on these observations, the authors used the Local Binary Pattern method for capturing micro-textures information. Several Local Binary Pattern Histograms were computed and concatenated, generating a feature vector with 833 dimensions. Finally, the Support Vector Machine (SVM) technique was used to train a binary classifier to decide whether an input sample was fake. The authors evaluated the proposed algorithm considering the NUAA database (c.f. Sec. 2.1), obtaining an AUC of 99%. In [35], the same authors extended their algorithm evaluating the use of the Histogram of Oriented Gradient (HOG)(c.f., Sec. 3.4.3) and the Gabor wavelet descriptors to detect printing defects and improve the texture description of the biometric samples.

Aiming at finding an appropriate feature space suitable to separate real from fake faces produced by printed photographs, Schwartz et al. [51] proposed a solution that explores different properties of the region of the face such as texture, color and shape to obtain a face holistic representation. Considering only the face region, for each frame of the video containing the facial information, it is generated a feature vector formed by combining of different low-level feature descriptors as HOG, Color Frequency (CF) [53], Gray Level Co-occurrence Matrix (GLCM)(c.f., Sec. 3.4.1), and Histograms of Shearlet Coefficients (HSC) [52]. Then, the feature vectors are combined into one feature vector containing a rich spatial-temporal information of the biometric sample and fed to a Partial Least Square classification technique. Excellent results were reported by the authors using the Print-Attack database.

In [25], Kim et al. explored the frequency and texture information to distinguish real faces from faces in photographs. According to the authors, the use of the frequency information makes sense for two reasons: (1) the difference in the existence of 3D shapes leads to the difference in low frequency regions which is closely related to the luminance component; and (2) the difference between real and fake faces generates a disparity in the high frequency information. The motivation for the use of texture information lies in the fact that printed faces tend to lose the richness of texture details. Their method extracts a feature vector from each biometric sample transforming the images to the frequency domain using the Fourier transform and calculating their respective Fourier spectrum in logarithm scale, from which average values of the energy of 32 concentric rings are extracted. These values are concatenated and normalized, generating a feature vector. Texture analysis is performed by using the Local Binary Pattern method. Finally,

fusion of the two binary classifiers is performed, which are trained one at each feature space using the Support Vector Machine technique.

Recently, Zhang et al. [63] proposed a simple algorithm for detecting photo-based attempted spoofing attacks based on the fact that fake faces present lower quality compared with real faces. For a given image captured by the acquisition sensor, four Difference of Gaussian filters (DoG) with different values of σ were used to extract high frequency information, generating four new images that were concatenated and used as input of a binary classifier trained using the Support Vector Machine technique.

In [13], Anjos et al. conducted a study to investigate the potential of texture descriptors based on Local Binary Pattern (LBP)(c.f., Sec. 3.4.2), such as $LBP_{3 \times 3}^{u2}$, transitional (tLBP), direction-coded (dLBP) and modified LBP (mLBP). From the histograms generated from the descriptors mentioned above, the authors evaluated a simple manner to classify them based on histogram comparisons through χ^2 distance. A set of classifiers was considered, such as Linear Discriminant Analysis (LDA) and SVM with a radial basis function as kernel. Evaluations were performed on the NUAA, Print-Attack, and Replay-Attack databases (c.f., Sec. 2.1).

2.4 Other Approaches

Optical flow analysis has also been considered in the literature for photo-based spoofing attack detection. Bao et al. [5] proposed an anti-spoofing solution based on the analysis of the characteristics of the optical flow field generated for a planar and 3D object.

Unlike the faces contained in photographs, which are regular planar objects, real faces are irregular and 3D objects, which lead to a differentiation between the optical flow fields generated for real and fake faces. In [26], Kollreider et al. analyzed the trajectory of three parts of the face: the region between eyes and nose, left ear, and right ear. Using optical flow patterns and a model based on Gabor decomposition, the authors note that, in real faces, these parts of the face move differently from fake faces.

To detect face liveness using 3D information, Marsico et al. [37] proposed an anti-spoofing solution based on the theory of 3D projective invariants. By the fundamental theorem of the invariant geometry, it is possible to show that the cross ratio of five points on the same plane are invariant to rotations if and only if these points satisfy specific collinearity or co-planarity constraints. Thus, six cross-ratio measures are computed to different configurations of points located in non-coplanar regions of the face (e.g. center of eyes, nose tip and chin). If a pose of the face located in front of the acquisition sensor changes, but the computed cross ratio remains constant, the points must be coplanar (i.e., they belong to a planar fake face).

In order to improve the attack detection rate, some authors have proposed fusion

schemes between methods with different approaches. According to Komulainen et al. [28] there is no single method for face spoofing attacks detection sufficiently robust to all types of attacks, due to the diversity of attempted attacks and the acquisition and display devices. With this in mind, some authors have explored fusion schemes between existing methods in the literature.

Komulainen et al. [28] proposed a fusion scheme at score level of the methods based on the texture and motion analysis. For that, a video is divided into overlapping windows with N frames with an overlap of $N-1$ frames. Thus, facial texture analysis is done using only the last frame of each window and the motion correlation analysis is carried out over the whole window. Each observation window produces a score for each approach independent of the other windows. Finally, the fusion between the scores obtained by both methods is done using linear logistic regression.

In order to get diversity among different databases, Pereira et al. [18] proposed a fusion scheme of three anti-spoofing methods [4, 13, 27] that were tuned in two different databases. A selection of classifiers is made to decide which classifiers participate of the fusion scheme and issue the final decision.

Considering face spoofing attacks performed with 3D masks, Erdogmus et al. [19] proposed a database containing videos of 17 subjects that represent valid accesses and attempted attacks performed with 3D masks. The data were collected in three sessions for all subjects and five videos of 300 frames in each session. These data were captured by a Microsoft Kinect sensor and each frame is composed of an RGB image and a depth image. The authors evaluated the properties of micro-textures to distinguish between attempted attack and valid access videos. The micro-texture were extracted by various LPB operators that were used to construct binary classifiers to decide whether a biometric data is real or fake. Similarly, [29] applied the proposed method in [34] to detect attempted attacks performed with 3D masks.

Finally, recent works have been developed in order to evaluate spoofing attacks in multi-modal biometric systems including [2, 3, 8, 9, 36, 46]. In these works, the authors investigate robust fusion schemes for spoofing attacks considering face and fingerprint biometric traits.

2.5 Problems with the Existing Approaches

Approaches based on clues of the scene have strong constraints that make sense only to photo-based spoofing attacks. In the case of attacks performed by video, such constraints certainly will fail due to the dynamic nature of the scene in this type of media (e.g., motion). The static background assumption made in some works described earlier is limiting since the face moves independently of the background in a video-based attempted

spoofing attack. Moreover, the assumption of a background previously known restricts the use of the method since in many applications (e.g., web and mobile applications) the data acquisition is performed remotely in an environment and, therefore, we can not assume a previously known background. Finally, we can easily change the background of an image through image manipulation packages.

In approaches based on optical flow and motion analysis, motion is easily simulated by rotating or bending the photographs. Moreover, such methods should be evaluated by considering video-based attempted spoofing attacks since these media carries motion information and, therefore, has potential to deceive such methods. Another disadvantage of approaches based on motion analysis based approaches is that the additional time required to capture some face motions prevents a fast spoofing detection. For example, a type of motion analysis extensively explored in the literature is the action of eye blink that occurs once every four or six seconds. However, this rate can be reduced to an average of three to eight every six seconds due to psychological factors [30]. In this case, at least 20 seconds are required to detect eye blinking.

Finally, methods based on texture analysis should consider attempted attacks performed with high resolution videos. Photo-based spoofing attacks have a characteristic that facilitates the detection of this type of attack, which is absent in video-based spoofing attacks: the decrease of quality of the biometric sample due to the printing process, since printers have limitations both in terms of resolution and number of colors that can be produced, which directly influence the texture of the biometric sample, being easily captured by texture information.

Finally, the method proposed in this work aims at overcoming such difficulties by capturing acquisition-related noise information features generated by the video recapture. The fact that noise signal is independent of the image signal makes our technique independent of the video content [33]. Furthermore, our method requires only 50 frames (≈ 2 seconds) for detecting the attempted attacks.

Chapter 3

Proposed Method

In this section, we present an algorithm for video-based attempted spoofing attack detection. Our solution relies on the fact that the addition of a noise pattern in the samples is inevitable during the acquisition step of the facial biometric samples. The acquisition process is performed by a camera that has an imaging sensor with thousands of photosensitive transducers that convert light energy into electrical charges, which are converted into a digital signal by an A/D converter. In [33], Lukäs et al. define two types of noise that can be present in an image: the fixed pattern noise (FPN) and the noise resulting from the photo-responsiveness of non-uniform light-sensitive cells (PRNU). The noise pattern has been widely explored in forensic analysis of digital documents as in the problem of identifying the specific camera that acquired a document [33, 45].

During a video-based spoofing attack, we have the insertion of artifacts in the biometric samples captured by the acquisition sensor, such as distortions, flickering, mooring, and banding effect [7]. Such artifacts, loosely referenced in this dissertation as *noise*, are added during the process of generation and viewing process of the attack video frames in display device screens. Thus, the biometric sample extracted of an attack video will probably contain more noise than the real biometric samples. With this in mind, we design a feature characterization process based on noise signatures along with video summarization methods that are used by a classification algorithm to find a decision boundary between real and fake biometric data. Figure 3.1 summarizes the steps of the proposed method, which are explained in detail in the following sections.

3.1 Calculation of the Residual Noise Videos

The first step of the algorithm is to isolate the noise information contained in the videos that were captured by the acquisition sensor, hereinafter referred to as input video ν . A video ν in the domain $2D + t$ can be defined as a sequence of t frames, where each frame

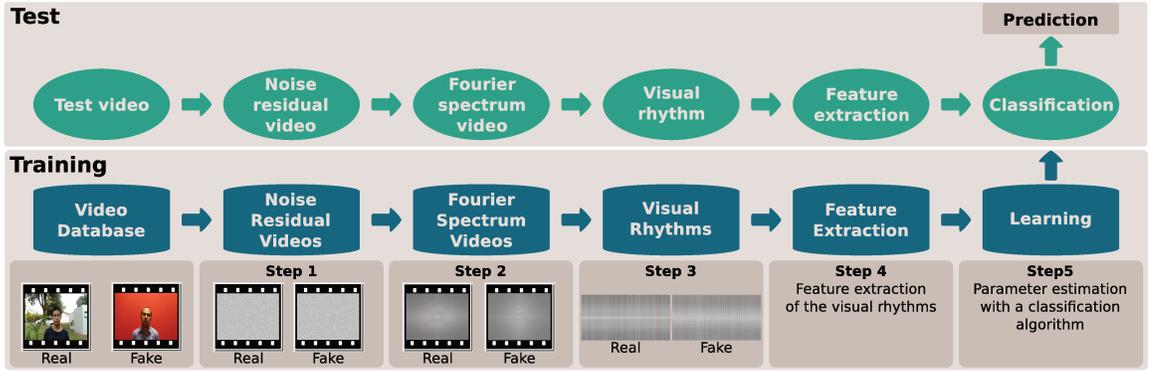


Figure 3.1: Proposed method. Given a training set consisting of videos of valid accesses, video-based spoofs and a test video, we first extract a noise signature of every video (training and testing) and calculate the Fourier Spectrum on logarithmic scale for each video frame and then summarize each video by means of its visual rhythm. Considering the training samples, we train a classifier using a summarized version of the visual rhythms obtained by the estimation of the gray level co-occurrence matrices, as features. With a trained classifier, we are able to test a visual rhythm for a given video under investigation and point out whether it is a valid access or a spoof.

is a function $f(x, y) \in \mathbb{N}^2$ of the brightness of each pixel in the position (x, y) of the scene.

The extraction of the noise signal of the input video ν is performed as follows. The frames in video ν are converted into gray-scale and an instance of ν_{Gray} is submitted to a filtering process using a low-pass filter in order to eliminate noise, generating a filtered video $\nu_{Filtered}$. Then, a frame-by-frame subtraction between the ν_{Gray} e $\nu_{Filtered}$ is performed, generating a new video that contains, mostly, the noise signal in which we are interested, hereinafter named as Residual Noise Video (ν_{NR}), as formalized in Equation 3.1.

$$\begin{cases} \nu_{Filtered}^{(t)} = f(\nu_{Gray}^{(t)}) \\ \nu_{NR}^{(t)} = \nu_{Gray}^{(t)} - \nu_{Filtered}^{(t)} \quad \forall t \in T = \{1, 2, \dots, t\}, \end{cases} \quad (3.1)$$

where $\nu^{(t)} \in \mathbb{N}^2$ is the t -th frame of ν and f a filtering operation.

