



Tohoku University
Graduate School of Information Sciences

High-Performance Biometric Recognition Using Phase-Based Image Matching

Luis Rafael Marval Pérez

A dissertation submitted for the degree of
Doctor of Philosophy (Information Sciences)

July 2019

Supervisor:
Professor Takafumi Aoki

Abstract

The objective of biometric recognition is the automatic authentication of individuals using their physiological or behavioral traits, e.g., fingerprint, iris, face, palmprint, and voice. Nowadays, various biometric traits are being employed for recognition in a wide range of applications, such as ATM services, authentication in personal devices, immigration controls, surveillance operations, etc. Biometric recognition is of great importance because it provides higher security, improves user convenience and can reduce the cost of business operations. The current deployment of biometric systems has reached a large population of users; and with it, the challenges and demands for higher recognition performance have significantly increased. In order to attain higher performance, a great deal of research has focused on improving accuracy, speed, and robustness of recognition methods. At the same time, recognition methods have to be practical so as to yield high performance with limited processing and memory resources.

Biometric recognition using phase-based image matching has demonstrated effectiveness across multiple biometric traits, such as fingerprint, iris, face, and palmprint. Phase-based image matching employs *phase features* computed by means of 2D Fourier transform of images. Recognition using phase features can achieve higher recognition accuracy than other techniques because phase features convey a detailed description of the biometric textures. In practice, for a wide variety of biometric traits, highly robust recognition is possible by (i) applying the phase-based matching technique to a set of local regions extracted from the given images and by (ii) introducing a coarse-to-fine strategy for searching corresponding pairs of local regions. This approach utilizes the facts that the local regions are less sensitive to nonlinear image deformations and occlusions frequently appeared in practical biometrics problems and that phase features offer high discrimination capability even for local textures.

Although biometric recognition using phase-based image matching has promising accuracy and robustness, there are two particular performance limitations. The first limitation is its high computational cost that results from the computation of a large number of local phase features across the given images and the

evaluation of local correlation functions defined on the local phase features. This computational cost is impractical in large-scale one-to-many recognition scenario, i.e., biometric identification, where an input image must be compared to all the images registered in the system. The second limitation is that the performance of phase-based image matching is sensitive to image quality and tends to decline on images that include regions with poor texture. Thus, for images with heterogeneous textures, such as periocular images, the high recognition accuracy attained by regions with distinct textures may be degraded significantly by the presence of regions with low-quality texture. This dissertation presents two major contributions that address each of the aforementioned problems, and it demonstrates practical impacts of the proposed ideas in real application scenarios: (i) designing a palmprint identification algorithm with reduced computational complexity and (ii) designing a periocular recognition algorithm with improved accuracy.

The first contribution of this dissertation is to propose a phase-based palmprint identification method that addresses the computational complexity in one-to-many comparisons. In order to reduce the computational cost for evaluating the vast number of phase-based correlation functions required in large-scale biometric identification, it is necessary to separate the overall computation into redundant and non-redundant parts. We found that the key for attaining such separation is to efficiently encode phase features by a set of weight vectors corresponding to a small number of common patterns. For this purpose, we propose a compact representation of phase features using Convolutional Sparse Coding (CSC) as well as a fast evaluation technique for phase-based correlation functions with this coding. On the basis of the proposed techniques, this dissertation also shows overall design of a palmprint identification algorithm applicable to large scale identification tasks; the algorithm effectively reduces the computation time of phase-based palmprint identification in one order of magnitude without significant degradation of identification accuracy.

The second contribution of this dissertation is to propose a phase-based periocular recognition method with improved accuracy. The periocular region is the region surrounding the eye, and this special biometric trait has attracted significant attention for recognizing individuals in unconstrained scenarios with low user cooperation; typical use cases include surveillance applications and immigration control applications. In order to achieve high recognition performance for phase-based image matching on periocular images, we have to deal

with the heterogeneous textures of the periocular components. We found that texture enhancement with variance normalization significantly improves representation capability of phase features for images with heterogeneous textures, such as those observed in the periocular region. Experimental evaluation using public databases demonstrates the higher recognition accuracy of the proposed method when compared with conventional methods. The proposed method yields state-of-the-art recognition accuracy in periocular recognition.

The contributions presented in this dissertation expand the applicability of phase-based biometrics and open new directions for future research.

To my fiancée Μαίκο, my family, and my dear friends.

Acknowledgments

Thanks to God for all the blessings.

This dissertation summarizes my doctoral research at Aoki Laboratory. Albeit the road to the doctoral degree has been arduous, long and uncertain, many amazing people helped me with their advice and support. To them, I owe the successes of this research.

First and foremost, I want to express my utmost appreciation to my supervisor Professor Takafumi Aoki for his guidance, and crucial support throughout my studies in Japan. His mentorship and unfailing enthusiasm encouraged me when everything seemed impossible. His assertive insights and advice were always enlightening when dealing with complex problems and intricate ideas. There are no words that can overstate the value of his support.

I gratefully acknowledge Professor Ayumi Shinohara and Professor Masayuki Kawamata for serving as my supervisory committee. I am grateful for their valuable comments and feedback during the examination of this dissertation. They helped me to improve the quality of this dissertation. I am honored to have my research examined by them.

Along with my supervisor, I would like to express my heartfelt gratitude to Associate Professor Koichi Ito for his support, thoroughly guidance and meticulous suggestions during my research. These acknowledgments cannot be completed without extending my gratitude to Professor Naofumi Homma. I thank him for his support, wise guidance, and entertaining Igo matches.

I would like to thank all my friendly colleagues and former students in the Aoki Laboratory. They have made my daily life more interesting and comfortable. In particular, I acknowledge Dr. Shoichiro Aoyama for his fruitful discussions and interchanges. Along with him, I especially thank Dr. Mamoru Miura, Dr. Shuji Sakai, Dr. Sho Endo, Dr. Rei Ueno, Dr. Shunsuke Tatsumi, Dr. Yuichiro Tajima, and Dr. Toru Takahashi.

I am also grateful to the Japanese Ministry of Education, Culture, Sports, Science, and Technology for granting me a scholarship during the period of 2009

until 2015. In the absence of this financial aid, I would not have been able to pursue graduate studies at Tohoku University.

I want to thank the wonderful friends that I made in Sendai. They made my life in Japan more pleasant and interesting. Especially, I thank my host mother Mrs. Yasuko Kokubun for her kind support and attentions. Along with her, I want to thank the members of the International Exchange Group Hagi-No-Kai. Through this group, I participated and enjoyed many incredible cultural activities that made my best memories of Japan. I thank my friend Mr. Masami Konno for his advice and supportive conversations. I want to thank all my international friends. As a personal note, I want to mention some of them: Gantoli Bayarbaatar, Ahmed Ebrahim Abdulla, Erick Mas Samanez, Erick Katsuragi, Andrzej Litwinowicz, Juan Gonzales, Iulia Prodan, Milena Mihaylova Mileva, Evelyn Gutierrez, Hector Villatoro, Lida Sánchez, Juandiego Fonseca, Paola Sanoni-Gagen, Luis Aparicio, Ricardo Garcia, Hector Villatoro, Jose Salazar, Marco León, and Luis Moya. I thank them and many more including the students of the Japanese language intensive course spring 2009 at Kawauchi campus. I treasure the memories of the time we spent together in Japan.

I thank my dear siblings, Jesus, Marisol, Elisa, Rosa, and Isolina, who cherished me through difficult times. I thank my loving aunts and relatives, especially, my cousins, Ezequiel Quijada and Marisol Fahnert. I owe a special thanks to my uncle Filadelfo Pérez for his valuable support. I thank Maiko Sato for her understanding and patience, and for many joyful moments. Last but not least, I thank my parents, Jesus and Isolina, for their prayers, unconditional love and encouragement. They were always with me in mind and heart at the distance.

Luis Rafael Marval Pérez

18th July 2019, Sendai

Contents

List of Figures	ix
List of Tables	xiii
1 Introduction	1
1.1 Biometric recognition using image matching	2
1.2 Phase-based image matching	4
1.2.1 The drawback of phase-based biometric identification	5
1.2.2 The drawback of phase-based image matching with heterogeneous textures	6
1.2.3 The limitations of a single biometric trait	6
1.3 Objectives and contributions	7
1.4 Structure of the dissertation	8
2 Preliminaries of biometric recognition	11
2.1 Biometrics	11
2.2 Biometric traits	12
2.2.1 Palmprint	15
2.2.2 Iris	15
2.2.3 Face	16
2.2.4 Periocular region	17
2.3 Biometric systems	17
2.3.1 Sensing module	18
2.3.2 Preprocessing	19
2.3.2.1 Palmprint preprocessing	20
2.3.2.2 Periocular region preprocessing	20
2.3.3 Feature extraction	21
2.3.4 Feature matching	21
2.4 Conventional image matching methods for biometrics	22
2.4.1 Methods based on Gabor-coding	23

2.4.2	LBP	25
2.4.3	SIFT	26
2.5	Recognition performance metrics	27
2.5.1	Verification metrics	29
2.5.2	Identification metrics	32
2.6	Summary	33
3	Phase-based image matching for biometric recognition	35
3.1	Cross-correlation	35
3.2	Phase-Only Correlation (POC)	37
3.2.1	Windowing	38
3.3	Spectral weighting	43
3.4	Band-Limited Phase-Only Correlation (BLPOC)	43
3.5	Subpixel estimation	44
3.5.1	Three values estimation	46
3.5.2	Two values estimation	46
3.5.3	Subpixel estimation for BLPOC	47
3.5.4	Subpixel estimation for images	47
3.6	Phase feature vectors	48
3.7	Robustness of phase-based image matching	50
3.8	Phase-Based Correspondence Matching (PB-CM)	53
3.8.1	Correspondence search	55
3.8.2	Similarity evaluation	57
3.8.3	Design with local phase array	57
3.9	Summary	58
4	Phase-based palmprint identification with sparse representation	59
4.1	Introduction	59
4.2	Review of BLPOC matching	62
4.3	Phase-based image matching with CSC	64
4.3.1	CSC representation of phase templates	65
4.3.2	CSC-based approximation of BLPOC functions	66
4.3.3	Computational advantage of CSC-BLPOC	66
4.4	CSC optimization algorithm	68
4.4.1	Problem statement	68
4.4.2	Basic strategy for optimization	69
4.4.3	Incremental CSC optimization algorithm	71

4.4.4	Sparse code optimization	71
4.4.5	Filter optimization	72
4.5	Palmprint identification with Phase-Based Hierarchical Block Matching . .	74
4.5.1	Phase-Based Correspondence Matching	75
4.5.2	Phase-Based Hierarchical Block Matching (PB-HBM)	78
4.5.3	Phase-Based Hierarchical Block Matching with Convolutional Sparse Coding (PB-HBM-CSC)	81
4.6	Experiments and discussion	84
4.6.1	Methods	84
4.6.2	Palmprint databases	84
4.6.3	Parameters	85
4.6.4	Evaluation of identification performance	86
4.6.5	CSC representation	89
4.7	Summary	90
5	Phase-based periocular recognition with texture enhancement	93
5.1	Introduction	93
5.2	Conventional methods for periocular recognition	95
5.3	Periocular image matching based on phase features	97
5.3.1	Texture enhancement	97
5.3.2	Correspondence search	100
5.3.3	Similarity evaluation	100
5.4	Overall periocular recognition algorithm	103
5.5	Experimental evaluation	104
5.5.1	Periocular databases	106
5.5.2	Baseline methods	106
5.5.3	Effect of texture enhancement on the recognition accuracy	107
5.5.4	Quantization of phase features	107
5.5.5	Comparative performance evaluation	110
5.5.6	Comparison with methods based on trained features	111
5.6	Summary	115
6	Conclusion	117
6.1	Future work	118

A	Score-level fusion of multi-biometrics	119
A.1	Introduction	119
A.1.1	Biometric recognition using phase-based image matching	121
A.2	Score fusion approaches	121
A.2.1	Density-based approach	122
A.2.2	Transformation-based approach	122
A.3	Experimental evaluation	123
A.3.1	Virtual multibiometric datababases	123
A.3.2	Comparative evaluation of the score level fusion	125
A.4	Summary	126
	References	129

List of Figures

2.1	(a) Detection of three facial traits: (b) face itself, (c) periocular region, and (d) iris.	13
2.2	Palm's major creases or principal lines.	16
2.3	Flow diagram of a biometric system for the case of periocular recognition. .	18
2.4	Palmprint preprocessing [61]. (a) Detection of the region of interest in a palm image: fiducial points (blue 'X' mark), center of the region of interest (red 'x' mark) and region of interest (black box), (b) Gray-scale preprocessed image after normalization.	20
2.5	Competitive code example. (a) Elliptical Gaussian filters, (b) Palmprint image and its competitive code.	23
2.6	Local Binary Pattern. (a) Local binary pattern extraction from a pixel with a square neighborhood, (b) circular neighborhood, (c) LBP and histogram extraction example.	26
2.7	Matching of SIFT features. a) Yellow markers indicate key-points, their scale and their main direction, b) histograms of gradients depicted in green color, c) correspondences obtained from feature matching.	28
2.8	Probabilistic distribution of matching scores for performance assessment. (a) Score distributions, (b) FAR and FRR, (c) Operative values for EER estimation.	30
2.9	Curves plot for performance evaluation of biometric systems. (a) ROC curve, (b) CMC curve.	32
3.1	Target localization with cross-correlation function and with phase-only filtering. (a) Degraded image: red square ground true for the target location, blue square target localization on padded image. (b) Pattern image or target object. (c) Spatial pattern of POF (reversed impulse response). (d) Zero-padded pattern image and the displacement (δ_1, δ_2) necessary for alignment. (e) Cross-correlation function and (d) correlation function with POF	36

3.2	Examples of POC functions and POF correlation functions. (a) Image pair comparison with small translational displacement. (b) Image pair comparison with large translational displacement. (c) Image pair without similarity: face images of different persons (impostor case).	39
3.3	Wrap around effect of the periodic expansion of the lena image. (a) Periodic image expansion and (b) periodic image expansion after applying a Hann window. Observe the smooth transition in the image border when the window function is applied.	40
3.4	Effect of window functions on two basis functions (1D case, $L = 33$). Spatial domain: the blue line is the real part of the basis functions, the red line is the imaginary part of the basis functions, and the green line indicates the envelope/window function. Sequence data indicates the Fourier transform of the function.	42
3.5	POC peak model	45
3.6	Effect of noise on POC and BLPOC functions: (a) image $g(n_1, n_2)$ for different levels of noise, (b) image $f(n_1, n_2)$ and phase difference of the image pairs, (c) image after applying Hann window $f(n_1, n_2)w(n_1, n_2)$ and phase difference of the image pairs, (d) weighted POC function and BLPOC function, (e) peak height comparison for POC and BLPOC variations.	51
3.7	Effect of rotation on POC and BLPOC functions: (a) image $g(n_1, n_2)$ for different rotations, (b) image $f(n_1, n_2)$ and phase difference of the image pairs, (c) image after applying Hann window $f(n_1, n_2)w(n_1, n_2)$ and phase difference of the image pairs, (d) weighted POC function and BLPOC function, (e) peak height comparison for POC and BLPOC variations.	52
3.8	Effect of scale variation on POC and BLPOC functions: (a) image $g(n_1, n_2)$ for different scales, (b) image $f(n_1, n_2)$ and phase difference of the image pairs, (c) image after applying Hann window $f(n_1, n_2)w(n_1, n_2)$ and phase difference of the image pairs, (d) weighted POC function and BLPOC function, and (e) peak height comparison for POC and BLPOC variations.	54
3.9	Sketch of phase-based correspondence search.	56
3.10	Sketch of three layers PB-CM.	57
4.1	Computation flow of BLPOC function between two image blocks extracted from the same person. Note that only real parts of $X(k_1, k_2)$, $Y(k_1, k_2)$ and $R(k_1, k_2)$ are visualized in order to demonstrate the effects on spectral weighting in Fourier domain.	63
4.2	Flow diagram of the CSC optimization algorithm.	70