3.2 Calculation of the Fourier Spectrum Videos

The analysis of the noise pattern and possible artifacts contained in the biometric samples is performed by applying a 2D discrete Fourier transform to each frame of the Noise Residual Video (ν_{NR}) using Equation 3.2. Next, the Fourier spectrum is computed on logarithm scale and with origin at the center of the frame (Equation 3.3). As a result of

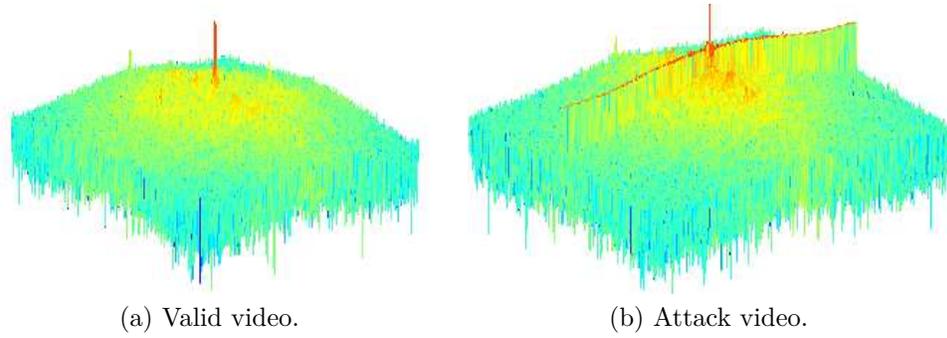


Figure 3.2: Example of a video frame of the spectra generated from (a) a valid video and (b) an attack video.

this process, we end up with a video of the spectra, further on in this document referred to as Fourier Spectrum Videos ν_{FS} . Figures 3.2(a) and 3.2(b) depict the logarithm of the Fourier spectrum of a video frame obtained from a valid access video and from an attempted attack video, respectively.

$$\mathcal{F}(v, u) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} \nu_{NR}(x, y) e^{-j2\pi[(vx/M)+(uy/N)]} \quad (3.2)$$

$$\begin{aligned} |\mathcal{F}(v, u)| &= \sqrt{\mathcal{R}(v, u)^2 + \mathcal{I}(v, u)^2} \\ \nu_{FS}(v, u) &= \log(1 + |\mathcal{F}(v, u)|) \end{aligned} \quad (3.3)$$

3.3 Calculation of the Visual Rhythms

In order to capture the temporal information contained in the Fourier Spectrum Videos (ν_{FS}) and summarize their content, we employ the visual rhythm technique [15]. Visual rhythm is a simplification of a video content in a 2D image obtained by sampling regions of the video. Applications of this concept can be found in the work by Chun et al. [14] that use visual rhythms for fast text caption localization on video, and Guimarães et al. [21] who propose a method for gradual transition detection in videos. The use of visual rhythm in our work is crucial since it allows us to capture patterns that are present in the Fourier Spectrum Videos providing an effective way of viewing a video as a still image.

Considering a video ν in the $2D + t$ domain with t frames of dimension $W \times H$ pixels, the visual rhythm I_{ν_R} is a representation of the video ν , in which regions of interest of

each frame are sampled and concatenated to form a new image, called visual rhythm. The regions of interest must be carefully chosen to be able to capture the patterns contained in ν_{FS} . Formally, a visual rhythm I_{ν_R} of a video ν can be defined by

$$I_{\nu_R}(z, t) = \nu(x(z), y(z), t), \quad (3.4)$$

where $x(z)$ and $y(z)$ are functions of the independent variable z . The visual rhythm is a two-dimensional image whose vertical z axis consists of a certain group of pixels extracted from video ν and the samples are accumulated along the time t . Therefore, according to the mapping of $x(z)$ and $y(z)$, we can generate several types of visual rhythms [15]. For instance, the sampling of the central vertical pixels can be performed by applying $I_{\nu_R}(z, t) = \nu(x(\frac{W}{2}), y(z), t)$. Similarly, the central horizontal pixels can be extracted by applying $I_{\nu_R}(z, t) = \nu(x(z), y(\frac{H}{2}), t)$.

Given that the lower responses are mainly concentrated on the abscissa and ordinate axes [54] of the Fourier spectrum (see Figure 3.2), initially we consider two regions of interest in the frames that form the spectrum video in the construction of two types of visual rhythms: (1) the horizontal visual rhythm formed by central horizontal lines; and (2) the vertical visual rhythm formed by central vertical lines. In both cases, we can summarize relevant content of the spectrum video in a single image. Figure 3.3 depicts the visual rhythms generated by two regions of interest considering a valid (Figures 3.3(a) and 3.3(c)) and an attack video (Figures 3.3(b) and 3.3(d)).

Even though the visual rhythms are different for valid and attack videos, their construction disregards the highest responses that are not in the abscissa and ordinate axes and, in some cases, such information is important to make a better distinction between valid access and attempted attack videos, as shown in Figure 3.4. With this in mind, we extract a third type of visual rhythm by traversing along the frames of Fourier Spectrum Videos (ν_{FS}) in a zig-zag scheme. Figure 3.5 shows the zig-zag visual rhythm generated for a valid access video and an attempted attack video.

3.4 Feature Extraction

Once the visual rhythms are computed, we can use machine learning techniques to train a classifier to decide whether a biometric sample is fake or not. However, if the intensity of the pixels composing the visual rhythms are directly considered, the dimensionality of the feature space will be extremely high and most of the traditional classification methods will not work properly. Therefore, we need to extract a compact set of feature descriptors that best discriminate the visual rhythms generated from the fake and valid videos. In this work, we evaluate the use of three feature descriptors: GLCM [22], LPB [39] and Histogram of Oriented Gradients (HOG) [17]. The choice for using GLCM and LBP

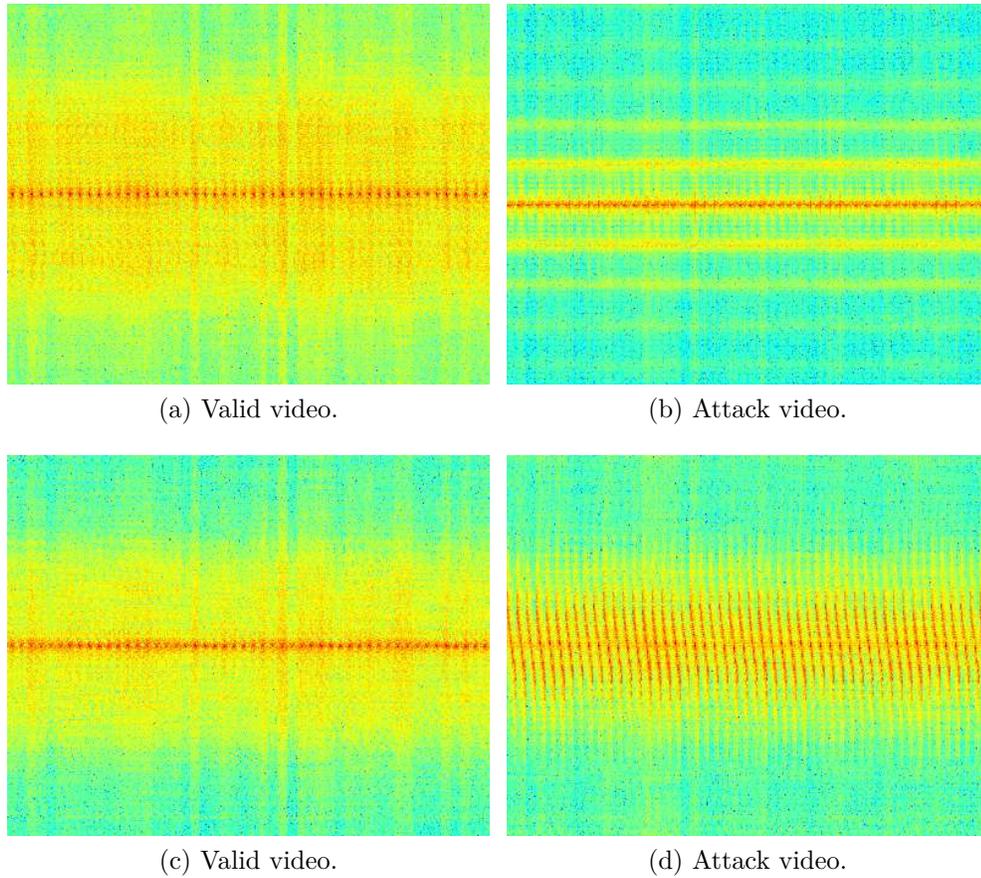


Figure 3.3: Examples of visual rhythms constructed from (a)-(b) central horizontal lines and from (c)-(d) central vertical lines. Note that the visual rhythm obtained from horizontal lines has been rotated 90 degrees for visualization purposes.

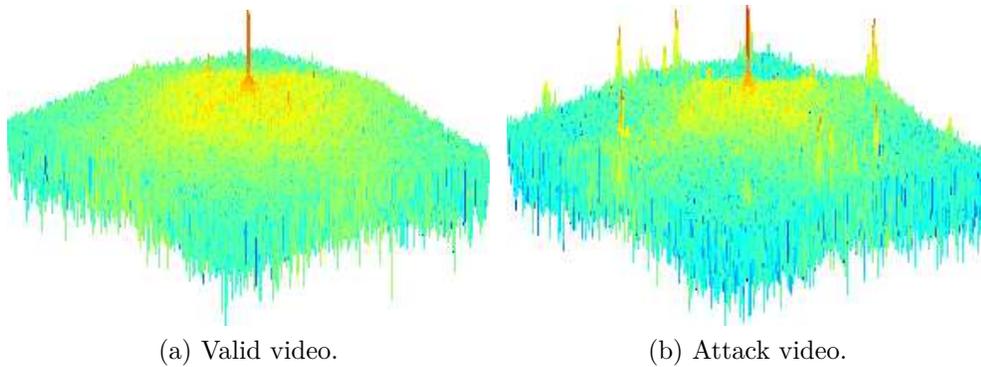


Figure 3.4: Examples of spectra whose highest responses are not only at the abscissa and ordinates axes.

descriptors is given by the fact that the visual rhythms can be interpreted as texture

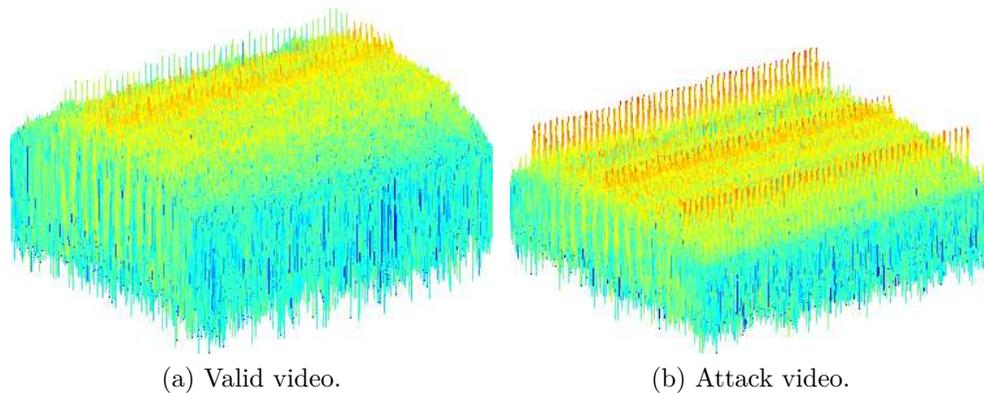


Figure 3.5: Examples of visual rhythms constructed by a traversal in zig-zag.

maps (see Figure 3.3). Moreover, if we consider the intensity values of the pixels of the visual rhythms as height and edge artifacts represented along the maps, we see (Figure 3.5) that such images have different edge forms, property that can be reasonably explored by the HOG descriptor.