4.3	Overall of palmprint identification using Phase-Based Correspondence Matching (PB-CM) with two-layer resolution pyramid. Left: enrollment process. Right: identification process.	76
4.4	Overall of palmprint identification using Phase-Based Hierarchical Block Matching (PB-HBM) algorithm with two-layer resolution pyramid. Red boxes on fine features indicate the selected features for similarity evaluation. Left: enrollment process. Right: identification process.	77
4.5	Enrolment for palmprint identification using two-layer Phase-Based Hierarchical Block Matching with Convolutional Sparse Coding (PB-HBM-CSC)	78
4.6	Overall palmprint identification using two-layer Phase-Based Hierarchical Block Matching with Convolutional Sparse Coding (PB-HBM-CSC) . . .	79
4.7	CMC curve evaluated using the CASIA database for all methods and magnified view of CMC curves for identification rates higher than 99.94%.	87
4.8	Phase templates in spatial domain and its reconstructed versions with $N_d = 16$ and $N_c = 32$	88
4.9	Mean square error (%) of the CSC reconstruction of phase templates for number of coefficients $N_c = 16, 24, 32, 40$ and number of filters $N_d = 12, 16, 20$.	89
4.10	Impulse response of filters used in PB-HBM-CSC with $N_d = 16$ and $N_c = 32$.	90
5.1	Components in the periocular region. Image sample from the UBIPr database.	94
5.2	Correspondence matching examples for two periocular images of the same person (genuine pair): (a) using m-SIFT features matching and (b) using PB-CM. Red dots indicate corresponding locations successfully estimated, and blue dots indicate failed estimations.	96
5.3	Texture enhancement with variance normalization. Consists of three steps: (i) extract one patch per pixel, (ii) normalize the variance of each patch and (iii) combine them into a new image.	97
5.4	Effect of texture enhancement on the BLPOC function for comparison of a genuine block pair with poor texture (left: without texture enhancement, right: with texture enhancement).	99
5.5	Overall of the proposed periocular recognition method. (a) Enrolment procedure of phase-based correspondence matching with texture enhancement. (b) Verification procedure of phase-based correspondence matching with texture enhancement.	101

5.6	Block center selection using thresholding to detect reflections: (a) The original image pair. (b) The reflection masks and block locations with selected locations indicated by red dot and the discarded locations indicated by blue dots. (c) The image after filling the areas that have reflections and the final block locations used for matching	103
5.7	Comparison of ROC curves of the proposed method and conventional methods.	113
A.1	Multibiometric systems: (a) conventional system using multiple matchers, and (b) proposed system using PB-CM (local phase array).	120
A.2	Examples of ROI images in each database and EERs(%) of each recognition algorithm: (a) face, (b) iris, (c) palmprint, and (d) finger knuckle.	124

List of Tables

2.1	Comparison of biometric traits [31]. (Low \triangle , Medium \circ , High \odot) . . .	14
2.2	Efficient 3-bit direction encoding for Competitive Code templates	25
3.1	Relation of the quantization error with the number of bits. Maximum and variance of the magnitude of the quantization error, i.e., $\text{Max}(\epsilon)$ and $\text{Var}(\epsilon)$, respectively.	50
4.1	Computational cost of BLPOC functions and CSC-BLPOC functions.	68
4.2	Parameters for Gabor-coding based methods.	85
4.3	Parameters for phase-based methods.	86
4.4	Experimental results of identification errors and computation times.	87
4.5	Registered data per image for each method.	88
5.1	Overview of periocular recognition works with traditional features.	95
5.2	Parameters for PB-CM and texture enhancement. Parameters indicated with * are used in the comparative evaluations	108
5.3	Verification performances EERs(%) of phase-based image matching with and without texture enhancement. Bold fonts indicate the best EER(%) per similarity measure.	109
5.4	EERs (%) of phase-based image matching with and without quantization of phase angles. Bold fonts indicate the best EER per similarity measure.	110
5.5	EERs (%) of the proposed method and conventional methods on CASIA database, UBIPr database and FOCS database, where TE indicates texture enhancement. Bold fonts indicate the best EER.	112
5.6	Experimental setup for comparative evaluation with SCNN.	114
5.7	Verification performances EERs(%) on a subset of CASIA database (comparison with SCNN). Bold fonts indicate the best EER.	114
A.1	EERs(%) for all combinations and fusion rules.	128

1

Introduction

Who are you? As simple as it may sound, this question has a significant role in today's society. Globalization, increase in population, and technological advances have brought an everyday growing complexity to society where this question is made and answered constantly. Concrete examples include: Are you the owner of this smart-phone? Are you a wanted criminal? Are you authorized to enter to this facility? In response, your identity can be verified by what you know, by what you have or, ultimately, by *what you are*. Passwords can be forgotten or guessed; cards and keys can be misplaced, stolen or duplicated; but physical and behavioral characteristics are intrinsic to the individuals and, therefore, reliable for their authentication. The study of techniques and systems for recognition of individuals through measurements of such characteristics is addressed by the field of *biometric recognition*, also referred to, with arguable ambiguity, as *biometrics*.

Nowadays, several characteristics or *biometric traits* are being utilized for recognition. Among the physical traits, there are fingerprint, face, iris, palmprint, palm veins, and DNA. Among the behavioral traits, there are voice, keystroke, and gait. There is not a single biometric trait perfect for all application scenarios, as each trait has different strengths and weaknesses in terms of distinctiveness, public acceptance and ease of measuring, among other performance attributes [1]. A suitable choice of a biometric trait, as well as sensor and system considerations, might differ from scenario to scenario. However, the measured or sensed data for most of these cases are collected in the form of image signals.

1.1 Biometric recognition using image matching

Biometric recognition using image matching has become widely available with the advent of imaging technologies and computing capabilities over the past decades [2]. Currently, it is already a common characteristic in cellphones, laptops, ATMs, border control, terminals and network access control, voting systems, subsidies and welfare entitlement systems, and prison access systems for inmates and visitors alike [3]. Biometric recognition increases the strength of security, provides greater convenience for users, can reduce the cost of business operations, and detects and deters frauds. It is easy to implement with a variety of commercial options for sensors and processing devices, such as low-cost tablets.

The current deployment of biometric systems has already reached a large population of users. For instance, border control of United Arab Emirates reports that around 14 billion iris cross-comparisons are carried out every day [4]; India Unique ID program combines fingerprint and iris recognition, and it has an enrolled population of 1.2 billion of users [5]. Despite its continuously growing popularity and success, biometric recognition is far from being a solved problem. On the contrary, with the spread of biometric applications, the challenges have only increased. This is because the larger the population of users is, the higher the performance requirements for the biometric system are. These requirements can be roughly simplified to: recognition accuracy, speed, convenience, system resources and robustness to attacks (such as spoofing). From the system requirements, high recognition accuracy is crucial in order to distinguish individuals with similar traits which are more common in large populations. At the same time, speed and convenience are not less important, but they are usually in conflict with the accuracy objective. Thus, the concept of high-performance biometric recognition, as regarded in the title of this dissertation, comprehend not only accuracy, but the other performance requirements as well.

Recognition involves sensing, preprocessing and image matching. The sensing step deals with the capture of raw images that contain the intended trait. Preprocessing consists in the extraction of the image area with the intended trait and the normalization of its properties such as scale, rotation, and contrast. Image matching essentially consists in computing feature vectors and comparing them to determine whether the images correspond to the trait of a specific person. Relaxing the constraints of the sensing setup improves user comfort and system convenience, but introduces variations that affect recognition accuracy. In palmprint recognition, for example, contactless acquisition introduces transformations due to the angle of the palm surface with respect to the camera and nonlinear deformations in the palmprint texture due to hand pose variations. Likewise, unconstrained face recognition has to deal with more complexity due to the 3D structure/shape of the face in addition to

occlusions related to scarves, hats and eyeglasses. In general, these problems are partially addressed by preprocessing but their burden ultimately falls on the image matching step.

Several feature vectors have been proposed for image matching. Trait-specific features such as minutia points can describe the ridge patterns in fingerprints. Minutiae are deviations of a single friction ridge, such as bifurcation, termination and crossovers [6]. However, minutia points do not have equivalents across other traits and cannot describe other important skin features like creases in palmprint. Features based on Gabor filters can describe the intricate texture pattern of the iris. The IrisCode method encodes and matches these features very efficiently [7]. It exhibits high recognition accuracy and fast matching speed. This fact has inspired other methods, such as Competitive Code [8] and Ordinal Code [9] in palmprint recognition and finger-knuckle recognition. However, these methods are not robust enough against deformations caused by hand pose variations. Face recognition using histograms of Local Binary Pattern (LBP) [10] showed robustness to facial expressions and illumination conditions. This method proved to be fast and versatile across multiple biometric traits, but its accuracy is limited. Scale-Invariant Feature Transform (SIFT) [11] proved effective in face and palmprint recognition [12]. SIFT is very robust against the problems of unconstrained recognition across different traits. Its accuracy increase with the number of feature points but this number depends on the texture quality of the image. Other methods learn the features from training data. In face recognition, for example, subspace and manifold methods have been largely studied [13, 14]. Recently, neural networks are being used as a general method for feature extraction on various traits and scenarios [15, 16]. These methods require a considerable amount of training samples which might not be available or representative of the intended scenario.

A highly accurate signal/image matching technique known as *Phase-Only Correlation* (POC) has a notable potential for biometric recognition. It utilizes phase information as feature vectors which can convey a detailed description of biometric textures. Biometric recognition using phase-based image matching has already accomplished high recognition accuracy and a considerable degree of robustness across different biometric traits [17]. Although it has a comparatively advantageous trade-off, it does not satisfy all the high performance requirement sought for applications with large population of users. This dissertation further develops the potential of phase-based image matching and contributes to its applicability on a wide range of scenarios.

1.2 Phase-based image matching

The research group to which the author belongs has focused on developing algorithms using POC. POC measures similarity and translational displacement between two signals in the form of a correlation function, i.e., *POC function*. The main difference with a cross-correlation function is that a POC function is computed using the phase component of the Fourier transform of the signals by normalizing the amplitude component or setting it to a predetermined value. POC functions display a sharp and distinctive peak when the signals have a consistent degree of similarity, e.g., images of the same palm. POC was introduced by Kuglin and Hines in 1975 [18] and its importance was identified by Oppenheim et al. 1981 [19]. Oppenheim pointed out that texture appearance or image information is mostly contained in the phase component of the Fourier transform. From the early 2000's, the research group has been developing algorithms to exploit POC's potential for image registration. Image registration is a fundamental task in computer vision that consists in finding the correspondence of pixels or coordinates between images. Takita et al. 2003 [20] accomplished sub-pixel registration by fitting an analytic model of the POC function. Then, Takita et al. 2004 [21] proposed a robust and accurate image registration technique that employs a coarse-to-fine strategy to find corresponding points between image pairs. These techniques allow accurate and robust 3D measurements [22] and motion estimation [23]. Ito et al. 2004 [24], 2005 [25] and 2006 [26] proposed and developed a variant of POC, *Band-Limited POC* (BLPOC), for fingerprint recognition. Miyazawa et al. 2005 [27] followed with a similar development for iris recognition. Motivated by Takita et al. [21], Ito et al. proposed phase-based correspondence matching for palmprint recognition [28] and for face recognition in a later work [29]. Aoyama et al. 2013 [17] formalized this recognition method and demonstrated the versatility of phase information to describe different biometric textures.

Biometric recognition using phase-based correspondence matching combines a coarse-to-fine correspondence search strategy with local region matching. Local regions are less affected by global transformations between images, illumination conditions and occlusions. Coarse-to-fine search using resolution pyramids can solve large translational displacements robustly. However, research on phase-based correspondence matching is far from being complete as there are performance limitations that remain to be studied.

This dissertation addresses two important problems related to the performance limitations of phase-based correspondence matching: The first problem derives from its relatively high computational cost, and the second problem is related to the degradation in accuracy of POC on images with poor texture. Each problem is addressed in a representative

application scenario, i.e., palmprint identification and periocular recognition, respectively. In addition, this dissertation includes a study on score-level fusion for recognition using multiple biometric traits. Such *multibiometric recognition* can overcome the limitations of recognition using a single trait, e.g., not universal across large populations of users, susceptible to spoofing attacks.

1.2.1 The drawback of phase-based biometric identification

In order to clarify this problem, we first explain the operation modes of biometric recognition systems, namely, *verification mode* and *identification mode* [30]. In the verification mode, the identity claimed by a user is verified by comparing a test image of the user trait with an image corresponding to the claimed identity. Identification mode, on the other hand, is to find the identity of the user among the identities registered in the systems by comparing a test image of the user trait with all the images registered in the system. Operation in verification mode is also referred to as one-to-one recognition, while operation in identification mode is also known as one-to-many recognition. Clearly, the computational cost of identification increases with the number of enrolled users while the computational cost of the verification is constant regardless of such number.

The computation of a single POC operation is relatively efficient due to the Fast Fourier Transform (FFT), but accurate recognition using phase-based correspondence matching involves a considerable number of POC operations for local comparisons. The computational cost of phase-based correspondence matching in the verification mode is acceptable. However, in the identification mode, the computational cost is significantly high, especially when a large number of users is enrolled in the system. As a consequence, the identification time of phase-based correspondence matching is impractical, if not prohibitive. To avoid this problem, conventional identification methods employ feature vectors with a minimal description of the biometric texture. Instead, phase information conveys a very detailed description of the biometric texture.

We address this problem in the scenario of palmprint identification due to its particular advantages. The palms of the human hands contain ridges, creases and wrinkles over an area larger than fingerprints [31]. Palmprint images can be captured by a simple camera setup instead of a specialized scanner, and palmprint recognition is convenient to users in commercial applications as their hands become their ID cards.

1.2.2 The drawback of phase-based image matching with heterogeneous textures

The second problem is regarding the comparison of images with low-quality textures. Although POC is effective across various biometric textures, its performance degrades significantly when it is applied to regions with poor texture. This is because phase features are not suitable for describing texture details that are minute and contribute poorly to the overall pixel variance. Consequently, POC's performance degrades on heterogeneous texture where the poor texture regions contribute negatively to the accuracy attained by regions with rich texture.

We address this problem in periocular recognition. The periocular region – the region surrounding the eyes – has attracted a great deal of attention for recognition in scenarios where iris recognition and face recognition are not reliable. Iris recognition requires high quality images that are difficult to acquire at the distance; and it can be argued that face recognition is strongly affected by occlusions, facial expressions, and lighting variations. The periocular region is the part of the face with highest discrimination performance. It has a heterogeneous texture, i.e., its components exhibit different strengths in texture appearance. For example, while eyebrows exhibit a very distinctive texture appearance, the facial skin might exhibit a poor texture appearance due to low contrast of minute features. Effective periocular recognition has to deal with partial occlusions caused by eyeglasses, specular reflections, hair and hats among others; also, it has to deal with variations in head pose and facial expression.

1.2.3 The limitations of a single biometric trait

In the appendix of this dissertation, we consider a problem that is inherent to biometric recognition with a single biometric trait. A single trait might not be a universal characteristic since not everyone has it, and it can be less accurate for certain groups of people; such as elderly, racially homogeneous populations, relatives and twins [32]. Recognition with a single trait has limited accuracy and is susceptible to spoofing. The answer to these limitations is to employ multiple biometric traits, i.e., multibiometric recognition. The limitations of a single trait can be overcome by combining multiple sources of information. For this purpose, information can be combined at any step, or level, of the recognition process. For example, we can fuse the preprocessed images, the feature vectors, the matching scores, or the decisions of each trait. Among these levels, fusion at score-level is more effective since scores comprise most of the matching information in a simple scalar value. However, an improvement over the accuracy of a single trait is not ensured since scores have different

distributions, and the complexity of the recognition system grows with the number of biometric traits. Phase-based correspondence matching is appealing for multibiometric recognition due to its effectiveness across different biometric traits, although there are still no studies considering it. Moreover, comparative research about the performance of different recognition methods on multibiometric recognition is limited.

There is an important motivation for using the same algorithm for multiple traits, i.e., phase-based correspondence matching, instead of specialized algorithms. Using the same algorithm simplifies the implementation of a multibiometric system as opposed to the implementation with specialized methods which adds hardware complexity and requirements to the system.

1.3 Objectives and contributions

This dissertation aims to expand the performance capabilities of phase-based image matching through two objectives. The first objective is to solve the problem of one-to-many comparison within identification for phase-based image matching. This is to reduce the computational complexity of identification, which increases with the number of enrolled users, so that identification can be performed in a practical time. The second objective is to attain high recognition accuracy in phase-based periocular recognition. For this purpose, we have to address the performance degradation of phase image matching due to the heterogeneous texture within periocular images. This dissertation comprises two main contributions:

- **(i) A practical method for phase-based palmprint identification with sparse representation:**

This method can perform palmprint identification in a practical time with no significant detriment of the high recognition accuracy of phase-based correspondence matching. The method utilizes a fast phase-based image matching derived from convolutional sparse coding of phase features. In addition, we propose an optimization algorithm for learning the convolutional codes necessary for fast phase-based image matching. The sparse representation also reduces the storage requirements for the users' biometric data. The strategy of this identification method can be extended to similar methods in the field of correlation pattern recognition [33].

- **(ii) A highly accurate method for phase-based periocular verification with texture enhancement:**

This method combines phase-based correspondence matching with a texture enhancement technique based on variance normalization. Variance normalization successfully

equalizes the heterogeneous texture within the periocular region and provides a clear improvement over recognition without texture enhancement. This performance improvement can be translated in phase-based recognition using other biometric traits. To the best of the author's knowledge, the proposed method accomplishes the highest recognition accuracy in the literature for two public databases.

We append to this dissertation a study on score-level fusion of multibiometric recognition using phase-based correspondence matching (PB-CM). This study presents a comprehensive experimental evaluation of score-level fusion for recognition using iris, face, palmprint and finger-knuckle print. The experimental evaluation demonstrates that in comparison with specialized methods, phase-based correspondence matching is advantageous in score-level fusion, since it provides scores that yield a successful performance improvement for score-fusion even with simple fusion rules.

1.4 Structure of the dissertation

This dissertation can be roughly divided into two parts. The first part comprises introduction and background in chapters 1-3, and the second part comprises the main contributions in chapters 4 and 5.

Chapter 2 presents the preliminaries of biometric recognition and image matching. This chapter covers fundamental concepts, such as, biometric traits attributes and performance metrics. It includes a brief revision of conventional recognition methods as well.

Chapter 3 describes the fundamentals of image matching using POC, and thoroughly revises the Band-Limited POC properties. It presents a baseline algorithm for biometric recognition using phase-based image matching, namely, phase-based correspondence matching.

Chapter 4 introduces a practical phase-based palmprint identification method using convolutional sparse coding. The chapter presents a novel derivation of an efficient one-to-many phase-based image matching from convolutional sparse coding of phase features. The chapter also introduces a novel optimization algorithm for learning the sparse representation required for the efficient one-to-many matching. This chapter includes an experimental evaluation that demonstrated the effectiveness of the proposed method in comparison with conventional methods.

Chapter 5 introduces a phase-based periocular recognition method that yields very high recognition accuracy. This chapter reviews conventional methods for periocular recognition before presenting the characteristic of the proposed method. The main characteristic is the introduction of a texture enhancement for periocular images that improves the recognition

performance. This chapter includes an experimental evaluation of the proposed method for different parameters, as well as an elaborated comparative evaluation with conventional methods.

Chapter 6 brings to an end the body of this dissertation with general discussions and the respective conclusion.

Additionally, the **Appendix** presents a study on score-level fusion for multibiometric using phase-based image matching. The appendix covers the fundamentals on multibiometric recognition, particularly, score-level fusion approaches. It includes a comprehensive experimental evaluation and analysis that show the advantages of phase-based image matching for score-level fusion of multibiometric recognition.

2

Preliminaries of biometric recognition

This chapter starts with a brief clarification of the term biometrics. Then it reviews the fundamentals of biometric traits in Sec. 2.2 and biometric systems in Sec. 2.3. This chapter also describes conventional biometric recognition methods in Sec. 2.4 and covers the basic metrics for evaluation of recognition performance.

2.1 Biometrics

The first appearance of the term *biometrics* was at the beginning of the 20th century as a synonym of *biometry*. Both words are derived from Greek words – *bio* – that means alive and – *metrics* or *metry* – which means to measure or related to measure [31]. In this sense, biometrics refers to the field of development of statistical and mathematical methods applicable to data analysis problems in the biological and medical sciences [34]. Examples of this field are data analysis for agriculture field experiments, human clinical trials, environmental studies, and variance between and within populations. Currently, biometry fits better this description, since biometrics acquired another meaning in the decade of the 80's [35]. Jain and Ross in “Handbook of Biometrics” define biometrics as “the sciences of establishing the identity of an individual based on physical, chemical or behavioral attributes of the person” [30]. Notwithstanding, this dissertation is concerned only with a narrow definition for biometrics or biometric recognition: “automated methods of recognizing a person based on a physiological or behavioral characteristic” [36]. For

simplicity, across this dissertation, we refer to these characteristics as biometric traits or simply as traits.

2.2 Biometric traits

Some one said long ago that it is impossible to find two leaves exactly alike; Nature never repeats herself. Select no matter what part of the human body, examine and compare it carefully in different subjects, and the more minute your examination is the more numerous the dissimilarities will appear: exterior variations, interior variations in the bony structure, the muscles, the tracing of the veins; physiological variations in the gait, the expressions of the face, the action and secretion of the organs, etc. (Alphonse Bertillon 1896 [37])

The excerpt above suggests a natural potential of any part of the human body to be used as a biometric trait. Detailed and precise measurements of a relatively invariable trait of a person can, in principle, lead to uniqueness with clear distinction from any other person. However, we have to rise the question: Is such measurement practical, or even meaningful? In this sense, we must consider different performance attributes that determine the strength and weakness of biometric traits. The followings are the attributes proposed by Jain et al. [30, 31]:

- **Universality:** Everyone should possess the trait and be able to use it. However, birth defects, accidents and customs might caused the loss or impede the use of a given biometric trait.
- **Distinctiveness/Uniqueness:** The traits should provide sufficient distinction to discern between any two individuals. Traits of twins and relatives can be very similar.
- **Permanence:** The trait should be sufficiently invariant over a functional period of time. Some traits can change significantly in short period of time or not be stable for some cases such as the face of infants.
- **Measurability/Collectability:** The trait should be measurable in a practical way, i.e., easy to acquire and digitize in a convenient manner to the user.
- **Performance:** The computational resources required to achieve an expected accuracy and throughput (number of transaction that can be processed per unit time) of the biometric system should suit the application's budget.

- **Acceptability:** The user population should be willing to present the trait to the system.
- **Robustness against circumvention/spoofing:** The trait should be difficult, if not impossible, to imitate or emulate using attacks, artifacts (e.g. fake fingers), or mimicry in the case of behavioral traits. Likewise, the trait should be difficult to alter or obfuscate so that users cannot evade recognition. Jain et al. [31, 38] refer to this attribute by its opposing term, circumvention. We prefer to use “robustness against/to” in order to maintain all the attributes as positive descriptions.

Notice that these attributes can not be separated from other factors, such as, the availability of technologies (e.g., sensor, processors) and cultural standards, which can affect in part these attributes. Table 2.1 shows a rough comparison of the attribute for common biometric traits.

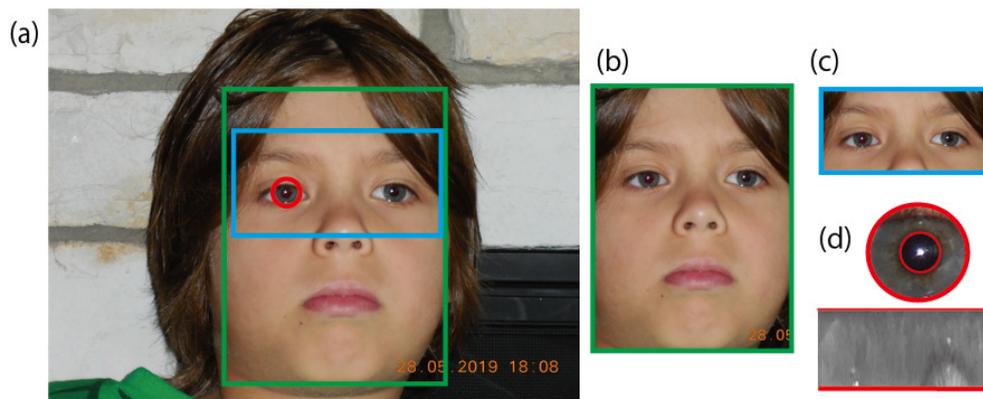


Figure 2.1: (a) Detection of three facial traits: (b) face itself, (c) periocular region, and (d) iris.

This dissertation focus mainly on two traits, palmprint and periocular region, which are dealt with in Ch. 4 and Ch. 5, respectively. These traits hold high potential for authentication systems because of their flexibility, acceptance and advantageous convenience. The periocular region can be regarded as a middle ground between face and iris; moreover, periocular recognition is, in the practice, a form of partial face recognition. In Appendix A, we consider for multibiometric recognition the traits face, iris, palmprint and finger-knuckle print. Here, we briefly describe palmprint, iris, face and periocular region.

Table 2.1: Comparison of biometric traits [31].
(Low \triangle , Medium \circ , High \odot)

Biometric trait	Universality	Uniqueness	Permanence	Collectability	Performance	Acceptability	Robustness to spoofing
Fingerprint	\circ	\odot	\odot	\circ	\odot	\circ	\circ
Face	\odot	\triangle	\circ	\odot	\triangle	\odot	\triangle
Face in infrared images	\odot	\odot	\triangle	\odot	\circ	\odot	\odot
DNA	\odot	\odot	\odot	\triangle	\odot	\triangle	\odot
Gait	\circ	\triangle	\triangle	\odot	\triangle	\odot	\circ
Iris	\odot	\odot	\odot	\circ	\odot	\triangle	\odot
Retina	\odot	\odot	\circ	\triangle	\odot	\triangle	\odot
Periocular	\odot	\odot	\circ	\odot	\odot	\odot	\circ
Keystroke	\triangle	\triangle	\triangle	\circ	\triangle	\circ	\circ
Voice	\circ	\triangle	\triangle	\circ	\triangle	\odot	\triangle
Hand veins	\circ	\circ	\circ	\circ	\circ	\circ	\odot
Palmprint	\circ	\odot	\odot	\circ	\odot	\circ	\circ
Finger-knuckle-print	\odot	\circ	\odot	\odot	\odot	\odot	\circ
Odor/Smell	\odot	\odot	\odot	\triangle	\triangle	\circ	\odot
Signature	\triangle	\triangle	\triangle	\odot	\triangle	\odot	\odot
Ear	\circ	\circ	\odot	\circ	\circ	\odot	\circ

2.2.1 Palmprint

The palm is defined as the inner surface of our hand, from the wrist to the root of the fingers [39]. Its print, the palmprint, is the impression of the skin patterns of the palm. A palmprint has a wider area than fingerprints and is rich in stable and unique features, namely, friction or papillary ridges and flexion creases [40]. Friction ridges are permanent thickenings of the epidermis in form of furrows. The purpose of the ridges is to increase friction with other surface on contact. Palmar flexion creases are folding lines of the skin and subcutaneous tissues during the movement of the palm and fingers. Most of the flexion creases generate on the fetus before the friction ridges, and are permanent through a lifetime. Flexion creases are divided into principal lines or major creases, and wrinkles (minor creases and secondary creases). In Fig. 2.2, we can observe the major creases: thenar crease, distal transverse crease, proximal transverse crease and sagittal crease [39, 41]. These are also referred by their names in palmistry, an esoteric art for divination, as follows: the life line, the heart line, the head line and the fate line, respectively [42, 43]. The wrinkles that form after the friction ridges do not have the same degree of permanence but are relatively stable. This also holds for wrinkles acquired after birth by frequent or intensive grip and grasp activities. As a consequence, palmprint have many features of different scales that are suitable for recognition. Palmprint recognition has a major role in forensic application since about 30% of the latents found at crime scenes are those of palms [1]. More on the importance on palmprint recognition is presented in Ch. 4.

2.2.2 Iris

The iris is the plainly visible pigmented ring-shaped membrane surrounding the pupil of the eye. It consists of a muscular structure between cornea and lens that regulates the amount of light transmitted to the retina by adapting the width of the pupil [36, 44, 45]. Irises contain intricate details of arching ligaments, striations, furrows, ridges, crypts, corona, freckles and a zigzag collarette [45]. These details form a rich random pattern that emerges during gestation and is unique with no detectable or known genetic dependencies [1]. Irises are uncorrelated between left and right eye of a single individual or eyes of monozygotic twins [45]. In this way, iris images provide such distinctive features that the iris is argued to be the most accurate biometric trait [36]. The iris pattern is formed in the third month of gestation, and becomes stable around the eighth month [46]. The iris stays relatively unchanged throughout a person's lifetime excepting cases of trauma and certain rare diseases. Iris recognition is highly universal since, only a very small part of the population is missing both eyes, moreover, some blind people can use it [36]. Two ophthalmologists, Leonard

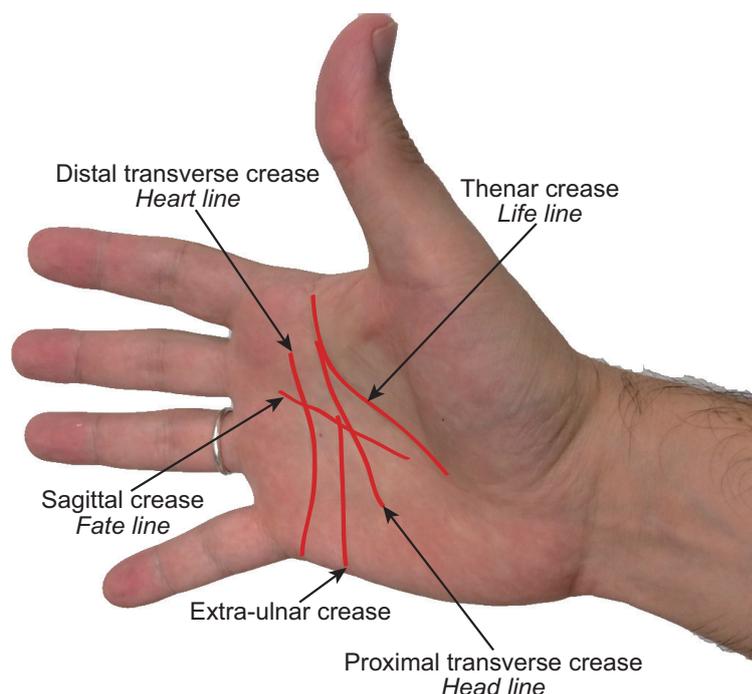


Figure 2.2: Palm's major creases or principal lines.

Flom and Arim Safir filed the first patent describing apparatus for iris recognition on visible features in 1987. However, the first modern automated iris recognition was patented by Dr. John Daugman of Cambridge University in 1984 [47]. He intensively developed algorithms and techniques to encode iris patterns and compare them in an efficient manner [1].

2.2.3 Face

The face is the frontal portion of the human head which extends from the forehead to the chin [31]. It is a perception center of the body as it includes the eyes for seeing, nose for smelling, mouth for tasting, ears for hearing and skin with relatively high touch sensitivity. It also reveals in a large extent gender, age, ethnicity, and emotional state of a person (e.g., happiness or anger). The face has two bones, the skull and jaw, and 44 muscles that are not attached to the bones. This enables a great liberty of movement and thus uncountable different facial expressions [14]. Face language, to some extent, is universal. Attention on the face for aesthetics and arts comes natural as the face can convey either hidden or visible beauty. The face is perhaps the most natural physical characteristic of the body for human to human recognition. Even newborns can distinguish their mother's face from other female faces [14]. At early stages of infancy humans have acquired the ability to process faces and differentiate facial attractiveness and facial expressions [14]. Thus, face photographs are incorporated for authentication in various tokens such as ID cards, passports, and

driver's licenses. Automatic face recognition is passive, natural and noninvasive, and its acceptability can be considered high since people are generally willing to share in social networks their face images with face tagging options [13, 31]. Although we humans perform it routinely and effortlessly in our daily lives, automatic face recognition still confront many challenges specially when face images are acquired in unconstrained setups [13].