3.4.1 Gray Level Co-occurrence Matrices (GLCM)

The GLCM is a procedure suggested for obtaining the textural features of an image. It is based on the assumption that the textural information on an image is contained in the overall or “average” spatial relationship that the gray tones in the image have to one another.

More specifically, it is assumed that the texture information of the image is adequately specified by the matrix of relative frequencies $P_{i,j}$ with that two neighboring pixels separated by a distance d , one pixel with gray tone i and the other with gray tone j [22]. Such matrix is a function of the angular relationship between the neighboring pixels as well as a function of the distance between them. The possible angular relations are shown in Figure 3.6(a).

After calculating the co-occurrence matrix for four different orientations, we extracted 12 measures to summarize the textural information of each matrix: angular second-moment, contrast, correlation, variance, inverse difference moment, sum average, sum variance, sum entropy, entropy, difference variance, difference entropy, and directionality [22]. Finally, all of the information are then combined to form a single feature vector as illustrated in Figure 3.6(b).

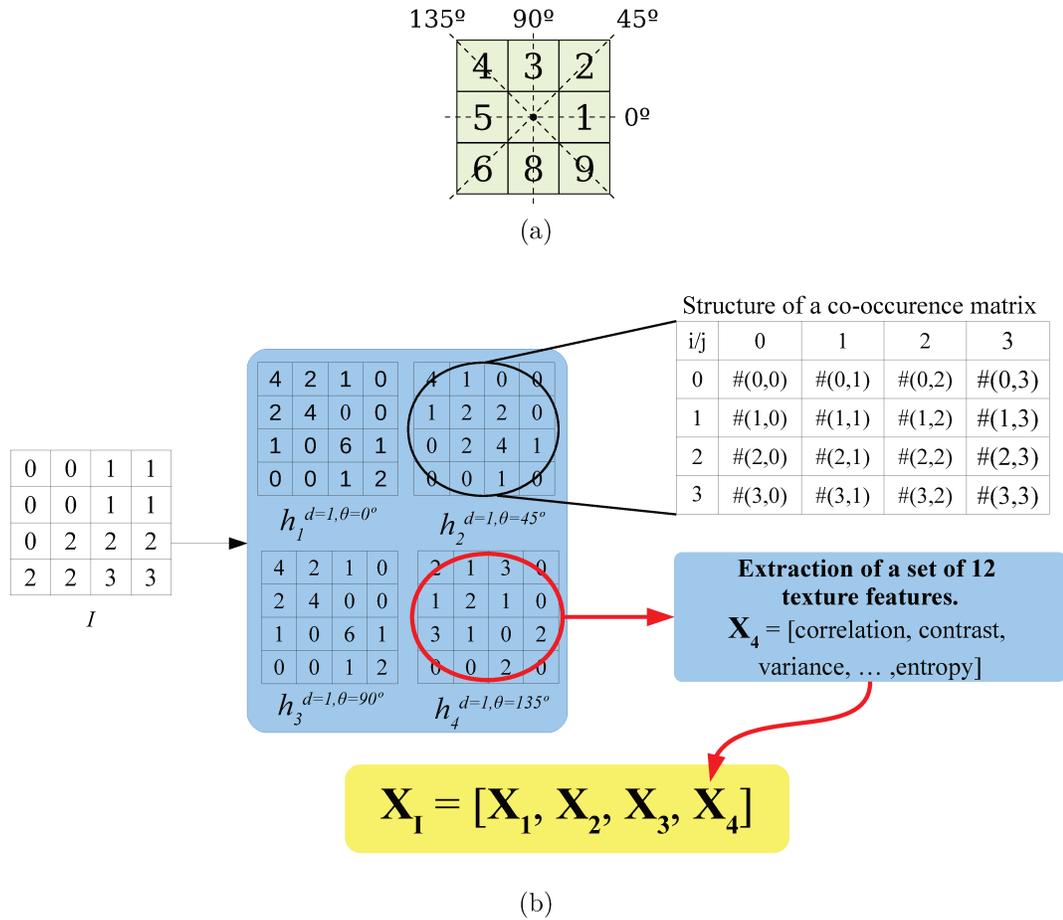


Figure 3.6: (a) Possible angular relationships θ between the center pixel ‘•’ and its neighbors that are at distance $d = 1$ and (b) An example of extraction of textural patterns of image I with the GLCM descriptor.

3.4.2 Local Binary Patterns (LBP)

The LBP operator provides a robust way to describe local binary patterns. This method allows to detect uniform local binary patterns at circular neighborhoods of any spatial resolution as well as at any quantization of the angular space.

The LBP operator can be derived for a general case based on a circularly symmetric neighbor set of P members on a circle of R radius, denoting the operator as $LBP_{P,R}$. Parameter P is used to control the quantization of the angular space, whereas R is used to control the spatial resolution of the operator, as formalized in Equation 3.5. Figure 3.7 illustrates an example for calculating the binary code with 8 bits ($P = 8$) of a pixel

considering a 3×3 neighborhood ($R = 1$).

$$LBP_{P,R} = \sum_{p=0}^{P-1} s(i_p - i_c) 2^p \quad (3.5)$$

$$s(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases}$$

where in this case p runs over the eight neighbors of the central pixel c , i_c and i_p are the gray-level values at c and p .

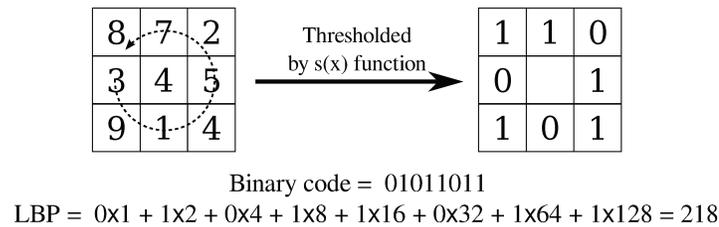


Figure 3.7: A window of size 3×3 is thresholded by the value of the central pixel. The pixel values are then multiplied by binomial weights and summed to obtain an LBP number to this window. Thus, LBP can produce up to $2^8 = 256$ different texture patterns, and a histogram with 256 bins is then calculated and used as a texture descriptor.

3.4.3 Histogram of Oriented Gradient (HOG)

HOG is a descriptor extensively used in computer vision to detect objects. The basic idea of this descriptor relies on the fact that the local appearance of the objects and shape can be well characterized by the distribution of local intensity gradients or edge directions, even without precise knowledge of the corresponding gradient or edge positions [17].

The computation of the HOG descriptor of a image can be describe as follow: First, a normalization of gamma and color in the input image is performed. In sequence, the normalized image is divided into small spatial regions, referenced as cells, and for each cell is calculated a histogram of gradient directions. A set of cells is grouped into a block and the concatenation of the histograms calculated from each cell followed by a normalization results in the HOG descriptor. Figure 3.8 illustrates the computation of the HOG descriptor.

3.5 Supervised Learning Algorithms

We approach the attempt attack detection problem as a binary classification problem. Therefore the valid videos were labeled as the positive class and attempted attack videos

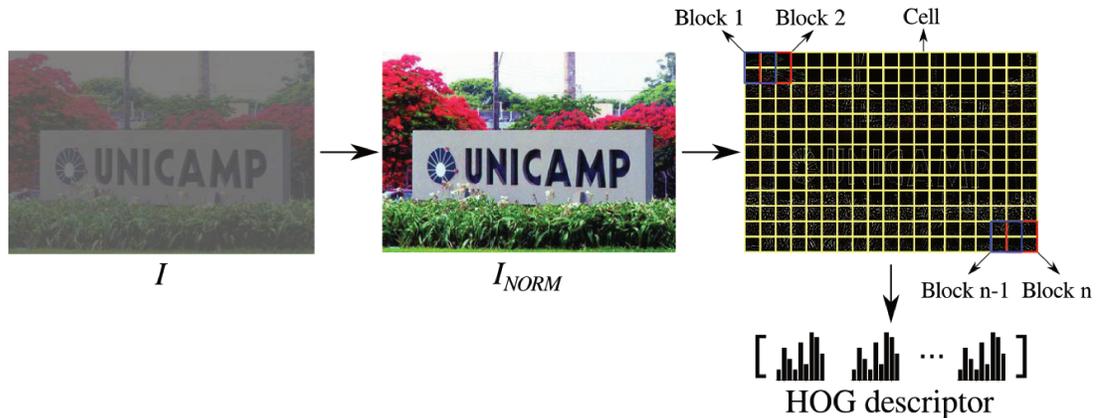


Figure 3.8: Example of extraction of the HOG descriptor of an input image I . After the color and gamma normalization in the image I , the resultant image I_{NORM} is divided into cells of size 8×8 pixels. In sequence, for each cell is calculated a histogram of gradients directions with nine bins. Then a set of four cells is grouped into a block and a normalization of the histograms calculated from each cell that compose the block is performed.

were labeled as the negative class. In this context, we evaluate the proposed characterization process using two classification techniques: SVMs and Partial Least Squares (PLS) that are used in the construction of a binary classifier to decide whether or not a sample is fake. More details about both algorithms SVM and PLS can be found in appendices A.1 and A.2, respectively.

The SVM algorithm [16] is a classification algorithm that has been used in many problems due to the great power of generalization achieved by the classifiers constructed with this algorithm. Basically, the SVM uses either a linear or a non-linear mapping, depending on the type of space used to transform the original data onto a higher dimensional one. Within this new space, the SVM finds an optimal hyperplane that separates the input data into classes.

In order to find the optimal hyperplane, Cortes et al. [16] introduces a concept of “margin” of a separating hyperplane that is the sum between the shortest distance from the separating hyperplane to the closest positive sample and the shortest distance from the separating hyperplane to the closest negative sample. The SVM looks for the separating hyperplane with largest margin. For that, the original problem is reformulated using the Lagrangian formulation and then the solution is found for an optimization algorithm.

PLS regression method [1, 23] is based on the linear transformation of a large number of descriptors to a new space based on a small number of orthogonal projection vectors. In other words, the projection vectors are mutually independent linear combinations of the original descriptors. These vectors are chosen to provide maximum correlation with

the dependent variables, which are the labels of the data belonging to the training set.

The PLS algorithm models the relation between the training and test sets, by the decomposition of both data sets as a product of a set of orthogonal factors and a set of specific loadings. For that, we use the Nonlinear Iterative Partial Least Squares (NIPALS) algorithm [58], and in this process, two important matrices are obtained, the matrix of latent vectors and the matrix of loading, which are used in the prediction of new observations.

Chapter 4

Database Creation

This section presents the Unicamp Video-Attack Database (UVAD) specifically built for evaluation of the video-based spoofing attack detection methods. The UVAD contains valid access and attempted attack videos of 304 different identities. All videos were created at Full HD quality, with 30 frames per second and are nine seconds long.

The generation of valid access videos was performed by filming each participant in two sections considering different backgrounds, lighting conditions, and places (indoor and outdoor). In total, 608 videos that represent valid accesses were generated with a 9.1 megapixels Sony CyberShot DSC-HX1 digital camera. We used a tripod to avoid disturbance in the videos during the recordings. The generated videos were cropped to maintain a resolution of $1,366 \times 768$ and allow the faces to be positioned at the center of the video frame. No resampling was performed whatsoever.

The attempted attack videos were generated by using three different digital cameras and eight different display devices, seven different monitors with a $1,366 \times 768$ pixel resolution and one HP tablet with $1,280 \times 768$ pixel resolution. In total, 608 videos were displayed on eight display devices and recaptured by the 9.1 megapixels Sony CyberShot DSC-HX1, the 10 megapixels Canon PowerShot SX1 IS, and the 10.3 megapixels Nikon Coolpix P100 digital cameras. Each monitor was positioned in front of each camera at a distance of 90 ± 5 cm supported in a tripod, so that to ensure $1,366 \times 768$ resolution for each video after cropping.

As the valid access videos were cropped to maintain a $1,366 \times 768$ resolution, we guarantee that there was no scaling transformations during their exhibition, except for the tablet where a scaling transformation was inevitable due to the screen's lower resolution. In total, we have generated 14,262 attempted attack videos and 608 valid access videos. Figures 4.1a and 4.1b illustrate real and fake video frames of UVAD dataset, respectively.



(a) Examples of valid access video frames for outdoor (images on the top) and indoor (images on the bottom) scenes.



(b) Examples of attempted attack video frames for outdoor (images on the top) and indoor (three images on the bottom) scenes using Sony (first and second columns), Canon (third and fourth columns) and Nikon (last column) cameras.

Figure 4.1: Examples of valid access video frames and attempted attack video frames that comprise the UVAD.

Table 4.1 shows a comparison between the proposed UVAD database and some other reference benchmarks in the literature. The diversity of display devices and acquisition sensors used in the generation of UVAD is an important characteristic that is not found in the other databases, which was essential to a better comprehension of the problem and for a precise evaluation of the methods.

Table 4.1: Comparison of the UVAD proposed database and other available reference benchmarks in the literature.

Database	Subjects	Valid accesses	Attacks by photo	Attacks by video	Devices used to create the attack videos
NUAA [55]	15	5, 105	7, 509	—	—
Print-Attack [4]	50	200	200	—	—
CASIA [63]	50	150	300	150	3 cameras and 1 display device
Replay-Attack [13]	50	200	200	800	2 cameras and 2 display devices
UVAD (proposed)	304	608	—	14, 262	3 cameras and 8 display devices

Chapter 5

Experimental Results

In this section, we show the details of the experiments and performance evaluations of the developed method. For that, we first consider the database UVAD which was introduced in Section 4 (Experiments I-V). The diversity of devices used allows us to answer important questions regarding some of the strengths and limitations of the proposed method. In addition, we also evaluate the proposed method using the Replay-Attack Database (c.f., Sec. 2.1) (Experiment VI).

5.1 Protocols for the UVAD Database

In this dissertation, we define appropriate protocols for each experiment.

Protocol A. The purpose of this protocol is to check the influence of the display devices over the detection method. Initially, the UVAD’s 304 identities were divided into two subsets, A and B , both with 152 disjoint identities with two view sessions for each person. The valid access videos of users in A were then divided to form two sets of valid access videos, both comprising 152 videos, given that each user was recorded at two sessions. The videos of the identities in B were used to simulate attempted attacks, in each biometric sensor, generating two sets of attempted attack videos: (1) videos recaptured by a biometric sensor under attempted attacks performed with four different display devices; and (2) videos recaptured by a biometric sensor under attack carried out with the other four different display devices. The partition considering different display devices for both attack sets was carried out to avoid that a classifier takes biased conclusions regarding videos coming from devices already seen during training even though using different videos.

Protocol B. The aim of this protocol is to check the influence of the biometric sensor on the proposed method. Similarly to the previous protocol, the set of 304 identities was

divided into two subsets, A and B , both with 152 disjoint identities and with two view sessions for each person. The valid access videos of the users of A were then divided to form two sets of the valid access videos, both with 152 videos. Then, we use the set B to simulate attacks in three biometric systems with three different biometric sensors. Thus we generate six groups of attempted attack videos: (1) videos recaptured by Sony; (2) videos recaptured by Canon; (3) videos recaptured by Nikon; (4) videos recaptured by Sony and Canon; (5) videos recaptured by Sony and Nikon; and (6) videos recaptured by Canon and Nikon. Our goal with these partitions is to train a classifier with videos from two cameras and test it with the videos generated with the third, considering all combinations, to verify the influence of the “biometric sensor” on the spoofing detection.

Protocol C. The aim of this protocol is at verifying whether attacks with tablets are more difficult to be detected. In this protocol, we generated two sets of valid access videos similarly to the previous protocols, but the videos of the identities in the B were used to generate three groups of the attempted attack videos: (1) videos recaptured by all cameras under attacks performed with a tablet; (2) videos recaptured by all cameras under attacks performed with three different monitors; and (3) videos recaptured by all cameras under attacks performed with the remaining four different monitors. Our goal with these partitions is to train a classifier with the set generated by attacks with monitors and to test it with the attacks generated with tablet and other different monitors. Once during training, the classifier will not have access to any data coming from tablets, a good effectiveness of this classifier in discriminating data from the tablet and monitors would indicate whether our method is robust to attacks performed with tablets.

5.2 Parameters for the Filtering Process, Visual Rhythm Analysis and Classification

To extract signal noise of the videos, as shown in Equation 3.1, we consider the use of spatial linear and non-linear filters: a Gaussian filter with $\mu = 0$, $\sigma = 2$, and size 7×7 and a Median filter with size 7×7 , respectively. These parameters were obtained empirically in [44] on a different dataset.

After calculating the noise signature using Equations 3.2 and 3.3, we extract the visual rhythms of each video (horizontal and vertical) considering the first 50 frames and a block of either 30 columns (vertical) pixels or 30 lines (horizontal). Since the visual vertical and horizontal rhythms of each video carries different temporal information, we evaluate the two types of visual rhythms along with their combinations. The horizontal visual rhythms are in a dimensional space of $1366 \times 1500-d$ while the vertical visual rhythms are in $768 \times 1500-d$. To generate the zig-zag visual rhythms, we also consider the first

50 frames of the Fourier Spectrum Videos. We extract block lines of 30 pixels through the traversal of the frames, from left to right, top to bottom. Thus, we obtained visual rhythms that are in a dimensional space of $17482 \times 1500-d$.

The high dimensionality and large amount of visual rhythms prevent us from using pixel intensities directly as features. Therefore, we consider the visual rhythms as texture maps and calculate their texture patterns using different characterization methods. For instance, for the standard configuration, we considered the GLCM descriptor with directions $\theta \in \{0^\circ, 45^\circ, 90^\circ, 135^\circ\}$, distance $d = 1$ and 16 bins. Table 5.1 shows the dimensionality of each feature (individually and combined).

Table 5.1: Number of features (dimensions) using either the direct pixel intensities as features or the features extracted by image description methods.

Name	Descriptor Dimensionality		
	Vertical	Horizontal	Zig-zag
Pixel Intensity	1, 152, 000	2, 049, 000	26, 223, 000
LBP	256	256	256
GLCM	48	48	48
HOG	36	36	36

In order to evaluate the robustness of the extracted features, we can use them to train a classifier and generate a model capable of distinguishing valid and attack videos, and test the model effectiveness. In this dissertation, we use two classification techniques: SVM and PLS. For SVM, we use the LibSVM [12] implementation and we analyze the radial basis function kernel, whose parameters were found using LibSVM’s built-in grid search algorithm. For PLS, we use the the DetectorPLS method [50] and we analyze different numbers of factors. The factors are latent variables that give us the best predictive power and they are extracted from a set of independent variables and are used to predict a set of dependent variables.

5.3 Experiment I: Influence of the Display Devices

The aim of this experiment is to check whether the presented method can detect attacks with different display devices. This is an important question to be answered because if the method is not robust to different devices, learning techniques considering an open scenario should be considered [48], given that in this case the classifier should be able to recognize attacks with display devices for which it has no prior knowledge.

Considering *Protocol A*, this experiment was performed in two rounds: in the first round, we train a classifier with attacks performed with four display devices and tested it with other different display devices to evaluate the model found by the classifier. In the second round, we train another classifier with data used in the testing step of the first round and tested it with data used in the training step of the first round. The results reported in Tables 5.2, 5.3, 5.4 and 5.5 correspond to the average and standard deviation of the two settings.

Table 5.2: Results showing Area Under the receiver operating characteristic Curve (AUC) of the experiment analyzing the influence of the display devices using a PLS Classifier and Median Filter.

Visual Rhythms	PLS classifier and Median filter		
	Sony	Canon	Nikon
Vertical	$\bar{x} = 74.43\%$ $\sigma = 2.67\%$	$\bar{x} = 99.98\%$ $\sigma = 0.01\%$	$\bar{x} = 99.95\%$ $\sigma = 0.06\%$
χ^2 test (<i>p-value</i>)	0.0747	0.5947	0.3294
Horizontal	$\bar{x} = 87.13\%$ $\sigma = 3.10\%$	$\bar{x} = 100.00\%$ $\sigma = 0.00\%$	$\bar{x} = 100.00\%$ $\sigma = 0.00\%$
χ^2 test (<i>p-value</i>)	0.0021	0.5738	0.5738
Vert. + Horiz.	$\bar{x} = 86.91\%$ $\sigma = 3.92\%$	$\bar{x} = 100.00\%$ $\sigma = 0.00\%$	$\bar{x} = 100.00\%$ $\sigma = 0.00\%$
χ^2 test (<i>p-value</i>)	0.1058	0.5738	0.5738
Zig-zag	$\bar{x} = 99.48\%$ $\sigma = 0.67\%$	$\bar{x} = 100.00\%$ $\sigma = 0.00\%$	$\bar{x} = 100.00\%$ $\sigma = 0.00\%$
χ^2 test (<i>p-value</i>)	2.76×10^{-5}	0.5738	0.5738

Comparing the results shown in Tables 5.2 and 5.3 and the results in Tables 5.4 and 5.5, we can see that the Gaussian filter provides a better classification rate than the Median filter in most cases (there is one notable exception in which the Median filtering showed better results, in the first line of Table 5.2).

To verify whether the differences in the results are statistically significant, we carried out a hypothesis test for two unpaired or independent samples. Once the sample values are nominal, the most appropriate statistical test is χ^2 test for two samples, whose values are also shown in all tables. We note that, in few settings, the *p-value* was lower than $\alpha = 0.05$, that is, the differences were statistically significant. Due to the high accuracy achieved, we can conclude that the display device plays an important role in the spoofing detection task.

Table 5.3: Results (AUC) of the experiment analyzing the influence of the display devices using a PLS Classifier and Gaussian Filter.

Visual Rhythms	PLS classifier and Gaussian filter		
	Sony	Canon	Nikon
Vertical	$\bar{x} = 70.30\%$ $\sigma = 4.29\%$	$\bar{x} = 99.99\%$ $\sigma = 0.00\%$	$\bar{x} = 100.00\%$ $\sigma = 0.01\%$
χ^2 test (<i>p-value</i>)	2.20×10^{-16}	0.0183	0.2400
Horizontal	$\bar{x} = 92.52\%$ $\sigma = 0.67\%$	$\bar{x} = 100.00\%$ $\sigma = 0.00\%$	$\bar{x} = 100.00\%$ $\sigma = 0.00\%$
χ^2 test (<i>p-value</i>)	2.08×10^{-5}	0.5738	0.5738
Vert. + Horiz.	$\bar{x} = 91.81\%$ $\sigma = 0.22\%$	$\bar{x} = 100.00\%$ $\sigma = 0.00\%$	$\bar{x} = 100.00\%$ $\sigma = 0.00\%$
χ^2 test (<i>p-value</i>)	0.0337	0.5738	0.5738
Zig-zag	$\bar{x} = 92.93\%$ $\sigma = 1.84\%$	$\bar{x} = 100.00\%$ $\sigma = 0.00\%$	$\bar{x} = 100.00\%$ $\sigma = 0.00\%$
χ^2 test (<i>p-value</i>)	0.0086	0.5738	0.5738

Table 5.4: Results (AUC) of the experiment analyzing the influence of the display devices using a SVM Classifier and Median Filter.

Visual Rhythms	SVM classifier and Median filter		
	Sony	Canon	Nikon
Vertical	$\bar{x} = 77.18\%$ $\sigma = 2.81\%$	$\bar{x} = 100.00\%$ $\sigma = 0.00\%$	$\bar{x} = 99.95\%$ $\sigma = 0.06\%$
χ^2 test (<i>p-value</i>)	0.2567	0.6658	0.2171
Horizontal	$\bar{x} = 86.66\%$ $\sigma = 6.52\%$	$\bar{x} = 100.00\%$ $\sigma = 0.00\%$	$\bar{x} = 100.00\%$ $\sigma = 0.00\%$
χ^2 test (<i>p-value</i>)	2.74×10^{-7}	6.27×10^{-6}	0.5459
Vert. + Horiz.	$\bar{x} = 90.29\%$ $\sigma = 5.33\%$	$\bar{x} = 100.00\%$ $\sigma = 0.00\%$	$\bar{x} = 100.00\%$ $\sigma = 0.00\%$
χ^2 test (<i>p-value</i>)	2.20×10^{-16}	0.0063	0.1715
Zig-zag	$\bar{x} = 97.58\%$ $\sigma = 2.07\%$	$\bar{x} = 100.00\%$ $\sigma = 0.00\%$	$\bar{x} = 100.00\%$ $\sigma = 0.00\%$
χ^2 test (<i>p-value</i>)	0.0040	2.20×10^{-16}	2.20×10^{-16}

5.4 Experiment II: Influence of the Biometric Sensors

This experiment aims at checking whether the presented method works well in different facial biometric systems (biometric sensors). Experiments performed with only one kind of

Table 5.5: Results (AUC) of the experiment analyzing the influence of the display devices using a SVM Classifier and Gaussian Filter.