2.2.4 Periocular region

The periocular region refers to the area of the face that includes the eyelids, eyelashes, eyebrow, and skin surrounding the eye [48].

Early indications of the potential of periocular region come from face recognition studies: Smeraldi and Begun 2002 [49], Savvides et al. 2006 [50] and Heisele et al. 2007 [51]. These studies evaluated person recognition performance using not only holistic-face images, but also images of specific parts of the face. For example, Savvides et al. [50] shows that images of the eye-region accounts for a recognition rate of 83%, $(\# \text{ of matches} / \# \text{ of test images}) \times 100\%$, while holistic-face images only improve this rate by 3% (i.e., 87% recognition rate). In 2009, Park et al. [52] explicitly proposed the periocular region as a biometric trait and, since then, it has received considerable attention (for surveys see [53, 54, 55, 56]). Although periocular recognition has attracted much attention, there is not a formal definition of the periocular region by formal bodies such as the U.S. Commerce Department's National Institute of Standards and Technology (NIST). Thus, the extent of the periocular region and whether it includes the eye, i.e., sclera and iris, varies across different studies [54, 55]. More on the importance of periocular recognition is presented in Ch. 5.

2.3 Biometric systems

A biometric system measures one or more biometric traits of an individual (test user) in order to determine or verify his identity [31]. Biometric systems have two operational stages, the enrolment stage and the recognition stage. The enrolment stage has the following steps: **(i)** acquisition of the raw biometric data from the user to be enrolled, **(ii)** preprocessing of the biometric data, **(iii)** extraction of a discriminative feature vector, and store the vector into the system's database, hereon referred to as the gallery. Fig. 2.3 depicts the flow of enrollment and recognition stages for the case of periocular recognition. The recognition stage has the following steps: **(i)** acquisition of the raw biometric data from an user to be tested, **(ii)** preprocessing of the biometric data, **(iii)** extraction of a discriminative feature vector, and **(iv)** comparison of the feature vector against one or a set of vectors registered in the database. From the result of this comparison, the system **(v)** decides and executes

an action [30]. The implementation of these steps leads to corresponding five modules in a biometric systems. **(i)** sensing module, **(ii)** preprocessing module, **(iii)** feature extraction module, **(iv)** matching module¹, and **(v)** decision module.

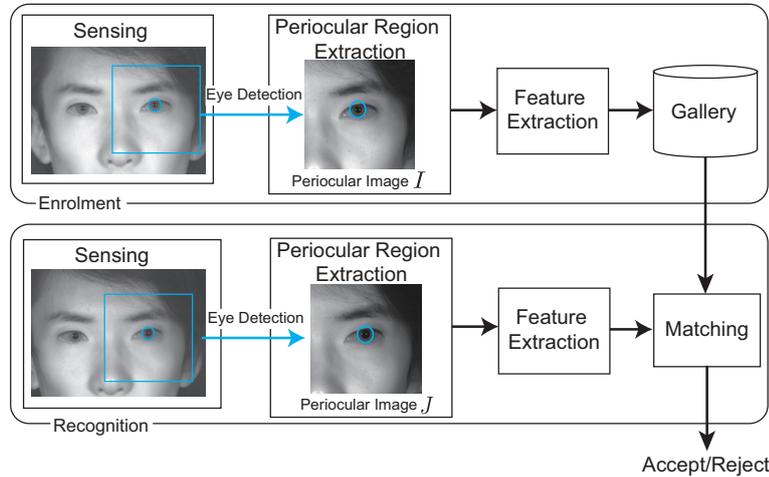


Figure 2.3: Flow diagram of a biometric system for the case of periocular recognition.

2.3.1 Sensing module

The sensor module defines the human machine interface, and it is essential to the performance of the whole system [31] as it acquires the raw biometric data of the users. In this dissertation, we are concerned only with images acquired by a camera sensor. The sensor module includes an imaging device (digital camera or camcorder), light source(s), and a control mechanism to trigger the acquisition or to track the user’s trait. Imaging devices have become cheaper and their technology is continuously improving in order to increase the image resolution, frame rate, and sensitivity. Early systems had very constrained interfaces. For instance, iris recognition used to employ an adjustable mechanical rack to position the users’ head. Since this affects user convenience, a great deal of research has focused in relaxing the constraints for user-friendliness and non-intrusiveness without affecting the recognition accuracy. This requires careful design of the sensor module and its camera setup. A less constrained system requires the trait to be placed in a designated space for its acquisition, e.g., standing in front of the camera for face/iris recognition. An unconstrained system could require the user to just walk normally through a gate or portal while being recorded, and it would grab the best frame(s) from the video. The problem with videos is that, for the same device, video acquisition provides lower resolution. In surveillance application, one approach to

¹The term *matcher* is also used to refer to the matching module or to the combination of feature extraction and matching module.

tackle the low resolution of video is to pair a static and a pan-tilt-zoom (PTZ) camera. A PTZ camera dynamically zooms in to obtain close-up images of the target objects while the static camera provides the wide field of view [31].

In settings with a fixed camera, the volumetric space where the trait can be captured (range of standoff distance) is determined by field of view and the *depth of field*. The depth of field (dof) is the distance in front of and beyond the subject that appears to be in focus [44]. When dof is increased, the luminance is decreased. This makes the picture darker and decreases the contrast of the image. This trade-off between image quality and depth of field is inevitable. Thus, camera setups must be fine-tuned to operate in a specific setting [44]. Another interesting case is palmprint recognition with personal devices such as cellphones [57] and tablets. In this scenario, the device aids the user by displaying a palm guide that overlaps the image to be acquired. Other setups take advantage of the physical mechanisms like door level setups for finger-knuckle-print acquisition [58].

Regarding the acquisition spectrum, biometric recognition is feasible in visible light spectrum (VIS), infrared spectrum (IR) and near-infrared wavelength (NIR). NIR and IR illumination have the advantages in low illumination environment and do not disrupt the user since they are not visible. Also, biometric textures such as skin and iris can be more stable in NIR acquisition.

2.3.2 Preprocessing

The purpose of preprocessing is to standardize the images for recognition so particular conditions of the image do not influence the recognition. In general, it comprise of the following steps: detecting the trait in the sensed image, extracting a trait image with the region of interest and normalizing it. This normalization involves correcting scale, rotation, alignment and illumination (photometric normalization). This process can vary considerably from one biometric trait to another. Thus, well defined landmarks that are relatively stable and robust for detection, i.e., fiducial points, are defined for the traits. For example, face landmarks such as eye corners or nose tip are known to be mildly affected by facial expressions [59]. They assist the detection of further landmarks such as the nostril corners, the mouth corners, the end points of the eyebrow arcs, ear lobes, chin, etc. In face recognition, we have to normalize the position of the face according to these fiducial points. In the case of iris recognition, we have to detect the inner boundary between the iris and pupil and the outer boundary between the iris and sclera, and then unwrap the iris region to a normalized rectangular block of a fixed size using polar coordinate transformation as it is shown in Fig. 2.1(d) [60]. Here, let us cover briefly palmprint preprocessing and periocular preprocessing.

2.3.2.1 Palmprint preprocessing

In Ch. 4, we employ the preprocessing method proposed in [61] to extract a palm region from a hand image. This method consists of 5 steps: (i) binarizing the images, (ii) extracting the contour of hand and/or fingers, (iii) detecting the fiducial points, (iv) establishing a coordinate system, and (v) extracting the central part. We detect two valleys in the contour of the fingers: the valley defined by index and middle fingers and the valley defined by the ring and little fingers. The points at the valley bottoms are used as fiducial points in the step (iii) to define the coordinate system. Observe in Fig. 2.4(a) the segment between fiducial points as a solid line and its perpendicular bisector line as a dashed line. The region of interest is a square region whose centroid is on the bisector line. We extract this palm/palmprint region and normalize it to a fixed size. A scale factor for the centroid coordinate and size of the region of interest is determined by the length of the segment, i.e., the distance between fiducial points. In this manner, this procedure normalizes scale, rotation and translation to an acceptable degree.

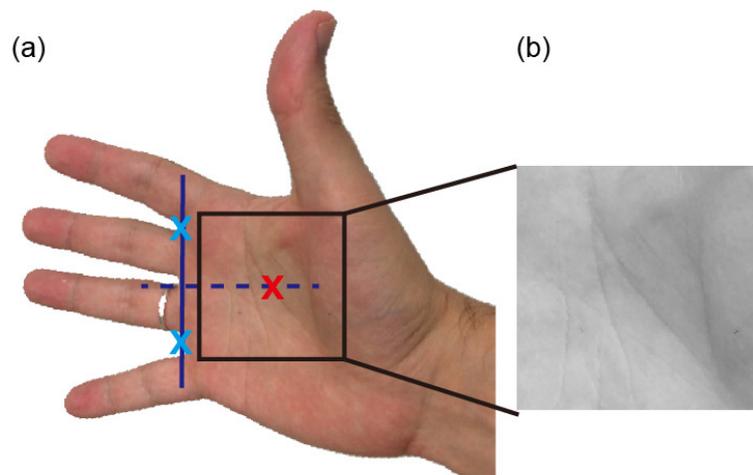


Figure 2.4: Palmprint preprocessing [61]. (a) Detection of the region of interest in a palm image: fiducial points (blue 'X' mark), center of the region of interest (red 'x' mark) and region of interest (black box), (b) Gray-scale preprocessed image after normalization.

2.3.2.2 Periocular region preprocessing

Periocular detection and segmentation has become a research topic by itself, which inherits the developments of iris detection. The following are some of the existing eye detection methods:

- Wildes 1997 [63] based on the Hough transform.

- Smeraldi and Bigun 2002 [49] based on features extracted with a Gabor filter bank.
- Viola and Jones 2004 [64, 65] based on AdaBoost. This method performs first face detection, and then it detects and extracts the eye regions from the face.
- Bolme et al. 2009 [66] employ a correlation filter called Average of Synthetic Exact Filter.
- Boddeti et al. 2011 [67] employ a correlation filter called Unconstrained Minimum Average Correlation Energy filter.
- Alonso-Fernandez and Bigun 2014 [68] proposed an eye detector using symmetry filters, computed from symmetry derivative of a Gaussian, that does not require training.

Although many studies have used the pupil or the iris as reference, it has been pointed out [69] that, as the eye is a moving part, this reference does not ensure alignment outside the eye. Instead, the corners of the eyes can be effective even when the eyes are closed.

The topic of Ch. 5 is periocular recognition, but we focus in the recognition itself rather than the preprocessing. For this reason, we employ manual detection of the pupil when needed or images that are already segmented.

2.3.3 Feature extraction

Feature vectors used for biometric recognition can be roughly classified into two categories. (i) Global features: these are extracted from the whole preprocessed image, for example, LBP and Gabor-coding based methods like IrisCode. (ii) Local features: these are extracted from a set of location or key-points using a limited surrounding area, for example, SIFT features and fingerprint minutiae. In the case of phase features for phase-based image matching, both global and local features have been considered. While the global phase-features are suitable for limited cases such as fingerprint recognition [24], local phase-features are more general and applicable for a broad range of traits and scenarios.

Features extracted and registered in the system's gallery are also referred to as templates. We employ this term in Ch. 4 to distinguish the features extracted during recognition stage from those extracted during enrolment stage.

2.3.4 Feature matching

Feature domain can vary broadly according to the nature of the feature, but, in general, the measures of similarity or dissimilarity are computed as scalar values. Similarity measures are usually inner products between vectors while dissimilarity scores are usually distance such as Euclidean distance between vectors or Hamming distance for binary feature vectors. If the features are normalized, their distance and inner products are normalized too. Given

two normalized feature vectors \mathbf{V}_a and \mathbf{V}_b , we can find a direct relation between their Euclidean distance D_{ab} and inner-product $\mathbf{V}_a^\dagger \mathbf{V}_b$.

$$D_{ab} = 2 - 2\mathbf{V}_a^\dagger \mathbf{V}_b, \quad (2.1)$$

where \mathbf{V}_a^\dagger is the Hermitian transpose of \mathbf{V}_a . Another common kind of score is a count of successful matching of local features, e.g., number of feature vector pairs that have high inner product values.

In the case of correlation methods, the correlation functions consist of inner-product values. A correlation function can be interpreted as a series of inner-products between images for different displacements. Thus, the computational cost of these methods is high but it is alleviated by the efficiency of the Fast Fourier Transform. We will delve deeper in this interpretation in Sec. 3.6.

As explained in the introduction chapter, there are two operation modes for recognition systems. The mode determines the kind of comparison in the recognition stage. In the *verification mode*, the system validates the identity claimed by the test user using a one-to-one comparison. Thus, the system compares the features of the the input image to the features of the registered image(s) corresponding to the claimed identity. In the *identification mode*, the system determines the identity of the test user through a one-to-many comparison. Thus, the system compares the features of the input image against the features of all registered images. Therefore, feature matching in identification is computationally expensive when a large number of registered images.

Finally, the **decision module** compares the matching score(s) provided by the matching module. In the verification mode, the score is compared with a operational threshold. If the score is higher the system executes an action such as grant access in access control application. In the identification mode, the module selects the registered identities specific to the highest scores. If the score values are greater than an operational threshold, the module executes an action, for example, notifying the identities to authorities in watch list applications. Also, the decision module can considerate confidence measures provided by other modules.

2.4 Conventional image matching methods for biometrics

We present here an overview of three kinds of methods: methods that employ Gabor-coding, methods that employ Local Binary Pattern (LBP) and methods that employ SIFT matching.

The image matching techniques of these methods have been applied to a considerable variety of biometric traits.

2.4.1 Methods based on Gabor-coding

IrisCode has been notoriously successful in iris recognition. Its commercial success derives from its computational advantages, including high matching speed, binomial impostor score distributions and robustness against local brightness and contrast variations. For this reason similar algorithms have been proposed to other biometric traits [70]. The Gabor functions are Gaussian-modulated complex exponential functions that provide the best theoretical trade-off between spatial resolution and frequency resolution [71]. This method encodes the iris texture pattern by convolving the unwrapped iris image (see low part of Fig. 2.1(d)) with a Gabor wavelet or filter.

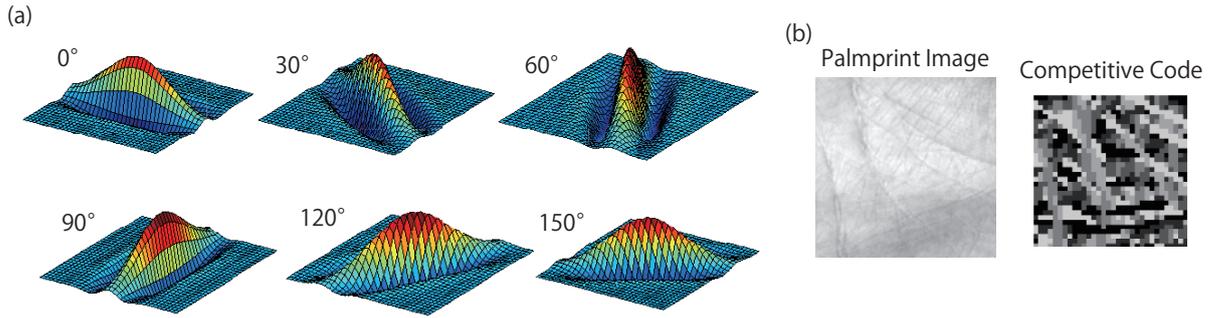


Figure 2.5: Competitive code example. (a) Elliptical Gaussian filters, (b) Palmprint image and its competitive code.

The Gabor function for polar coordinates of unwrapped image is

$$G(\rho, \varphi, \omega, \sigma, \beta) = \exp \left[-\pi \left(\frac{\rho^2}{\sigma^2} + \frac{\varphi^2}{\beta^2} \right) \right] \exp(j\omega\varphi), \quad (2.2)$$

where ρ is the radial coordinate and φ is the angular coordinate.

Let $I(\rho, \varphi)$ be the polar-coordinate normalized iris image, then the filtered image h is

$$h(\rho, \varphi) = h_{\text{real}}(\rho, \varphi) + jh_{\text{imag}}(\rho, \varphi) = \int_{\rho} \int_{\varphi} G(\rho, \varphi, \omega, \sigma, \beta) I(\rho, \varphi) \rho d\rho d\varphi \quad (2.3)$$

The IrisCode consists of the phase encoding of $h(\rho, \varphi)$ into 2 bits denoted as $C_{I,Re}$ and

$C_{I,Im}$ as follows:

$$C_{I,Re} = \begin{cases} 1, & \text{if } h_{\text{real}}(\rho, \varphi) > 0, \\ 0, & \text{if } h_{\text{real}}(\rho, \varphi) < 0, \end{cases} \quad (2.4)$$

$$C_{I,Im} = \begin{cases} 1, & \text{if } h_{\text{imag}}(\rho, \varphi) > 0, \\ 0, & \text{if } h_{\text{imag}}(\rho, \varphi) < 0. \end{cases} \quad (2.5)$$

This is to quantize the phase information into four different levels, one for each quadrant of the complex plane. Thus, an IrisCode template consists of an array of two bits values for 32×32 coordinates of (ρ, φ) .

The normalized Hamming distance between two IrisCode templates is used as a measure of dissimilarity between two irides $C_1(i)$ and $C_2(i)$ where $i = 1, \dots, 2048$. The Hamming distance is

$$D_{\text{Hamming}} = \frac{1}{2048} \sum_{i=1}^{2048} C_1(i) \oplus C_2(i), \quad (2.6)$$

where \oplus denotes the XOR operation. This measure is usually computed with a mask to disregard noisy or occluded regions. A coding method similar to IrisCode has been proposed for palmprint recognition.

The following equation correspond to the elliptical Gabor function:

$$G(x, y, \theta, \omega, \sigma, \beta) = \frac{1}{2\pi\sigma\beta} \exp \left[-\pi \left(\frac{x'^2}{\sigma^2} + \frac{y'^2}{\beta^2} \right) \right] \exp(j\omega x'), \quad (2.7)$$

and the proxy coordinates x' and y' are given by:

$$x' = (x - x_0) \cos(\theta) + (y - y_0) \sin(\theta), \quad (2.8)$$

$$y' = -(x - x_0) \sin(\theta) + (y - y_0) \cos(\theta), \quad (2.9)$$

where x_0 and y_0 are the coordinates for the center of the filter (usually $(0, 0)$).

Algorithms inspired by IrisCode utilize multiple Gabor or Gabor-like filters with different orientation, scales and frequencies. The palmprint algorithm Competitive Code proposed by Kong et al. 2004 [8], for example, uses the negative real part of a Gabor function to produce N_G filters.

$$G_i^{\text{comp}}(n_1, n_2) = \frac{1}{2\pi\sigma^2} - \text{real}(G(n_1, n_2, \theta_i, \omega, \sigma, \beta)) \text{ for } i = 1, \dots, N_G \quad (2.10)$$

where $\theta_i = (i - 1)\pi/N_G$ and $\text{real}(z)$ returns the real part of its argument z . Usual values for N_G are six or twelve directions. Competitive Code uses the *winner takes all* rule to encode the dominant direction. Let $I(n_1, n_2)$ be a palmprint image, the coding $C_C(n_1, n_2)$ is

$$C_C(n_1, n_2) = \arg \min_i [I(n_1, n_2) * G_i^{\text{comp}}(n_1, n_2)], \quad (2.11)$$

Table 2.2: Efficient 3-bit direction encoding for Competitive Code templates

Index i	Direction	3-bit code
1	0°	000
2	30°	001
3	60°	011
4	90°	111
5	120°	110
6	150°	100

where $*$ denotes 2D convolution operation.

The distance metric for Competitive Code is the circular difference between the encoded directions. This can be computed efficiently as a Hamming distance for six directions encoded by 3-bits as presented in Table 2.2.

Double-Orientation Code (DOC) is a recent variation of the Competitive Code that was proposed by Fei et al. 2016 [72] to address the sensitiveness of Competitive Code to noise and small rotations. DOC templates consist of two orientation templates, one with the strongest direction and another with the second strongest direction. It employs a non-linear operator to determine the distance between codes.

Another coding algorithm that has shown good performance is the Ordinal Code proposed by Sun et al. 2005 [9]. The ordinal filters are defined as follows:

$$G_i^{\text{ordi}}(n_1, n_2) = \text{real}(G(n_1, n_2, \theta_i, \omega, \sigma, \beta)) - \text{real}(G(n_1, n_2, \theta_i + \pi/2, \omega, \sigma, \beta)), \quad (2.12)$$

where $i = 1, \dots, N_G$, $\theta_i = (i - 1)\pi/N_G$ and $N_G = 3$. The trait (e.g. palmprint) image is convoluted by each filter. An Ordinal Code template consists of the sign of each convoluted image. Ordinal Code templates are compared by Hamming distance between the codes. We employ Competitive Code, Ordinal Codes and DOC in Ch. 4. Also, we employ Ordinal Code for iris recognition in Appendix A.

2.4.2 LBP

The LBP operator describes the small scale appearance of images, and it was initially proposed for texture classification [73]. The operator assigns a label to every pixel of an image by comparing the pixel to its neighborhood. Each comparison determines a bit of the label. Early binary patterns use a simple neighborhood of 9×9 -pixels, but interpolation can be used to define a circular neighborhood [10]. Fig. 2.6(a) shows the case

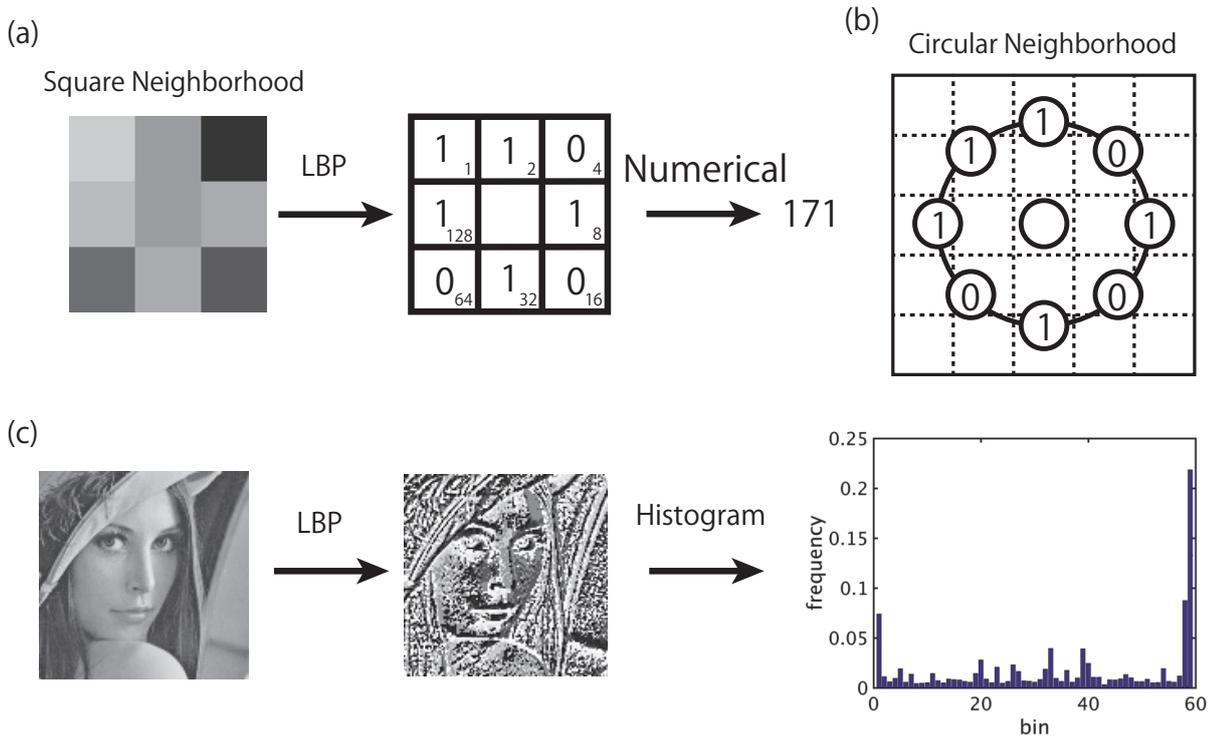


Figure 2.6: Local Binary Pattern. (a) Local binary pattern extraction from a pixel with a square neighborhood, (b) circular neighborhood, (c) LBP and histogram extraction example.

for simple neighborhood and Fig. 2.6(b) depicts the interpolation locations for the circular neighborhood. An example of a binary pattern is depicted in Fig. 2.6(c).

After LBP encoding of each pixel, the binary pattern of the trait image is divided into several blocks and a histogram of local binary patterns in each window is computed [31]. The number of bins in the histogram is 8 for the basic LBP. The final feature vector consists of the collection of histograms of all the individual blocks. Finally, two face-images can be matched by computing the similarity (or distance) between their histogram vectors [31]. Fig. 2.6(c) displays a histogram for the lena image. We employ LBP for periocular recognition in Ch. 5.

2.4.3 SIFT

Scale-Invariant Feature Transform (SIFT) was proposed by D. Lowe in 2004 [11]. Since then, it has become one of the most popular schemes used in object recognition [31]. SIFT features consist in a key-point coordinate, a governing direction, a scale or size value, and a Histogram of Gradients (HOG) as feature descriptor. The computation of SIFT features consists of two steps: key-point extraction and descriptor calculation. The key-point, direction and scale define the frame or area from where the histogram is computed.

In Fig. 2.7, we illustrate an example of SIFT features. The computation and matching of SIFT features is described in the following 5 steps:

- **Scale space construction:** Convolve the preprocessed image with a filter bank of Gaussian functions with different scales. This decomposition is known as the Gaussian scale space. Maxima at the scale space have a wide range of projection over affine transformations for which some degree of invariance is expected. For this reason, these maxima are used as key-points.
- **Key-point localization:** Estimate the key-points from the difference of neighborhood planes of the Gaussian scale space, i.e., the Difference of Gaussians (DoG) of the image.
- **Orientation assignment:** For every key-point, take a patch centered at it and compute a HOG by grouping the orientation of gradients into 10 bins. The direction with the maximum bin is assigned as the dominant direction.
- **Descriptor computation:** For every key-point, compute an HOG using a frame normalized at the dominant orientation. This HOG is computed by grouping orientation gradients into 8 bins.
- **Feature matching** The key-points provide tolerance against pose variations, but their number can be arbitrarily large in the order of hundreds. Thus, finding the correspondences between the useful key-points from two different images is a challenging task [31]. The strategy in computer vision is to employ a random sample consensus (RAMSAC) approach. Nonetheless, in biometrics, preprocessing provides a region of interest stable enough that key-point correspondence problem can be simplified by implementing simple constraints [74].

We employ methods based on SIFT matching in Ch. 5 for periocular recognition and in Appendix A for palmprint recognition.

2.5 Recognition performance metrics

Every biometric system is a classifier that makes predictions from biometric data. To measure the system performance is to estimate how often these predictions are correct or wrong. Thus, a performance metric measures either the rate of success or the rate of errors that the system makes. Ultimately, the prediction depends on the matching score provided by the matcher. As explained above, the matching score is an assessment of the similarity

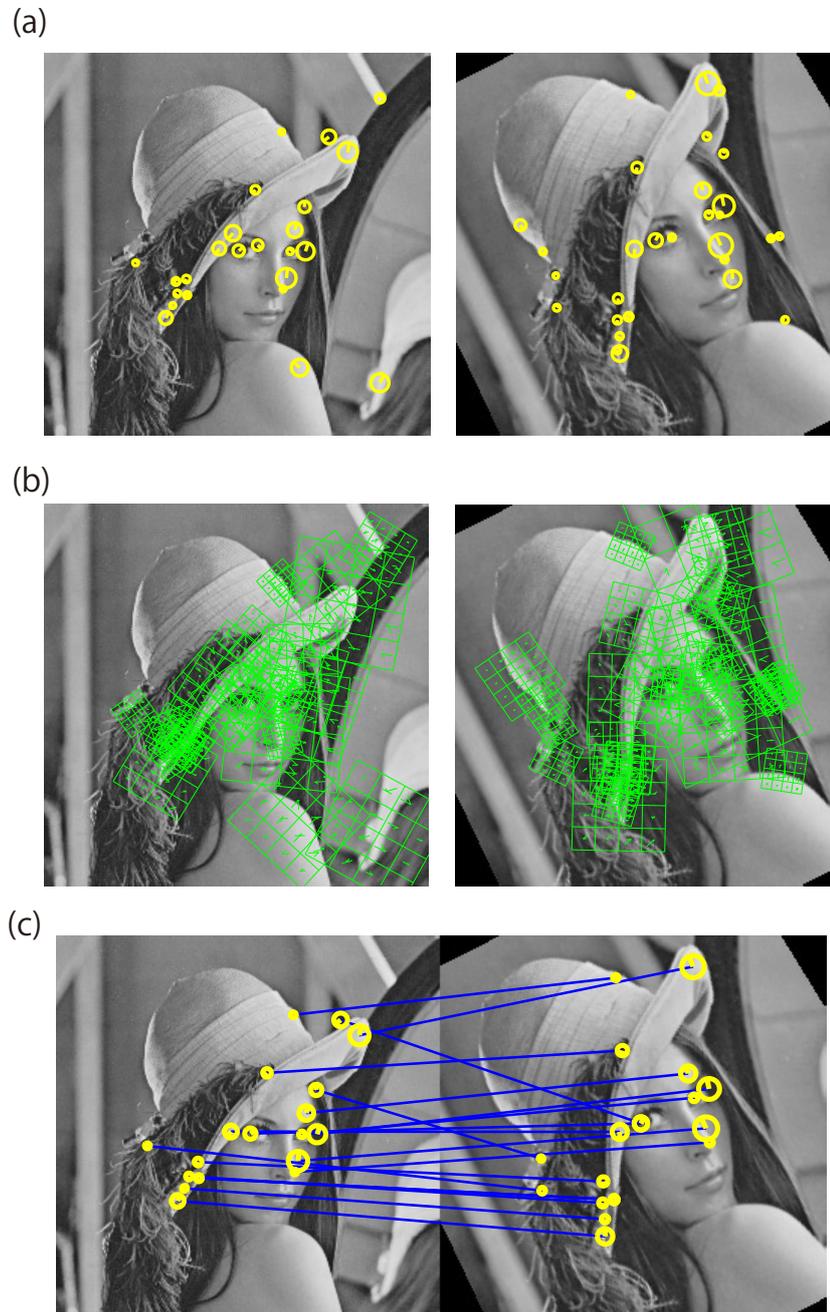


Figure 2.7: Matching of SIFT features. a) Yellow markers indicate key-points, their scale and their main direction, b) histograms of gradients depicted in green color, c) correspondences obtained from feature matching.

between a probe image and a registered image. There are two classes of matching scores, *genuine* class and *imposter* class. Genuine scores are generated by matching trait images of the same person, i.e., genuine image pair. Impostor scores are generated by matching trait images of different persons. For high rate of success, it is desired small *intra-class* variations, i.e., small variance of score distributions and large *inter-class* variations, i.e., large distance between the means of the genuine and impostor distributions. Basically, every performance metric measures these qualities of the classes' distributions.

In verification mode, the system predicts if the score belongs to the genuine class or to the impostor class. In identification mode, the predicts the identity or possible identities of the test user. This is to predict which scores are likely to belong to the genuine class. Due to this difference between verification and identification, different metrics are usually dedicated to each recognition mode.