Visual Rhythms	SVM classifier and Gaussian filter		
	Sony	Canon	Nikon
Vertical	$\bar{x} = 77.94\%$ $\sigma = 7.08\%$	$\bar{x} = 99.99\%$ $\sigma = 0.01\%$	$\bar{x} = 100.00\%$ $\sigma = 0.01\%$
χ^2 test (<i>p-value</i>)	0.9161	0.6679	0.5789
Horizontal	$\bar{x} = 89.42\%$ $\sigma = 0.60\%$	$\bar{x} = 100.00\%$ $\sigma = 0.00\%$	$\bar{x} = 100.00\%$ $\sigma = 0.00\%$
χ^2 test (<i>p-value</i>)	0.0062	0.0014	0.7341
Vert. + Horiz.	$\bar{x} = 92.55\%$ $\sigma = 2.52\%$	$\bar{x} = 100.00\%$ $\sigma = 0.00\%$	$\bar{x} = 100.00\%$ $\sigma = 0.00\%$
χ^2 test (<i>p-value</i>)	2.04×10^{-9}	0.1548	0.2031
Zig-zag	$\bar{x} = 93.44\%$ $\sigma = 0.44\%$	$\bar{x} = 100.00\%$ $\sigma = 0.00\%$	$\bar{x} = 100.00\%$ $\sigma = 0.00\%$
χ^2 test (<i>p-value</i>)	4.96×10^{-13}	2.20×10^{-16}	2.20×10^{-16}

biometric sensor does not guarantee a broad diversity. The importance of this experiment is due to the fact that the separability of the data in the classification may occur for reasons inherent to the camera used, and not to the method effectiveness. Using *Protocol.B*, we evaluate the different classifiers and filters considered in this dissertation, whose results are shown in Tables 5.6, 5.7, 5.8 and 5.9.

Table 5.6: Results (AUC) of the experiment analyzing the influence of the biometric sensors using a PLS Classifier and Median Filter.

Visual Rhythms	PLS classifier and Median filter		
	Test with Sony camera and train with others	Test with Canon camera and train with others	Test with Nikon camera and train with others
Vertical	49.49%	97.08%	77.55%
Horizontal	82.18%	87.26%	98.57%
Vert. + Horiz.	77.69%	85.79%	98.29%
Zig-zag	67.07%	99.94%	99.14%

We note that the attempted attacks performed under the Sony camera were the most difficult to be detected by the method whose best AUC was 84.63%. This value was achieved by using the PLS classification technique, Gaussian filter and horizontal visual

Table 5.7: Results (AUC) of the experiment analyzing the influence of the biometric sensors using a PLS Classifier and Gaussian Filter.

Visual Rhythms	PLS classifier and Gaussian filter		
	Test with Sony camera and train with others	Test with Canon camera and train with others	Test with Nikon camera and train with others
Vertical	55.78%	96.83%	95.55%
Horizontal	84.63%	88.98%	99.96%
Vert. + Horiz.	82.47%	83.14%	99.97%
Zig-zag	78.68%	99.85%	98.66%

Table 5.8: Results (AUC) of the experiment analyzing the influence of the biometric sensors using a SVM Classifier and Median Filter.

Visual Rhythms	SVM classifier and Median filter		
	Test with Sony camera and train with others	Test with Canon camera and train with others	Test with Nikon camera and train with others
Vertical	42.46%	99.85%	78.32%
Horizontal	83.74%	100.00%	100.00%
Vert. + Horiz.	82.71%	99.61%	99.55%
Zig-zag	66.92%	99.97%	98.73%

Table 5.9: Results (AUC) of the experiment analyzing the influence of the biometric sensors using a SVM Classifier and Gaussian Filter.

Visual Rhythms	SVM classifier and Gaussian filter		
	Test with Sony camera and train with others	Test with Canon camera and train with others	Test with Nikon camera and train with others
Vertical	50.76%	90.76%	79.69%
Horizontal	82.77%	99.54%	100.00%
Vert. + Horiz.	82.17%	77.97%	81.12%
Zig-zag	71.56%	99.75%	99.97%

rhythm. In contrast, attempted attacks performed under Canon and Nikon cameras were easily detected by the method, which obtained an AUC of 100% in both cameras using the SVM classification technique, Median filter and horizontal visual rhythm. Due to this large percentage difference, we can conclude that the biometric sensor plays an important

role in the spoofing detection task.

5.5 Experiment III: Attack with Tablets

With this experiment, we aim at checking whether our method is able to detect attacks performed with tablets. As an attempted attack video can be performed with several classes of devices, it is important to know if a specific method detects attacks carried out with different devices other than LCD monitors. We performed this experiment using the *Protocol C* and evaluate the two classifiers and filters considered in this dissertation. The results are shown in Tables 5.10, 5.11, 5.12 and 5.13.

Table 5.10: Results (AUC) of the experiment analyzing attacks with tablets using a PLS Classifier and Median Filter.

Attempted attack mode	Visual Rhythms	PLS classifier and Median filter		
		Sony	Canon	Nikon
tablet	Vertical	72.27%	99.99%	99.99%
	Horizontal	78.46%	100.00%	100.00%
	Vert. + Horiz.	80.43%	100.00%	100.00%
	Zig-zag	98.56%	100.00%	100.00%
monitor	Vertical	75.36%	100.00%	99.97%
	Horizontal	89.85%	100.00%	100.00%
	Vert. + Horiz.	89.39%	100.00%	100.00%
	Zig-zag	99.99%	100.00%	100.00%

Table 5.11: Results (AUC) of the experiment analyzing attacks with tablets using a PLS Classifier and Gaussian Filter.

Attempted attack mode	Visual Rhythms	PLS classifier and Gaussian filter		
		Sony	Canon	Nikon
tablet	Vertical	71.24%	99.97%	99.99%
	Horizontal	89.73%	100.00%	100.00%
	Vert. + Horiz.	89.46%	100.00%	100.00%
	Zig-zag	92.78%	100.00%	100.00%
monitor	Vertical	78.28%	100.00%	100.00%
	Horizontal	95.41%	100.00%	100.00%
	Vert. + Horiz.	95.24%	100.00%	100.00%
	Zig-zag	97.83%	100.00%	100.00%

Table 5.12: Results (AUC) of the experiment analyzing attacks with tablets using a SVM Classifier and Median Filter.

Attempted attack mode	Visual Rhythms	SVM classifier and Median filter		
		Sony	Canon	Nikon
tablet	Vertical	72.66%	100.00%	99.99%
	Horizontal	76.03%	100.00%	100.00%
	Vert. + Horiz.	85.68%	100.00%	100.00%
	Zig-zag	98.51%	99.98%	100.00%
monitor	Vertical	77.98%	100.00%	99.32%
	Horizontal	85.84%	100.00%	100.00%
	Vert. + Horiz.	90.68%	100.00%	100.00%
	Zig-zag	99.69%	100.00%	100.00%

Table 5.13: Results (AUC) of the experiment analyzing attacks with tablets using a SVM Classifier and Gaussian Filter.

Attempted attack mode	Visual Rhythms	SVM classifier and Gaussian filter		
		Sony	Canon	Nikon
tablet	Vertical	83.16%	99.61%	100.00%
	Horizontal	83.64%	100.00%	100.00%
	Vert. + Horiz.	91.08%	100.00%	100.00%
	Zig-zag	90.95%	100.00%	100.00%
monitor	Vertical	82.58%	99.87%	100.00%
	Horizontal	87.00%	100.00%	100.00%
	Vert. + Horiz.	91.39%	100.00%	100.00%
	Zig-zag	95.02%	100.00%	100.00%

According to the results, although in both attack modes the proposed method achieved reasonable results, attacks with monitors were more easily detected, except in some settings (highlighted values). However, considering the setting that we have obtained the best classification result (Table 5.11), the obtained AUC was 95,41% for the attacks with monitors against an AUC of 89,73% for attacks carried out with tablets (a difference of 5,68 percentage points). Therefore, we can conclude with this experiment that tablet-based attacks normally are harder to be detected than LCD-based attacks.

5.6 Experiment IV: Influence of the Feature Descriptors

In this section, we evaluate other important feature characterization methods found in the literature, namely LBP and HOG descriptors. Although we have considered the visual rhythms as texture maps, it is interesting to analyze the use of shape descriptors such as HOG as well. With this experiment, it is possible to discover whether considering the visual rhythms as texture maps is the best choice. We carried out these experiments considering the *Protocol B* whose results are shown in Tables 5.14 and 5.15.

According to the results shown in the tables, the highlighted results obtained by HOG and LBP descriptors were better than the results obtained by GLCM. Considering the best results obtained with GLCM (using the horizontal visual rhythm), we can conclude that GLCM was able to extract the most discriminant information of the visual rhythms. Therefore, it is reasonable to consider the visual rhythms as texture maps in the proposed manner.

Table 5.14: Results showing AUC using the PLS classification technique.

Descriptor	Visual Rhythms	PLS + Median filter			PLS + Gaussian filter		
		Test with Sony camera	Test with Canon camera	Test with Nikon camera	Test with Sony camera	Test with Canon camera	Test with Nikon camera
GLCM	Vertical	49.49%	97.08%	77.55%	55.78%	96.83%	95.55%
	Horizontal	82.18%	87.26%	98.57%	84.63%	88.98%	99.96%
	Vert. + Horiz.	77.69%	85.79%	98.29%	82.47%	83.14%	99.97%
	Zig-zag	67.07%	99.94%	99.14%	78.68%	99.85%	98.66%
LBP	Vertical	44.91%	89.84%	89.54%	49.65%	98.89%	97.21%
	Horizontal	53.82%	90.94%	84.30%	50.76%	97.20%	94.17%
	Vert. + Horiz.	54.42%	90.38%	83.04%	51.28%	94.26%	90.86%
	Zig-zag	69.31%	95.83%	79.54%	74.31%	97.85%	80.29%
HOG	Vertical	51.08%	98.34%	97.54%	50.95%	99.09%	98.93%
	Horizontal	63.49%	93.48%	95.02%	59.25%	96.40%	96.70%
	Vert. + Horiz.	56.37%	97.36%	95.94%	55.00%	98.25%	98.13%
	Zig-zag	33.10%	91.91%	78.28%	33.22%	97.95%	86.99%

Table 5.15: Results showing AUC using the SVM classification technique.

Descriptor	Visual Rhythms	SVM + Median filter			SVM + Gaussian filter		
		Test with Sony camera	Test with Canon camera	Test with Nikon camera	Test with Sony camera	Test with Canon camera	Test with Nikon camera
GLCM	Vertical	42.46%	99.85%	78.32%	50.76%	90.76%	79.69%
	Horizontal	83.74%	100.00%	100.00%	82.77%	99.54%	100.00%
	Vert. + Horiz.	82.71%	99.61%	99.55%	82.17%	77.97%	81.12%
	Zig-zag	66.92%	99.97%	98.73%	78.68%	99.85%	98.66%
LBP	Vertical	48.26%	89.50%	82.60%	56.15%	90.66%	92.55%
	Horizontal	47.50%	80.05%	75.33%	44.33%	82.80%	82.17%
	Vert. + Horiz.	47.05%	92.14%	92.10%	44.72%	85.36%	83.85%
	Zig-zag	49.12%	94.03%	85.90%	55.36%	91.81%	76.96%
HOG	Vertical	43.95%	93.78%	91.89%	47.07%	90.29%	88.14%
	Horizontal	45.51%	91.30%	88.56%	49.66%	98.31%	96.39%
	Vert. + Horiz.	41.85%	95.20%	95.96%	48.83%	98.77%	98.93%
	Zig-zag	31.49%	98.39%	94.71%	28.92%	92.49%	97.05%

5.7 Experiment V: Comparison to a State-of-the-Art Method

In the final round of experiments concerning the UVAD database, we compare our method with the one proposed in [51]. We considered the *Protocol B* to compare both methods. It was not possible to run the algorithm by Schwartz et al. by using the same parameters described in [51] due to the high dimensionality of the data, even on a machine with 48GB of RAM. The dimensionality of the feature vector generated by the original algorithm is higher than five million dimensions for each video frame.