2.5.1 Verification metrics

In the case of verification mode, the decision module predicts whether the images are from the same trait(s) of a user by comparing the score with an operational threshold value. If the score¹ is greater than the threshold value, the system assumes a *match*, and it *accepts* the user. Otherwise, the system assumes a *non-match*, and it *rejects* the user. In this way, the threshold value determines how strict or forgiving is the system to accept a user. Four cases are possible when the decision module receives a score s and made a decision based on a threshold T .

- Hit[75], or match: the score is greater than the threshold and is genuine, $s > T, s \in \textit{genuine}$.
- False alarm[75], or false match: the score is greater than the threshold and is an impostor, $s > T, s \in \textit{imposter}$.
- Miss[75], or false non-match: the score is lower than the threshold and is genuine, $s < T, s \in \textit{genuine}$.
- Correct rejection [75], or non-match: the score is lower than the threshold and is an impostor, $s < T, s \in \textit{imposter}$.

Notice that false matches and false non-matches are two kind of errors that the decision module can make. These errors go by other names: false match is also known as False Accept (FA) or Type I error; false non-match is also known as False Reject (FR) errors or

¹similarity score

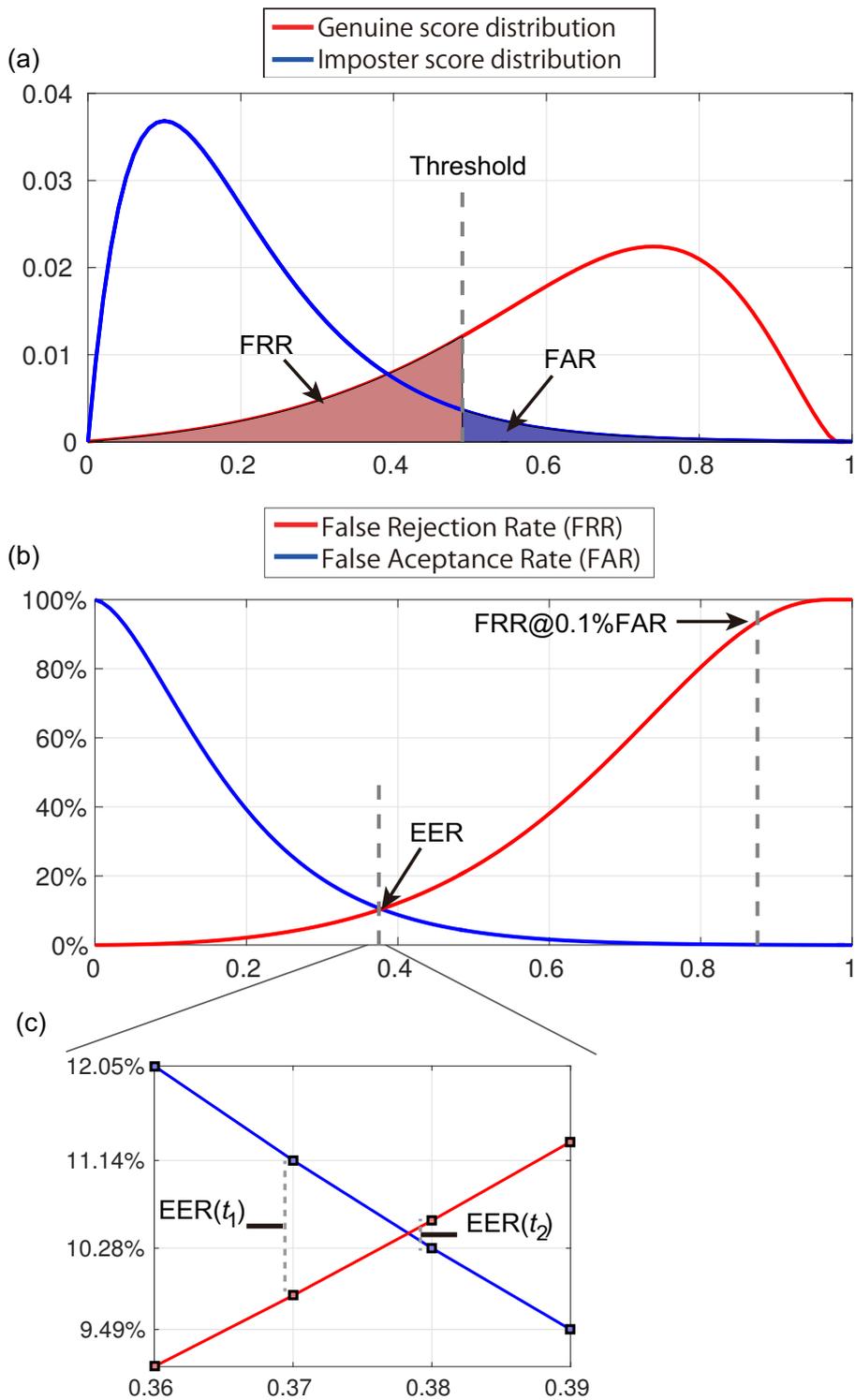


Figure 2.8: Probabilistic distribution of matching scores for performance assessment. (a) Score distributions, (b) FAR and FRR, (c) Operative values for EER estimation.

Type II errors. Fig. 2.8(a) shows an example of the score distribution. Notice the areas between the distributions and threshold, their sizes are the expectations for the two types of error.

The rate or probability estimation of each error goes also by different names. The rate of FA is known as False Match Rate (FMR), False Accept Rate and False Acceptance Rate (FAR). The rate of FR is known as False Non-Match Rate (FNMR), False Reject Rate and False Rejection Rate (FRR). The empirical estimation of FAR and FRR are given by

$$\text{FAR}(\%) = \frac{1}{N_{\text{imposter}}} \sum_{s_i \in \text{Imposter}} I(s_i - T) \times 100\%, \quad (2.13)$$

$$\text{FRR}(\%) = \frac{1}{N_{\text{genuine}}} \sum_{s_i \in \text{Genuine}} I(T - s_i) \times 100\%, \quad (2.14)$$

$$\text{where } I(x) = \begin{cases} 1, & \text{if } x \geq 0, \\ 0, & \text{if } x < 0. \end{cases} \quad (2.15)$$

In Eq. 2.13 and Eq. 2.14, N_{genuine} and N_{imposter} denote the number of genuine scores and the number of imposter scores, respectively.

Although these terms are widely used interchangeably in the literature, strictly speaking FMR and FNMR are not exactly the same as FAR and FRR. FAR and FRR consider the Fail To Enroll (FTE) error rate and the Fail To Compare (FTC) error rate (see Ch. 5 in [76] and Ch. 2 in [14] for details). We do not take into account such system-level errors in our evaluation but we still make use of the FAR and FRR convention. Fig. 2.8(b) shows the FAR and FRR for the score distributions shown in Fig. 2.8(a). There is a trade-off between these two error rates since we cannot set a threshold that minimize one kind of error without increasing the other.

Matching scores can have different meanings and distributions for different systems, so the measures of FRR or FAR for a given threshold are not comparable. Nonetheless, the trade-off between FAR and FRR is a useful characteristic of the system. This trade-off is described by the Receiver Operating Characteristic (ROC) curve shown in Fig. 2.9(a). The important benefit of the ROC curve is that it distinguishes between discriminability and decision bias. The former is the inherent property of the recognition system that we want to describe and the latter is due to the operation setting [75]. Characteristic values of the ROC curve are the Equal Error Rate (EER) and $\text{FRR}@0.1\%\text{FAR}$. The former is the value at which FAR and FRR are equal, the later is the FRR value when FAR is 0.1%. These two metrics are appropriate for comparative evaluation of verification performances. In general, there is not a value of T for which FAR and FRR are equal; so, for operative estimation of EER, we employ the consecutive threshold values $T = t_1$ and $T = t_2$ where $(\text{FAR}(t_1) - \text{FRR}(t_1)) > 0 > (\text{FAR}(t_2) - \text{FRR}(t_2))$ (shown in Fig. 2.8(c)). One option is

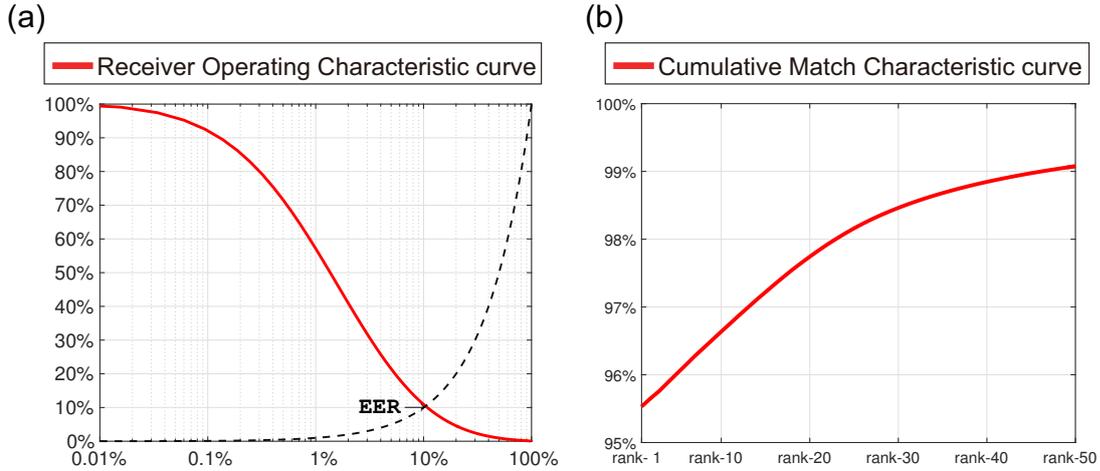


Figure 2.9: Curves plot for performance evaluation of biometric systems. (a) ROC curve, (b) CMC curve.

to estimate EER by linear interpolation. Another option is the estimator introduced by Maio et al. [77] that chooses between two estimations of EER, EER at t_1 and t_2 . These estimations and the choice for EER are given as follows:

$$\text{EER}(t_1) = (\text{FAR}(t_1) + \text{FRR}(t_1))/2, \quad (2.16)$$

$$\text{EER}(t_2) = (\text{FAR}(t_2) + \text{FRR}(t_2))/2, \quad (2.17)$$

$$\text{EER} = \begin{cases} \text{EER}(t_1), & \text{if } \text{EER}(t_1) < \text{EER}(t_2), \\ \text{EER}(t_2), & \text{if } \text{EER}(t_2) < \text{EER}(t_1). \end{cases} \quad (2.18)$$

2.5.2 Identification metrics

In the case of identification mode, the decision module provides the identity of the top matches. Whereas in verification we have genuine class and importer class scores, in identification we have the identification rank. The identification rank is defined as the rank of user's correct identity in the top matches [31]. The identification performance is measured by the rank- n identification rate which indicates the frequency of a correct identification within the top rank- n . The rank- n identification rate for different values of n is shown in the Cumulative Match Characteristic (CMC) curve. It displays the chance of a correct identification within the top ranked match results. An example of a CMC curve is shown in Fig. 2.9(b).

2.6 Summary

This chapter reviewed the attributes of biometric traits required for recognition applications. It covered the modules of a biometric system, and recognition methods that employ, for image matching, Gabor-based coding, LBP and SIFT features. These methods are applicable to various biometric traits for recognition. We presented the performance metrics to be used in this dissertation. These include verification metrics, e.g., equal error rate, and identification metrics. This chapter laid the foundation on biometric recognition necessary for the discussions in the rest of this dissertation. The next chapter covers the fundamentals on phase-based image matching and describes a baseline recognition method that is the center of discussion for Ch. 4 and 5.

3

Phase-based image matching for biometric recognition

In this chapter, we cover the fundamentals of phase-based image matching and introduce biometric recognition using phase-based correspondence matching. We start with a review of cross-correlation and move to Phase-Only Correlation (POC) in Sec. 3.2. We cover POC with spectral weighting and Band-Limited POC in Sec. 3.3 and Sec. 3.4, respectively. Then, we observe the effect of these techniques under deformations and noise. In Sec. 3.8, we describe phase-based correspondence matching which employs a coarse-to-fine strategy for accurate and robust image matching. This recognition method is the center of the discussion for this dissertation.

Hereafter, the symbol $\lfloor \cdot \rfloor$ means rounding towards minus infinite, and the symbol $\lceil \cdot \rceil$ means rounding towards infinite.

3.1 Cross-correlation

Let us consider the problem of localizing a target $f(n_1, n_2)$, i.e., pattern image, inside an image $J(n_1, n_2)$. Let the image $J(n_1, n_2)$ have a size $height \times width$ and let the pattern image $f(n_1, n_2)$ have a smaller size $L_1 \times L_2$, where $height > L_1$ and $width > L_2$. We denote as $f_{\text{pad}}(n_1, n_2)$ a zero-padded image of $f(n_1, n_2)$. The support of coordinates for $f_{\text{pad}}(n_1, n_2)$ is given by $n_1 = -L_1^-, \dots, L_1^+$ and $n_2 = -L_2^-, \dots, L_2^+$ where $L_1^- = \lceil (L_1 - 1)/2 \rceil$, $L_1^+ = \lfloor (L_1 - 1)/2 \rfloor$, $L_2^- = \lceil (L_2 - 1)/2 \rceil$ and $L_2^+ = \lfloor (L_2 - 1)/2 \rfloor$. The pattern can be

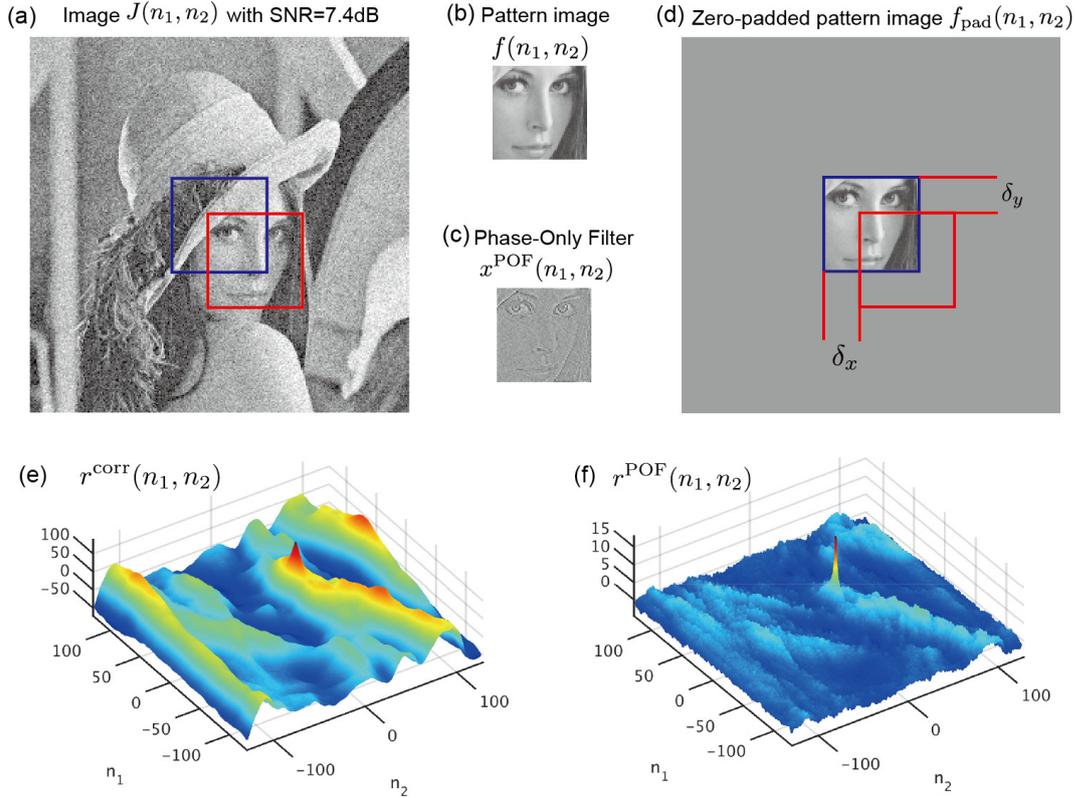


Figure 3.1: Target localization with cross-correlation function and with phase-only filtering. (a) Degraded image: red square ground true for the target location, blue square target localization on padded image. (b) Pattern image or target object. (c) Spatial pattern of POF (reversed impulse response). (d) Zero-padded pattern image and the displacement (δ_1, δ_2) necessary for alignment. (e) Cross-correlation function and (d) correlation function with POF

detected by the circular cross-correlation function

$$r^{\text{cross}}(n_1, n_2) = f_{\text{pad}}(n_1, n_2) \otimes J(n_1, n_2), \quad (3.1)$$

where \otimes denotes 2D circular cross-correlation. The reason for using circular cross-correlation instead of cross-correlation is a practical one. Circular cross-correlation can be computed efficiently by the means of the Discrete Fourier Transform (DFT) algorithm Fast Fourier Transform (FFT) (except when L_1 and L_2 are very small). The maximum value of $r^{\text{cross}}(n_1, n_2)$ is expected to be at the coordinates where $f(n_1, n_2)$ is localized in $J(n_1, n_2)$.