In order to reduce the dimensionality of the feature vectors, we applied the HOG descriptor with blocks of sizes 128×128 and 256×256 with strides of 128 and 256 pixels, respectively. The other parameters were set as described in [51]. With this, we were able to reduce the feature vector dimensionality to 11,000- d . Table 5.16 shows the results obtained by using the algorithm in [51] and our method, considering the configuration that yielded the lowest classification error. For result evaluation, we performed the McNemar statistical test, since the data are paired in this case. Furthermore, the computational time spent by the algorithm in [51] was ≈ 237 hours to process all the data, whereas the method proposed in this work spent ≈ 72 hours. All experiments were conducted on an Intel Xeon E5620, 2.4GHz quad core processor with 48GB of RAM under Linux operating system.

With this experiment, we can conclude that our method better characterized video-based attacks while being more efficient and suitable for different classification techniques, once it provides more compact feature vectors.

Table 5.16: Comparison between the method presented in [51] and the method proposed in this work considering the use of horizontal visual rhythm, Median filter and SVM classification technique. The Results showing AUC. According to McNemar test, the methods are statistically different.

Method	Test with Sony camera	Test with Canon camera	Test with Nikon camera
Schwartz et al. [51]	85.19%	97.08%	97.62%
Our method	83.74%	100.00%	100.00%

5.8 Experiment VI: Evaluation of the Method in the Replay-Attack Database

We evaluate our method on the Replay-Attack database (c.f., 2.1) which contains photo-based and video-based spoofing attacks. The goal of this experiment is to verify the effectiveness of our method on these several types of attacks. We use the experimental protocol described in [13], and results are shown in Table 5.17. Although our method is designed to video-based spoofing attack detection, we have obtained a promising AUC of $\approx 93\%$. For reference, in [13], the authors reported a Half Total Error Rate (HTER) of 34.01% while our method yields an HTER of 14.27% (less than half of the previous classification error). We use a Gaussian filter with $\mu = 0$, $\sigma = 0.5$ and size 3×3 , and a Median filter with size 3×3 , empirically obtained by using the Replay-Attack Database. With this experiment, we can conclude that the proposed method is able not only to detect video-based spoof attacks but also photo-based spoof attacks.

Table 5.17: Results showing AUC for the test set.

Visual Rhythms	PLS classifier		SVM classifier	
	Median	Gaussian	Median	Gaussian
Vertical	83.99%	89.01%	86.26%	91.56%
Horizontal	81.98%	85.66%	80.67%	73.36%
Vert. + Horiz.	90.69%	92.98%	92.01%	91.81%
Zig-zag	78.39%	85.35%	86.56%	77.72%

Chapter 6

Conclusions and Future Work

Biometric authentication systems have been shown to be vulnerable to spoofing attacks in the sense that impostors can gain access privileges to resources as valid users. Spoofing attacks to a face recognition system can be performed by presenting to it a photograph, a video, or a face mask of a legitimate user.

This dissertation proposed and evaluated a spatio-temporal method for video-based face spoofing detection through the analysis of noise signatures generated by the video acquisition process, which can be used to distinguish between valid and fake access videos. Noise properties are captured using Fourier spectrum for each frame of the video. A compact representation, called visual rhythm, is employed to detect temporal information in the Fourier spectrum. Three different video traversal strategies are considered to form the visual rhythms. Features are extracted from the visual rhythms through GLCM, LBP and HOG descriptors to allow a proper distinction between fake and real biometric data. The GLCM descriptor provided the most discriminant and compact information from the visual rhythms.

An extensive data set, containing real access and spoofing attack videos, was created to evaluate the proposed method, as well as the state-of-the-art approaches. Through the conducted experiments, it is possible to conclude that the display devices and biometric sensors play an important role in the spoofing detection task. In particular, attacks performed with tablets are more difficult to be detected than those performed with monitors, which is a concern due to the increasing availability of tablets. The proposed video-based face spoofing detection method provided competitive or even superior results in some tests when compared to state-of-the-art approaches.

Although this dissertation represents a step towards solving the spoofing problem, it makes it clear that the problem is not fully-solved yet and poses new questions on future methods about how to better handle and tackle attacks related to the ever-growing market of handheld and smartphone devices. In this sense, the dataset provided in this paper

will be available at the IEEE Information Forensics and Security Technical Committee website (<http://tinyurl.com/pas4t9r>) in order to push the spoofing detection research frontier way beyond.

Future research efforts branch out into devising other spatio-temporal descriptors that capture motion telltales associated with the recapture process as well as verifying other liveness detection problems other than face recognition such as video recapturing, piracy detection, among others. Investigations of facial biometric systems under attack with 3D masks are also of interest along with an in-depth evaluation of the vulnerability of others biometrics modalities under spoofing attacks.

Bibliography

- [1] Hervé Abdi. Partial Least Squares Regression and Projection on Latent Structure Regression (PLS Regression). *Wiley Interdisciplinary Reviews: Computational Statistics*, 2(1):97–106, 2010.
- [2] Zahid Akhtar, Giorgio Fumera, Gian Luca Marcialis, and Fabio Roli. Robustness evaluation of biometric systems under spoof attacks. In *Proceedings of the 16th international conference on Image analysis and processing: Part I, ICIAP’11*, pages 159–168, Berlin, Heidelberg, 2011. Springer-Verlag.
- [3] Zahid Akhtar, Giorgio Fumera, Gian Luca Marcialis, and Fabio Roli. Evaluation of serial and parallel multibiometric systems under spoofing attacks. In *IEEE Intl. Conference on Biometrics: Theory Applications and Systems*, pages 283–288, Sept. 2012.
- [4] A. Anjos and S. Marcel. Counter-measures to photo attacks in face recognition: A public database and a baseline. In *Intl. Joint Conference on Biometrics*, pages 1–7, Oct. 2011.
- [5] Wei Bao, Hong Li, Nan Li, and Wei Jiang. A liveness detection method for face recognition based on optical flow field. In *IEEE Intl. Conference on Image Analysis and Signal Processing*, pages 233–236, Apr. 2009.
- [6] Matthew Barker and William Rayens. Partial least squares for discrimination. *Journal of Chemometrics*, 17(3):166–173, 2003.
- [7] Andy Beach. *Real world video compression*. Peachpit Press, Berkeley, CA, USA, first edition, 2008.
- [8] B. Biggio, Z. Akhtar, G. Fumera, G. L. Marcialis, and F. Roli. Security evaluation of biometric authentication systems under real spoofing attacks. *IET Biometrics*, 1(1):11–24, Mar. 2012.

- [9] B. Biggio, Z. Akthar, G. Fumera, G. L. Marcialis, and F. Roli. Robustness of multi-modal biometric verification systems under realistic spoofing attacks. In *IEEE Intl. Conference on Biometrics: Theory Applications and Systems*, pages 1–6, Oct. 2011.
- [10] Ileana Buhan and Pieter Hartel. The state of the art in abuse of biometrics. Technical Report TR-CTIT-05-41, Centre for Telematics and Information Technology University of Twente, Enschede, Sept. 2005.
- [11] Christopher J. C. Burges. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2(2):121–167, June 1998.
- [12] Chih-Chung Chang and Chih-Jen Lin. LIBSVM: A Library for Support Vector Machines. *ACM Transactions on Intelligent Systems and Technology*, 2, 2011. Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.
- [13] I. Chingovska, A. Anjos, and S. Marcel. On the effectiveness of local binary patterns in face anti-spoofing. In *Proceedings of the International Conference of the Biometrics Special Interest Group*, pages 1–7, Sept. 2012.
- [14] SeongSoo Chun, Hyeokman Kim, Kim Jung-Rim, Sangwook Oh, and Sanghoon Sull. Fast text caption localization on video using visual rhythm. In Shi-Kuo Chang, Zen Chen, and Suh-Yin Lee, editors, *Recent Advances in Visual Information Systems*, volume 2314 of *Lecture Notes in Computer Science*, pages 259–268. Springer Berlin Heidelberg, 2002.
- [15] M.-G. Chung, J. Lee, H. Kim, S. M.-H. Song, and W.-M. Kim. Automatic Video Segmentation based on Spatio-Temporal Features. *Korea Telecom*, 1(4):4–14, 1999.
- [16] Corinna Cortes and Vladimir Vapnik. Support-Vector Networks. *Mach. Learn.*, 20(3):273–297, Sept. 1995.
- [17] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *IEEE Intl. Conference on Computer Vision and Pattern Recognition*, volume 1, pages 886–893, June 2005.
- [18] Tiago de Freitas Pereira, André Anjos, José Mario De Martino, and Sébastien Marcel. Can face anti-spoofing countermeasures work in a real world scenario? In *IAPR International Conference on Biometrics*, June 2013.
- [19] Nesli Erdogmus and Sébastien Marcel. Spoofing in 2D face recognition with 3D masks and anti-spoofing with kinect. In *IEEE Intl. Conference on Biometrics: Theory Applications and Systems*, Sept. 2013.

- [20] A.S. Georghiadis, P.N. Belhumeur, and D. Kriegman. From few to many: Illumination cone models for face recognition under variable lighting and pose. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 23(6):643–660, June.
- [21] S. J. F. Guimaraes, M. Couprie, N. J. Leite, and A. A. Araujo. A Method for Cut Detection Based on Visual Rhythm. In *Brazilian Symposium on Computer Graphics and Image Processing*, pages 297–304, 2001.
- [22] Robert M. Haralick, K. Shanmugam, and Its’Hak Dinstein. Textural features for image classification. *IEEE Trans. on Systems, Man, and Cybernetics*, SMC-3(6):610–621, Nov. 1973.
- [23] Agnar Hoskuldsson. PLS Regression Methods. *Journal of Chemometrics*, 2(3):211–228, 1988.
- [24] A. K. Jain and A. Ross. *Handbook of Biometrics*, chapter Introduction to Biometrics, pages 1–22. Springer, 2008.
- [25] Gahyun Kim, Sungmin Eum, Jae Kyu Suhr, Dong Ik Kim, Kang Ryoung Park, and Jaihie Kim. Face liveness detection based on texture and frequency analyses. In *IAPR International Conference on Biometrics*, pages 67–72, Apr. 2012.
- [26] Klaus Kollreider, Hartwig Fronthaler, and Josef Bigun. Non-intrusive liveness detection by face images. *Image and Vision Computing*, 27(3):233–244, 2009.
- [27] Jukka Komulainen, Abdenour Hadid, and Matti Pietikäinen. Face spoofing detection using dynamic texture. In Jong-Il Park and Junmo Kim, editors, *Computer Vision - ACCV 2012 Workshops*, volume 7728 of *Lecture Notes in Computer Science*, pages 146–157. Springer Berlin Heidelberg, 2013.
- [28] Jukka Komulainen, Abdenour Hadid, Matti Pietikainen, André Anjos, and Sébastien Marcel. Complementary countermeasures for detecting scenic face spoofing attacks. In *IAPR International Conference on Biometrics*, June 2013.
- [29] Neslihan Kose and Jean-Luc Dugelay. Countermeasure for the protection of face recognition systems against mask attacks. In *IEEE Intl. Conference on Automatic Face and Gesture Recognition*, pages 22–26, Shanghai, CHINA, Apr. 2013.
- [30] Jiang-Wei Li. Eye blink detection based on multiple gabor response waves. In *IEEE Intl. Conference on Machine Learning and Cybernetics*, volume 5, pages 2852–2856, July 2008.