The maximum value itself indicates a measurement of the intensity of the pattern in $J(n_1, n_2)$. Thus, it indicates a rough measurement of similarity as well. If we consider the image as vectors, the function $r^{\text{cross}}(n_1, n_2)$ can be interpreted as an inner product were one of the image is circularly translated by (n_1, n_2) . Fig. 3.1(e) shows an example of $r^{\text{cross}}(n_1, n_2)$ for the standard *lena* image with additive Gaussian noise, Fig. 3.1(a), with pattern image $f(n_1, n_2)$ Fig. 3.1(b), and its padded version $f_{\text{pad}}(n_1, n_2)$.

As a first step towards the phase-based image matching, we consider the Phase-Only Filter (POF) proposed by Horner and Gianino in 1984 [78]. Given the 2D Discrete Fourier Transform (2D DFT) $F(k_1, k_2)$ of a pattern image $f(n_1, n_2)$, the POF, here denoted as $\overline{X^{\text{POF}}(k)}$, is

$$\overline{X^{\text{POF}}(k_1, k_2)} = \frac{\overline{F(k_1, k_2)}}{|F(k_1, k_2)|} = e^{-\theta_F(k_1, k_2)}, \quad (3.2)$$

where $\theta_F(k_1, k_2)$ is the phase component of $F(k_1, k_2)$ and $\overline{F(k_1, k_2)}$ denotes the complex conjugate of $F(k_1, k_2)$. Let $x^{\text{POF}}(n_1, n_2)$ be the reversed impulse response of $\overline{X^{\text{POF}}(k_1, k_2)}$, i.e., the POF pattern, and let $x_{\text{pad}}^{\text{POF}}$ denote its zero-padded version. Then, the filtered image can be expressed as the following correlation function:

$$r^{\text{POF}}(n_1, n_2) = x_{\text{pad}}^{\text{POF}}(n_1, n_2) \otimes J(n_1, n_2). \quad (3.3)$$

Fig. 3.1(c) shows the $x^{\text{POF}}(n_1, n_2)$ for the $x(n_1, n_2)$ shown in Fig. 3.1(b), and Fig. 3.1(f) show the respective POF correlation function $r^{\text{POF}}(n_1, n_2)$. Notice that $r^{\text{POF}}(n_1, n_2)$ has a well define peak shape. This improves over the measurement of similarity using $r^{\text{cross}}(n_1, n_2)$, but it still depends on the local intensity of $I(n_1, n_2)$.

3.2 Phase-Only Correlation (POC)

Now, let us consider the problem of aligning an image pair and measuring their similarity. Let us have two images $f(n_1, n_2)$ and $g(n_1, n_2)$ of size $L_1 \times L_2$ where the ranges of image coordinates are given by $n_1 = -L_1^-, \dots, L_1^+$ and $n_2 = -L_2^-, \dots, L_2^+$. The coordinates limits are given by $L_1^- = \lceil (L_1 - 1)/2 \rceil$, $L_1^+ = \lfloor (L_1 - 1)/2 \rfloor$, $L_2^- = \lceil (L_2 - 1)/2 \rceil$ and $L_2^+ = \lfloor (L_2 - 1)/2 \rfloor$. Let $f(n_1, n_2) = g(n_1 + \delta_1, n_2 + \delta_2)$ for $-L_1^- + \delta_1 \leq n_1 \leq L_1^+$ and $-L_2^- + \delta_2 \leq n_2 \leq L_2^+$.

The 2D Discrete Fourier Transforms (DFTs) of $f(n_1, n_2)$ and $g(n_1, n_2)$ be denoted by $F(k_1, k_2)$ and $G(k_1, k_2)$ so that

$$F(k_1, k_2) = A_F(k_1, k_2)e^{\theta_F(k_1, k_2)}, \quad (3.4)$$

$$G(k_1, k_2) = A_G(k_1, k_2)e^{\theta_G(k_1, k_2)}, \quad (3.5)$$

where $k_1 = -L_1^-, \dots, L_1^+$, $k_2 = -L_2^-, \dots, L_2^+$. The normalized (cross-)correlation spectrum or cross-phase spectrum is:

$$R(k_1, k_2) = \frac{\overline{F(k_1, k_2)}G(k_1, k_2)}{\left| \overline{F(k_1, k_2)}G(k_1, k_2) \right|}, \quad (3.6)$$

$$= e^{j(\theta_G(k_1, k_2) - \theta_F(k_1, k_2))}. \quad (3.7)$$

It is assumed that $|F(k_1, k_2)| \neq 0$ and $|G(k_1, k_2)| \neq 0$, otherwise they should be replaced with a positive constant. Notice that $(\theta_G(k_1, k_2) - \theta_F(k_1, k_2)) \approx \frac{-2\pi}{L_1 L_2} (\delta_1 k_1 + \delta_2 k_2)$. Then, the Phase-Only Correlation (POC) function $r^{POC}(n_1, n_2)$ is defined as the 2D Inverse DFT (2D IDFT) of $R(k_1, k_2)$,

$$r^{POC}(n_1, n_2) = \frac{1}{L_1 L_2} \sum_{k_1, k_2} R(k_1, k_2) W_{L_1}^{-k_1 n_1} W_{L_2}^{-k_2 n_2}, \quad (3.8)$$

where \sum_{k_1, k_2} denotes $\sum_{k_1=-L_1^-}^{L_1^+} \sum_{k_2=-L_2^-}^{L_2^+}$. The twiddle factors are given by $W_{L_1} = e^{-j \frac{2\pi}{L_1}}$ and $W_{L_2} = e^{-j \frac{2\pi}{L_2}}$. The POC function indicates a distinctive sharp peak when image blocks $f(n_1, n_2)$ and $g(n_1, n_2)$ are similar. When they are dissimilar, the peak height, denoted as α , drops significantly. The peak location indicates the translational displacement $\delta = [\delta_1, \delta_2]$ between image blocks [18]. POC can be defined using two POF since $X^{POF}(k_1, k_2)$ is just the phase component of $f(n_1, n_2)$. For this reason, POC has been referred to as Symmetric Phase-Only Matched Filtering [79].

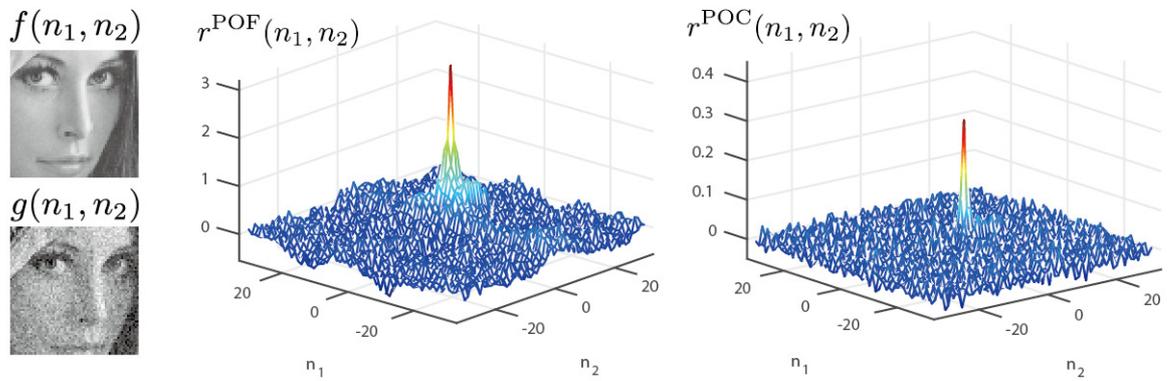
Fig. 3.2 presents a comparison between the POC function and the POF correlation function for the three following cases: Similar images with small translational displacement, similar images with large translational displacement, and dissimilar images. Note that POC improves over POF with a more defined peak shape and regular lower values in the rest of the function.

3.2.1 Windowing

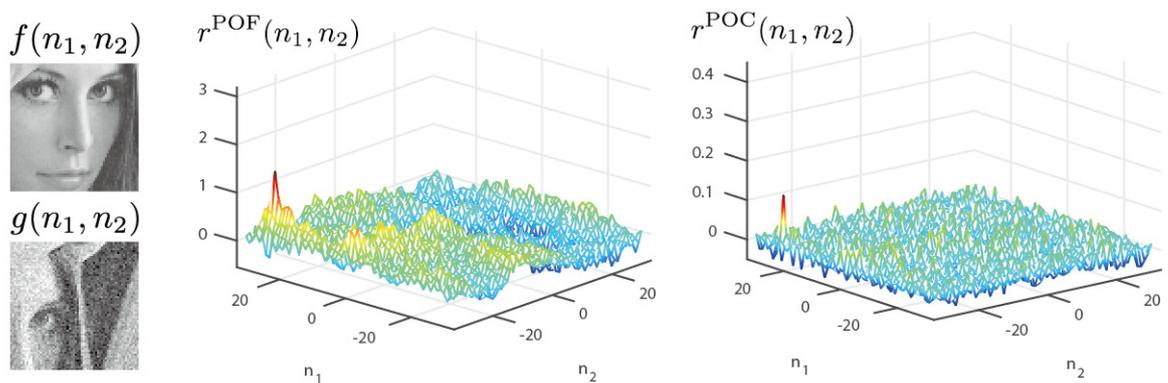
In order to have a suitable Fourier representation, $F(k_1, k_2)$ and $G(k_1, k_2)$, of $f(n_1, n_2)$ and $g(n_1, n_2)$, two considerations must be made. First, the inherent wraparound effect of DFT produces abrupt border changes. These changes are a sort of discontinuities that have a strong influence over the transformed signal. This makes it necessary to apply a window function, i.e., *windowing*, in order to smooth the borders. Fig 3.3 illustrates the abrupt changes of the wraparound effect and the smooth transition when a Hann window is applied.

Second, the level of brightness, or DC component of an image, does not contribute to the description of the image information, such as texture details or structures, because it is a constant component across the image. Thus, it should be subtracted before windowing. Let the DC components of the images, $f(n_1, n_2)$ and $g(n_1, n_2)$, be assigned as f^{DC} and g^{DC} , respectively, and let $w(n_1, n_2)$ denote a window function. If we do not subtract the DC components, the windowed function would become $(f(n_1, n_2) - f^{DC})w(n_1, n_2) + f^{DC}w(n_1, n_2)$, where we can see that the second term is independent of the image texture or content. This would cause a bias or artifact peak at $(0, 0)$ in the POC function since we would be comparing in the POC function $f^{DC}w(n_1, n_2)$ and $g^{DC}w(n_1, n_2)$ too. Therefore,

(a) Small translational displacement



(b) Large translational displacement



(c) Impostor image pair

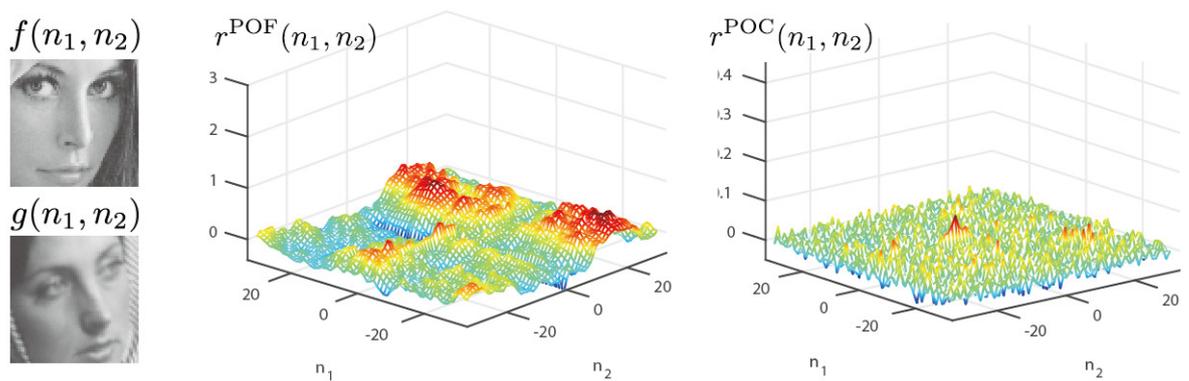


Figure 3.2: Examples of POC functions and POF correlation functions. (a) Image pair comparison with small translational displacement. (b) Image pair comparison with large translational displacement. (c) Image pair without similarity: face images of different persons (impostor case).

suitable frequency representations $F(k_1, k_2)$ and $G(k_1, k_2)$ of the blocks are given by the following expressions:

$$F(k_1, k_2) = \sum_{n_1, n_2} [f(n_1, n_2) - f^{DC}] w(n_1, n_2) W_{L_1}^{k_1 n_1} W_{L_2}^{k_2 n_2}, \quad (3.9)$$

$$G(k_1, k_2) = \sum_{n_1, n_2} [g(n_1, n_2) - g^{DC}] w(n_1, n_2) W_{L_1}^{k_1 n_1} W_{L_2}^{k_2 n_2}, \quad (3.10)$$

where $k_1 = -L_1^-, \dots, L_1^+$, $k_2 = -L_2^-, \dots, L_2^+$, and \sum_{n_1, n_2} denotes $\sum_{n_1=-L_1^-}^{L_1^+} \sum_{n_2=-L_2^-}^{L_2^+}$. Another interpretation of the windowing operation is a low-pass or smooth filtering on the Fourier domain.

In this dissertation, we employ the 2D Hann window function¹, i.e.,

$$w(n_1, n_2) = \frac{1 + \cos\left(\frac{2\pi n_1}{L_1}\right)}{2} \frac{1 + \cos\left(\frac{2\pi n_2}{L_2}\right)}{2}, \quad (3.11)$$

where $n_1 = -L_1^-, \dots, L_1^+$, $n_2 = -L_2^-, \dots, L_2^+$. However, in order to argue the choice of a window function over another, we have to delve deeper into the implications of applying a window function.

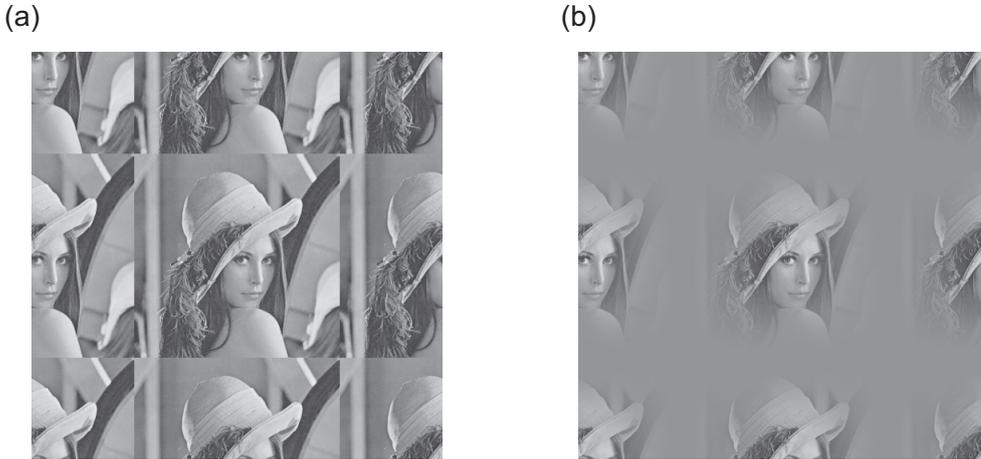


Figure 3.3: Wrap around effect of the periodic expansion of the lena image. (a) Periodic image expansion and (b) periodic image expansion after applying a Hann window. Observe the smooth transition in the image border when the window function is applied.