- [31] Jiangwei Li, Yunhong Wang, Tieniu Tan, and Anil K. Jain. Live face detection based on the analysis of fourier spectra. In *Biometric Technology for Human Identification*, pages 296–303, 2004.
- [32] Ana Carolina Lorena and André CPLF de Carvalho. Uma introdução às support vector machines. *Revista de Informática Teórica e Aplicada*, 14(2):43–67, 2007.
- [33] J. Lukäs, J. Fridrich, and M. Goljan. Digital camera identification from sensor pattern noise. *IEEE Trans. on Information Forensics and Security*, 1(2):205–214, June 2006.
- [34] J. Määttä, A. Hadid, and M. Pietikäinen. Face spoofing detection from single images using micro-texture analysis. In *Intl. Joint Conference on Biometrics*, pages 1–7, Oct. 2011.
- [35] J. Määttä, A. Hadid, and M. Pietikäinen. Face spoofing detection from single images using texture and local shape analysis. *IET Biometrics*, 1(1):3–10, Mar. 2012.
- [36] Emanuela Marasco, Peter Johnson, Carlo Sansone, and Stephanie Schuckers. Increase the security of multibiometric systems by incorporating a spoofing detection algorithm in the fusion mechanism. In *Proceedings of the 10th International Conference on Multiple Classifier Systems, MCS’11*, pages 309–318, Berlin, Heidelberg, 2011. Springer-Verlag.
- [37] M. De Marsico, M. Nappi, D. Riccio, and J. Dugelay. Moving face spoofing detection via 3D projective invariants. In *IAPR International Conference on Biometrics*, pages 73–78, Apr. 2012.
- [38] K. Muller, S. Mika, G. Ratsch, K. Tsuda, and Bernhard Scholkopf. An introduction to kernel-based learning algorithms. *IEEE Transactions on Neural Networks*, 12(2):181–201, 2001.
- [39] T. Ojala, M. Pietikainen, and T. Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 24(7):971–987, July 2002.
- [40] G. Pan, Z. Wu, and L. Sun. *Recent Advances in Face Recognition*, chapter Liveness detection for face recognition, pages 235–252. InTech, 2008.
- [41] Gang Pan, Lin Sun, Zhaohui Wu, and Shihong Lao. Eyeblink-based anti-spoofing in face recognition from a generic webcam. In *IEEE Intl. Conference on Computer Vision*, pages 1–8, Oct. 2007.

- [42] Gang Pan, Lin Sun, Zhaohui Wu, and Yueming Wang. Monocular camera-based face liveness detection by combining eyeblink and scene context. *Telecommunication Systems*, 47:215–225, 2011.
- [43] B. Peixoto, C. Michelassi, and A. Rocha. Face liveness detection under bad illumination conditions. In *IEEE Intl. Conference on Image Processing*, pages 3557–3560, Sept. 2011.
- [44] Allan Pinto, Helio Pedrini, William Robson Schwartz, and Anderson Rocha. Video-based face spoofing detection through visual rhythm analysis. In *Brazilian Symposium on Computer Graphics and Image Processing*, pages 221–228, Ouro Preto, MG, Brazil, Aug. 2012.
- [45] Anderson Rocha, Walter Scheirer, Terrance Boult, and Siome Goldenstein. Vision of the unseen: Current trends and challenges in digital image and video forensics. *ACM Comput. Surv.*, 43(4):26:1–26:42, Oct. 2011.
- [46] R. N. Rodrigues, N. Kamat, and V. Govindaraju. Evaluation of biometric spoofing in a multimodal system. In *IEEE Intl. Conference on Biometrics: Theory Applications and Systems*, pages 1–5, Sept. 2010.
- [47] Roman Rosipal and Nicole Krämer. Overview and recent advances in partial least squares. In Craig Saunders, Marko Grobelnik, Steve Gunn, and John Shawe-Taylor, editors, *Subspace, Latent Structure and Feature Selection*, volume 3940 of *Lecture Notes in Computer Science*, pages 34–51. Springer Berlin Heidelberg, 2006.
- [48] W. Scheirer, A. Rocha, A. Sapkota, and T. Boult. Towards open set recognition. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, PP(99):1, 2013.
- [49] Bernhard Scholkopf and Alexander J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, Cambridge, MA, USA, 2001.
- [50] William Robson Schwartz, A. Kembhavi, D. Harwood, and L.S. Davis. Human Detection Using Partial Least Squares Analysis. In *IEEE Intl. Conference on Computer Vision*, 2009.
- [51] William Robson Schwartz, Anderson Rocha, and Helio Pedrini. Face spoofing detection through partial least squares and low-level descriptors. In *Intl. Joint Conference on Biometrics*, pages 1–8, Oct. 2011.

- [52] W.R. Schwartz, R.D. da Silva, L.S. Davis, and H. Pedrini. A novel feature descriptor based on the shearlet transform. In *IEEE Intl. Conference on Image Processing*, pages 1033–1036, 2011.
- [53] W.R. Schwartz, A. Kembhavi, D. Harwood, and L.S. Davis. Human detection using partial least squares analysis. In *IEEE Intl. Conference on Computer Vision*, pages 24–31, 2009.
- [54] Steven W. Smith. *The Scientist and Engineer’s Guide to Digital Signal Processing*. California Technical Publishing, San Diego, CA, USA, 1997.
- [55] Xiaoyang Tan, Yi Li, Jun Liu, and Lin Jiang. Face liveness detection from a single image with sparse low rank bilinear discriminative model. In *European Conference on Computer Vision*, pages 504–517, Berlin, Heidelberg, 2010. Springer-Verlag.
- [56] R. Tronci, D. Muntoni, G. Fadda, M. Pili, N. Sirena, G. Murgia, M. Ristori, and F. Roli. Fusion of multiple clues for photo-attack detection in face recognition systems. In *Intl. Joint Conference on Biometrics*, pages 1–6, Oct. 2011.
- [57] Vladimir N. Vapnik. *The nature of statistical learning theory*. Springer-Verlag New York, Inc., New York, NY, USA, 1995.
- [58] H. Wold. *Path models with latent variables: The NIPALS approach*. Academic Press, 1975.
- [59] H. Wold. Partial Least Squares. In S. Kotz and N.L. Johnson, editors, *Encyclopedia of Statistical Sciences*, volume 6, pages 581–591. Wiley, New York, 1985.
- [60] S. Wold, A. Ruhe, H. Wold, and W. J. Dunn III. The collinearity problem in linear regression, the partial least squares (PLS) approach to generalized inverses. *SIAM Journal of Statistics and Computations*, 5:735–743, 1984.
- [61] Cui Xu, Ying Zheng, and Zengfu Wang. Eye states detection by boosting local binary pattern histogram features. In *IEEE Intl. Conference on Image Processing*, pages 1480–1483, Oct. 2008.
- [62] Junjie Yan, Zhiwei Zhang, Zhen Lei, Dong Yi, and Stan Z. Li. Face liveness detection by exploring multiple scenic clues. In *IEE Intl. Conference on Control Automation Robotics and Vision*, Dec. 2012.
- [63] Zhiwei Zhang, Junjie Yan, Sifei Liu, Zhen Lei, Dong Yi, and S.Z. Li. A face antispoofing database with diverse attacks. In *IAPR International Conference on Biometrics*, pages 26–31, Apr. 2012.

Appendix A

Learning Algorithms

In this appendix, we present more details of the algorithms SVM and PLS. In the next two sections, the boldface italic lowercase letters represent vectors, boldface uppercase letters are matrices, and scalar variables are written with letters in italics.

A.1 SVM Algorithm

The SVM is a learning algorithm [16] used to find an optimal hyperplane that separates the input data into classes. The SVM is supported by the statistical learning theory [57] and has been used in several applications due to great generalization obtained for classifiers constructed with this algorithm.

In this section, we present the basic mathematical foundations of the SVM to the linear case. The explanations below about the SVM was based on the works [11,32,38,49]. More details about this algorithm for both the linear and nonlinear cases can be found in these tutorials.

We consider linear binary machines trained on a separable data set $\mathcal{T} = \{(\mathbf{x}_i, y_i); i = 1, \dots, n\}$, $(\mathbf{x}_i, y_i) \in \mathbb{R}^d \times Y$, $Y = \{-1, +1\}$. Suppose we have some hyperplane which separates the positive from the negative samples as Figure A.1 depicts. The points \mathbf{x} which lie on the hyperplane satisfy the equation $\mathbf{w} \cdot \mathbf{x} + b = 0$, in that \mathbf{w} is normal to the hyperplane, $| -b|/\|\mathbf{w}\|$ is the perpendicular distance from the hyperplane to the origin, with $b \in \mathbb{R}$ and $\|\mathbf{w}\|$ the Euclidean norm of \mathbf{w} .

Let d_+ be the shortest distance from the separating hyperplane to the closet positive sample and d_- be the shortest distance from the separating hyperplane to the closet negative sample, we can define the margin of a separating hyperplane as $d = d_+ + d_-$. Basically, the SVM algorithm looks for the separating hyperplane with largest margin,

that is done as follow. Suppose that all the training data satisfy the following constraints:

$$\mathbf{x}_i \cdot \mathbf{w} + b \geq +1 \quad \text{for } y_i = +1 \quad (\text{A.1})$$

$$\mathbf{x}_i \cdot \mathbf{w} + b \leq -1 \quad \text{for } y_i = -1 \quad (\text{A.2})$$

Combining the inequalities A.1 and A.2 we have:

$$y_i(\mathbf{x}_i \cdot \mathbf{w} + b) - 1 \geq 0 \quad \forall (\mathbf{x}_i, y_i) \in \mathcal{T} \quad (\text{A.3})$$

Now consider the points for which the equality in Equation A.1 is satisfied. The points which lie on the hyperplane $H_1: \mathbf{x}_i \cdot \mathbf{w} + b = 1$ with normal \mathbf{w} and perpendicular distance from the origin $|1 - b|/\|\mathbf{w}\|$. Similarly, the points for which the equality in Equation A.2 which lie on the hyperplane $H_2: \mathbf{x}_i \cdot \mathbf{w} + b = -1$, with normal again \mathbf{w} , and perpendicular distance from the origin $|-1 - b|/\|\mathbf{w}\|$. Hence $d_+ = d_- = 1/\|\mathbf{w}\|$ and the margin is simply $2/\|\mathbf{w}\|$. This value distance is obtained by projecting $(\mathbf{x}_1 - \mathbf{x}_2)$ in the direction of \mathbf{w} , perpendicular to the hyperplane $\mathbf{w} \cdot \mathbf{x}_i + b = 0$ as formalized in Equation A.4. Note that H_1 and H_2 are parallel and that no training points fall between them.

$$(\mathbf{x}_1 - \mathbf{x}_2) \left(\frac{\mathbf{w}}{\|\mathbf{w}\|} \cdot \frac{(\mathbf{x}_1 - \mathbf{x}_2)}{\|\mathbf{x}_1 - \mathbf{x}_2\|} \right) \quad (\text{A.4})$$

Hence $\mathbf{w} \cdot \mathbf{x}_1 + b = 1$ and $\mathbf{w} \cdot \mathbf{x}_2 + b = -1$, the difference between these equations gives us $\mathbf{w} \cdot (\mathbf{x}_1 - \mathbf{x}_2) = 2$. Replacing this value in Equation A.4, we have:

$$\frac{2}{\|\mathbf{w}\|} \cdot \frac{(\mathbf{x}_1 - \mathbf{x}_2)}{\|\mathbf{x}_1 - \mathbf{x}_2\|} = \frac{2}{\|\mathbf{w}\|} \quad (\text{A.5})$$

We can find the pair of hyperplanes which gives the maximum margin by minimizing $\|\mathbf{w}\|^2$. This can be formulated as a quadratic optimization problem

$$\min_{\mathbf{w}, b} L(\mathbf{w}) = \frac{1}{2} \|\mathbf{w}\|^2 \quad (\text{A.6})$$

subject to constraints A.3. Thus, we expect the solution for a typical two dimensional case to have the form shown in Figure A.1. Those training points for which the equality in Equation A.3 is satisfied, and whose removal would change the solution found, are called support vectors and they are highlighted with a extra circles in Figure A.1.

We will now switch to a Lagrangian formulation of the problem and there are two reasons for doing this. The first is that the constraints A.3 will be replaced by constraints on the Lagrange multipliers themselves, which is much easier to solve. The second is that in this reformulation of the problem, the training data will only appear in the form of dot products between vectors and this is a important property because allows to generalize the procedure to the nonlinear case.

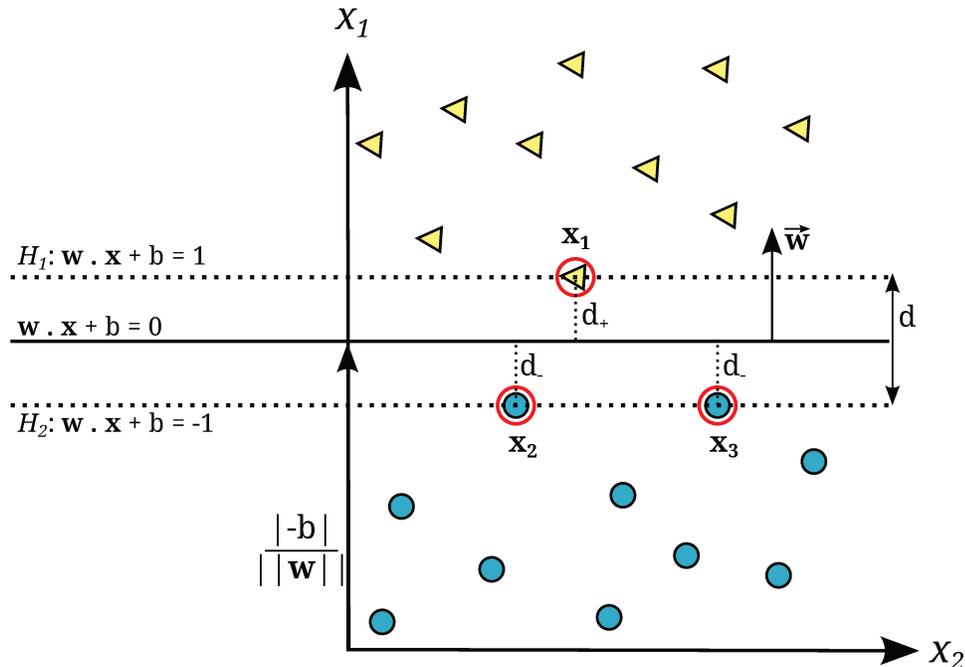


Figure A.1: Considering a simple binary classification problem that consist in separates balls from triangles. The optimal hyperplane is represented by solid line and there is a weight vector \mathbf{w} and a threshold b such that $y_i \cdot ((\mathbf{w} \cdot \mathbf{x}_i) > 0$. Rescaling \mathbf{w} and b such that the points closest to the hyperplane satisfy the equation $|(\mathbf{w} \cdot \mathbf{x}_i) + b| = 1$, we obtain a form (\mathbf{w}, b) of the hyperplane with $y_i((\mathbf{w} \cdot \mathbf{x}_i) + b) \geq 1$. Image adapted from [11].

Thus, we introduce positive Lagrange multipliers $\alpha_i \geq 0, i = 1, \dots, n$, one for each of the inequality constraints A.3. Recall that the rule is that for constraints of the form $c_i \geq 0$, the constraint equations are multiplied by positive Lagrange multipliers and subtracted from the objective function, to form the Lagrangian. For equality constraints, the Lagrange multipliers are unconstrained. This give us the following Lagrangian:

$$L_P(\mathbf{w}, b, \boldsymbol{\alpha}) = \frac{1}{2} \|\mathbf{w}\|^2 - \sum_{i=1}^n \alpha_i y_i (\mathbf{x}_i \cdot \mathbf{w} + b) + \sum_{i=1}^n \alpha_i \quad (\text{A.7})$$

The task now is to minimize Eq. A.7 with respect to \mathbf{w} , b and to maximize it with respect to $\boldsymbol{\alpha}$. We have the following saddle point equations, at the optimal point:

$$\frac{\partial L_P}{\partial b}(\mathbf{w}, b, \boldsymbol{\alpha}) = 0 \quad \text{and} \quad \frac{\partial L_P}{\partial \mathbf{w}}(\mathbf{w}, b, \boldsymbol{\alpha}) = 0 \quad (\text{A.8})$$

leading to

$$\mathbf{w} = \sum_i \alpha_i y_i \mathbf{x}_i \quad (\text{A.9})$$

$$\sum_i \alpha_i y_i = 0. \quad (\text{A.10})$$

By replacing A.9 and A.10 into the Lagrangian A.7, one eliminates the primal variables \mathbf{w} and b , arriving at the so-called dual optimization problem, which is the problem that one usually solves in practice:

$$\max_{\boldsymbol{\alpha}} L_D(\mathbf{w}) = \sum_i^n \alpha_i - \sum_{i,j=1}^n \alpha_i \alpha_j y_i y_j \mathbf{x}_i \cdot \mathbf{x}_j \quad (\text{A.11})$$

subject to constraints

$$\alpha_i \geq 0, i = 1, \dots, n \quad (\text{A.12})$$

$$\sum_{i=1}^n \alpha_i y_i = 0. \quad (\text{A.13})$$

Note that we have now given the Lagrangian different labels (P for primal, D for dual) to emphasize that the two formulations are different: L_P and L_D is constructed from the same objective function but with different constraints; and the solution is found by minimizing L_P or by maximizing L_D . The dual form have constraints simpler than primal and allows the representation of the optimization problem in terms of inner production between data.

Support vector training therefore amounts to maximizing L_D with respect to the α_i , subject to constraints A.12 and A.13. There is a Lagrange multiplier α_i for every training point, and in the solution, those points for which $\alpha_i > 0$ are called Support Vectors (SVs), and lie on one of the hyperplanes H_1, H_2 and all other training points have $\alpha_i = 0$. For the SVM, the support vectors are the critical elements of the training set. They lie closest to the decision boundary; if all other training points were removed (or moved around, but so as not to cross H_1 or H_2), and training was repeated, the same separating hyperplane would be found.

Let $\boldsymbol{\alpha}^*$ the solution of the dual problem and \mathbf{w}^* and b^* the solutions of the primal form. Obtained the values of $\boldsymbol{\alpha}^*$, \mathbf{w}^* can be determined by equation A.9. The parameter b^* is defined by $\boldsymbol{\alpha}^*$ and Kühn-Tucker constraints, from the theory of constrained optimization and that must be satisfied at the optimal point. For the dual problem formulated has [32]:

$$\alpha_i^* (y_i (\mathbf{w}^* \cdot \mathbf{x}_i + b^*) - 1) = 0, \forall i = 1, \dots, n \quad (\text{A.14})$$

The b^* parameter value is calculated from the SVs and the conditions shown in Equation A.14. For this, we compute the average shown in Equation A.15 for every \mathbf{x}_j such

that $\alpha_j > 0$. In this equation, \mathcal{S} is the set of SVs and $|\mathcal{S}|$ denotes its cardinality.

$$b^* = \frac{1}{|\mathcal{S}|} \sum_{\mathbf{x}_j \in \mathcal{S}} \frac{1}{y_j} - \mathbf{w}^* \cdot \mathbf{x}_j \quad (\text{A.15})$$

Replacing \mathbf{w}^* by Equation A.9 we have

$$b^* = \frac{1}{|\mathcal{S}|} \sum_{\mathbf{x}_j \in \mathcal{S}} \left(\frac{1}{y_j} - \sum_{\mathbf{x}_i \in \mathcal{S}} \alpha_i^* y_i \mathbf{x}_i \cdot \mathbf{x}_j \right) \quad (\text{A.16})$$

To test new sample we simply determine on which side of the decision boundary a given test pattern \mathbf{x} lies and assign the corresponding class label by Equation A.17

$$g(\mathbf{x}) = \text{sgn} \left(\sum_{\mathbf{x}_i \in \mathcal{S}} \alpha_i^* y_i \mathbf{x}_i \cdot \mathbf{x} + b^* \right) \quad (\text{A.17})$$

A.2 PLS Algorithm

PLS regression method [59, 60] is based on the linear transformation of a large number of descriptors to a new space based on a small number of orthogonal projection vectors. In other words, the projection vectors are mutually independent linear combinations of the of independent variables. These vectors are chosen to provide maximum correlation with the dependent variables. When the PLS is used to model a classification problem, the independent variables are the descriptors and the dependent variables are the labels of the data belonging to the training set. Next, we give a brief description of the PLS for linear case. The explanation below about the PLS was based on the works [1, 6, 23, 47]. More details about this method can be found in these tutorials.

Consider the general case of a linear PLS algorithm to model the relation between two data sets, \mathbf{X} and \mathbf{Y} . Denote by $\mathbf{X} \subset \mathcal{R}^N$ an N -dimensional space of variables representing the first set and similarly by $\mathbf{Y} \subset \mathcal{R}^M$ a space representing the second set of variables. The PLS algorithm models the relations between \mathbf{X} and \mathbf{Y} by means of score vectors. Considering n data samples from each set of variables, PLS decomposes the $(n \times N)$ matrix of zero-mean variables \mathbf{X} and the $(n \times M)$ matrix of zero-mean variables \mathbf{Y} using the Equation A.18

$$\begin{aligned} \mathbf{X} &= \mathbf{TP}^T + \mathbf{E} \\ \mathbf{Y} &= \mathbf{UQ}^T + \mathbf{F} \end{aligned} \quad (\text{A.18})$$

where the \mathbf{T} , \mathbf{U} are $(n \times p)$ matrices of the p extracted score vectors (latent vectors), the $(N \times p)$ matrix \mathbf{P} and the $(M \times p)$ matrix \mathbf{Q} are matrices of loadings and the $(n \times N)$

matrix \mathbf{E} and the $(n \times M)$ matrix \mathbf{F} are the matrices of residuals. There is several ways to be obtain the latent vectors, and that a common algorithm used for this is the nonlinear iterative partial least squares (NIPALS) algorithm [58], which was used in this dissertation. This algorithm finds weight vectors \mathbf{w} , \mathbf{c} such that

$$[\text{cov}(\mathbf{t}, \mathbf{u})]^2 = [\text{cov}(\mathbf{X}\mathbf{w}, \mathbf{Y}\mathbf{c})]^2 = \max_{|r|=|s|=1} [\text{cov}(\mathbf{X}\mathbf{r}, \mathbf{Y}\mathbf{s})]^2 \quad (\text{A.19})$$

where $\text{cov}(\mathbf{t}, \mathbf{u}) = \mathbf{t}^T \mathbf{u} / n$ denotes the sample covariance between the score vectors \mathbf{t} and \mathbf{u} . The NIPALS algorithm starts with random initialization of the \mathbf{Y} -space score vector \mathbf{u} and repeats the sequence of steps describe in Algorithm 1.

Algorithm 1 NIPALS

Require: \mathbf{X} , \mathbf{Y} , \mathbf{u}

```

while ( $t > \delta$ ) do
   $\mathbf{w} = \mathbf{X}^T \mathbf{u} / (\mathbf{u}^T \mathbf{u})$ 
   $\|\mathbf{w}\| \rightarrow 1$ 
   $\mathbf{t} = \mathbf{X}\mathbf{w}$ 
   $\mathbf{c} = \mathbf{Y}^T \mathbf{t} / (\mathbf{t}^T \mathbf{t})$ 
   $\|\mathbf{c}\| \rightarrow 1$ 
   $\mathbf{u} = \mathbf{Y}\mathbf{c}$ 
end while

```

PLS is an interactive process. After the NIPALS algorithm stop, that is, when \mathbf{t} has converged, it is compute the value of b which is used to predict \mathbf{Y} from \mathbf{t} as $b = \mathbf{t}^T \mathbf{u}$, and compute the factor loadings for \mathbf{X} as $\mathbf{p} = \mathbf{X}^T \mathbf{t}$. Now subtract the effect of \mathbf{t} from both \mathbf{X} and \mathbf{Y} as follows $\mathbf{X} = \mathbf{X} - \mathbf{t}\mathbf{p}^T$ and $\mathbf{Y} = \mathbf{Y} - b\mathbf{t}\mathbf{c}^T$. The vectors \mathbf{t} , \mathbf{u} , \mathbf{w} , \mathbf{c} , and \mathbf{p} are then stored in the corresponding matrices, and the scalar b is stored as a diagonal element of \mathbf{B} . The sum of squares of \mathbf{X} and \mathbf{Y} explained by the latent vector is computed as $\mathbf{p}^T \mathbf{p}$ and b^2 , respectively, and the proportion of variance explained is obtained by dividing the explained sum of squares by the corresponding total sum of squares. If \mathbf{X} is a null matrix, then the whole set of latent vectors has been found, otherwise the process can be started again from Algorithm 1.

The dependent variables are estimated as $\mathbf{Y} = \mathbf{T}\mathbf{B}\mathbf{C}^T = \mathbf{X}\mathbf{B}_{PLS}$ with $\mathbf{B}_{PLS} = (\mathbf{P}^{T+})\mathbf{B}\mathbf{C}^T$ (where \mathbf{P}^{T+} is the pseudo-inverse of \mathbf{P}^T , \mathbf{B} is a diagonal matrix with the regression weights as diagonal elements and \mathbf{C} is the weight matrix of the dependent variables, and the columns of \mathbf{T} are the latent vectors).