Observe that equations 3.9 and 3.10 can be interpreted as a transformation with non-orthogonal basis functions (or basis vectors) $b_{k_1, k_2}(n_1, n_2)$. These basis functions are formed

¹Hann window is attributed to Julius von Hann, an Australian meteorologist, but due to historical reasons this window function is widely known as Hanning window (pp. 468 in [80]).

by the Fourier basis (complex exponential functions) with an envelop given by the window function $w(n_1, n_2)$, i.e.,

$$\begin{aligned} b_{k_1, k_2}(n_1, n_2) &= w(n_1, n_2) W_{L_1}^{k_1 n_1} W_{L_2}^{k_2 n_2}, \\ &= w(n_1, n_2) \exp\left(-j2\pi\left(\frac{k_1 n_1}{L_1} + \frac{k_2 n_2}{L_2}\right)\right). \end{aligned} \quad (3.12)$$

For clarity, let us explore the characteristics of these basis functions in the 1D case, i.e.,

$$b_k(n) = w(n) \exp\left(-j2\pi\frac{kn}{L}\right), \quad (3.13)$$

where k , n , and L are the frequency variable, the spatial variable, and the length of the signal, respectively. In general, $w(n)$ is one lobe shaped with maximum at the center of the support, i.e., $n = 0$, where $n = -\lceil(L-1)/2\rceil, \dots, \lfloor(L-1)/2\rfloor$. The effect of the window function is a trade-off between the spatial resolution and the frequency resolution of the basis functions. Observe in Fig. 3.4 the effect on the spectral resolution of five window functions on two basis vectors ($k = 3$ and $k = 7$). These window functions are the rectangular window (no window case), the Tukey window, the Hann window, the Gaussian window ($\sigma = L/5$), and the Bartlett window. While the rectangular window provides high frequency resolution (observed as one frequency component in Fourier domain), the other windows have a spread around the specific frequency component. The Tukey window is a sort of middle ground between the rectangular window and Hann window due to its flat region, but it has a wide spread of the frequency component. Tukey window is better suited for long values of L when high spatial resolution is not needed, e.g., when addressing border artifacts in deconvolution problems. Like the Tukey window, the Hann window values are zero in the extremes. In the Fourier domain, the spread for the Hann window is very compact since it only extends to the direct neighbors $k \pm 1$. Therefore, the Hann window has the smallest loss in frequency resolution. The Gaussian window can have an envelope shape similar to the Hann window at the center by setting a parameter σ , but the values at the borders are not zero. Gauss window has a spread in Fourier domain that is larger than the Hann window. The values at the borders can be decreased by changing σ , but this results in a narrow envelope. The Bartlett window yields zero values at the border, but it has a discontinuous derivative at the center and a wider spread across the spectrum. We can state from these observations that the Hann window is the most suitable for addressing the border effect of the periodic expansion in image matching.

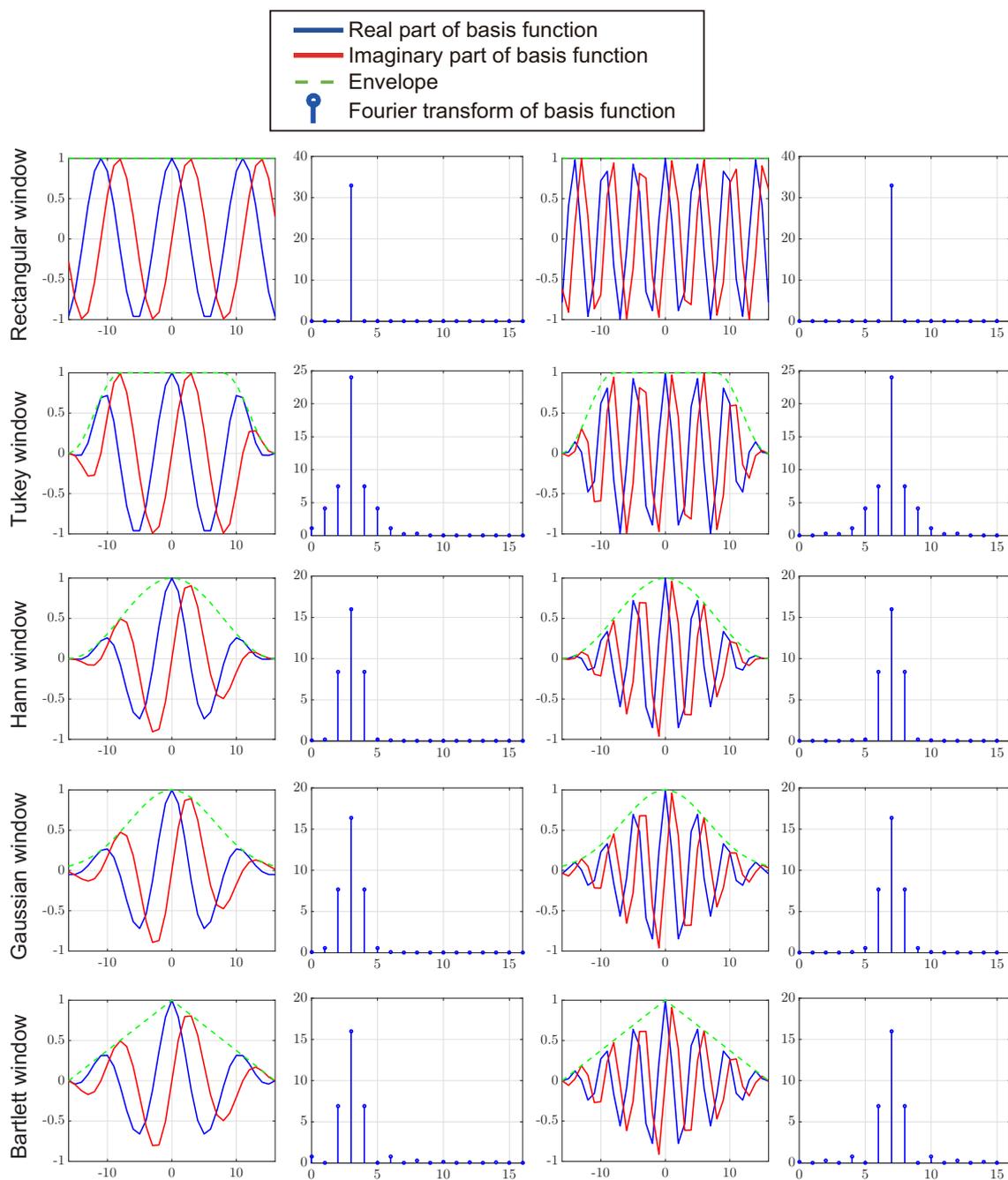


Figure 3.4: Effect of window functions on two basis functions (1D case, $L = 33$). Spatial domain: the blue line is the real part of the basis functions, the red line is the imaginary part of the basis functions, and the green line indicates the envelope/window function. Sequence data indicates the Fourier transform of the function.

3.3 Spectral weighting

For natural images, typically most of the energy is concentrated in the low spatial frequency components. As the cross-phase spectrum $R(k_1, k_2)$ is flat, there is no distinction between the contribution of the low frequency band and the high frequency band. However, the high frequency band has lower Signal-to-Noise Ratio (SNR) than the low frequency band. Thus, it affects the reliability of POC. The POC function can be very distorted (*noisy*) due to the low signal-to-noise ratio in high frequency domain. This problem has been addressed with a spectral weighting function to decrease the influence of high frequency components [20]. For this purpose, let us define the weighted cross-phase spectrum $R^{\text{POC}}(k_1, k_2)$ as follows:

$$R^{\text{POC}}(k_1, k_2) = H(k_1, k_2) \frac{\overline{F(k_1, k_2)}G(k_1, k_2)}{\left| \overline{F(k_1, k_2)}G(k_1, k_2) \right|}, \quad (3.14)$$

$$= H(k_1, k_2) e^{j\theta_G(k_1, k_2) - \theta_F(k_1, k_2)}, \quad (3.15)$$

The spectral weighting function $H(k_1, k_2)$ improves POC performance by decreasing the influence of phase ambiguity in low-energy high-frequency components. Usual choices of $H(k_1, k_2)$ are a rectangular low-pass filter function and a Gaussian function [20]. In general, a Gaussian function is expected to be closer to the SNR distribution in the spectrum for natural images. Then, we redefine the POC function $r^{\text{POC}}(n_1, n_2)$ so as to include spectral weighting as,

$$r^{\text{POC}}(n_1, n_2) = \frac{1}{N_1 N_2} \sum_{k_1, k_2} R^{\text{POC}}(k_1, k_2) W_{N_1}^{-k_1 n_1} W_{N_2}^{-k_2 n_2}. \quad (3.16)$$

Notice that for the choice $H(k_1, k_2) = 1$, Eq. 3.16 becomes Eq. 3.8.

3.4 Band-Limited Phase-Only Correlation (BLPOC)

Another strategy to deal with low SNR in the high frequencies is – to omit the high frequency components. The idea is to focus only on the essential frequency band when calculating the correlation function. This approach is called Band-Limited POC (BLPOC) [24] since it limits the frequency band to a smaller size $B_1 \times B_2$ where $B_1 < L_1$ and $B_2 < L_2$. This is different to employ a low-pass filter as a weighting function because the limited band becomes the function support in the 2D IDFT computation. Therefore, we define a band-limited cross-phase spectrum R^{BLPOC} with support $k_1 = -B_1^-, \dots, B_1^+$ and $k_2 = -B_2^-, \dots, B_2^+$ where $B_1^- = \lceil (B_1 - 1)/2 \rceil$, $B_1^+ = \lfloor (B_1 - 1)/2 \rfloor$, $B_2^- = \lceil (B_2 - 1)/2 \rceil$

and $B_2^+ = \lfloor (B_2 - 1)/2 \rfloor$. Then, the BLPOC function $r(n_1, n_2)$ is the following 2D IDFT of $R(k_1, k_2)$:

$$r(n_1, n_2) = \frac{1}{B_1 B_2} \sum'_{k_1, k_2} R(k_1, k_2) W_{B_1}^{-k_1 n_1} W_{B_2}^{-k_2 n_2}, \quad (3.17)$$

where \sum'_{k_1, k_2} denotes $\sum_{k_1=-B_1^-}^{B_1^+} \sum_{k_2=-B_2^-}^{B_2^+}$. The resulting BLPOC function $r(n_1, n_2)$ is defined for a support $n_1 = -B_1^-, \dots, B_1^+$ and $n_2 = -B_2^-, \dots, B_2^+$. The BLPOC function is particularly effective for matching a variety of biological textures and is useful for biometric authentication [81]. In the same way as with POC, the location of the correlation peak $[\delta_1^{\text{BL}}, \delta_2^{\text{BL}}]$ indicates the translational displacement between images but with a factor, $\delta_1 = \delta_1^{\text{BL}} L_1 / B_1$ and $\delta_2 = \delta_2^{\text{BL}} L_2 / B_2$. The reduction in spatial resolution of the measured displacement and similarity can be addressed using a subpixel estimation of α , and (δ_1, δ_2) .

3.5 Subpixel estimation

In this section, we present a subpixel estimation of the correlation peak's location and height [82]. We derive it from the 1D case. For this purpose, within the scope of this subsection, we define a specific meaning for the following variables: t , T , $r(n)$ and α . Let us have an underlying continuous signal f_c and two discrete sampled sections $f(n)$ and $g(n)$ of size L as follows:

$$f(n) = f_c(t)|_{t=nT}, \quad (3.18)$$

$$g(n) = f_c(t - \delta T)|_{t=nT}, \quad (3.19)$$

where $n \in \mathbb{Z}$, δ is a delay (or translational displacement) between the two signals, and T denotes a sampling interval.

Let us denote the DFT of $f(n)$ and $g(n)$ as $F(k)$ and $G(K)$, respectively. The cross-phase spectrum $R(K)$ of $f(n)$ and $g(n)$ is given by

$$R(K) = \frac{\overline{F(k)} G(K)}{\left| \overline{F(k)} G(K) \right|} \approx e^{j(\theta_G(K) - \theta_F(k))}, \quad (3.20)$$

$$\approx e^{-j \frac{2\pi k}{N} \delta}, \quad (3.21)$$

$$r^{\text{POC}}(n) \approx \frac{1}{L} \sum_{k=-\lceil (L-1)/2 \rceil}^{\lfloor (L-1)/2 \rfloor} e^{-j \frac{2\pi k}{L} \delta} e^{j \frac{2\pi}{L} n k}, \quad (3.22)$$

$$\approx \frac{1}{L} \frac{\sin(\pi(n - \delta))}{\sin\left(\frac{\pi}{L}(n - \delta)\right)}, \quad (3.23)$$

Here, $r(n)$ denotes the (1D-)POC function of $f(n)$ and $g(n)$.

We can simplify Eq. 3.23 with the following approximation:

$$r(n) = \alpha \frac{\sin(\pi(n - \delta))}{\pi(n - \delta)}, \quad (3.24)$$

$$r^{\text{BL}}(n) = \alpha \frac{\sin(\pi(n - \delta^{\text{BL}}))}{\pi(n - \delta^{\text{BL}})}, \quad (3.25)$$

where $r(n)$ and $r^{\text{BL}}(n)$ are the closed-form peak models of the 1D-POC and 1D-BLPOC functions, respectively. Eq. 3.24 is a continuous version for $r(n)$ when the argument is changed as $n \leftarrow t/T$. The continuous model is illustrated in Fig. 3.5.

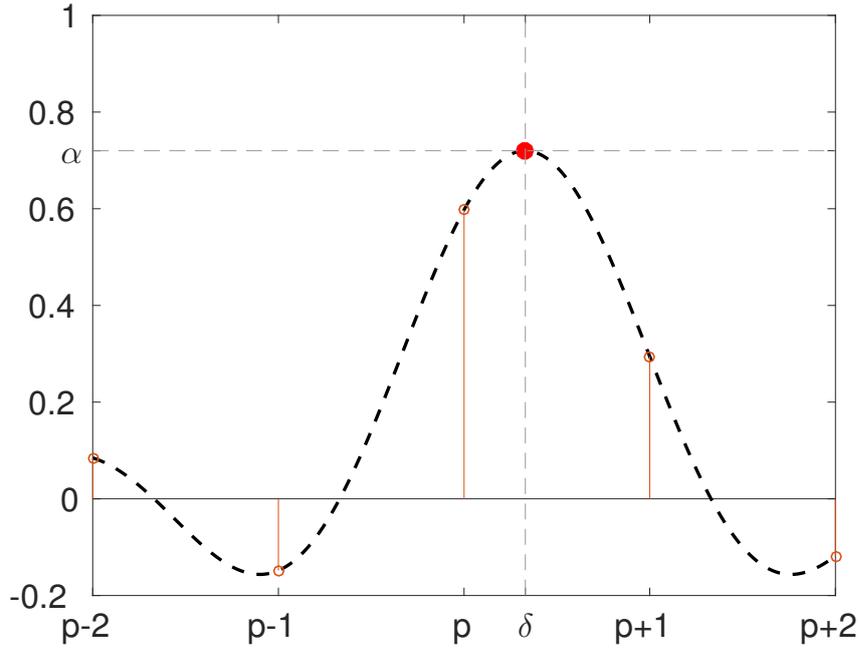


Figure 3.5: POC peak model

The approximations in Eq. 3.24 and Eq. 3.25 hold better for small values of δ , e.g., $|\delta| < 1$, so the effect of translational displacement and window can be neglected. For simplicity, let us assume that the peak value of $r(n)$ is at p which can be $\lfloor \delta \rfloor$ or $\lceil \delta \rceil$. Then, the values of $r(n)$ for n close to p are the highest and, therefore, they have the highest SNRs.

3.5.1 Three values estimation

Given the three values $r(p-1)$, $r(p)$ and $r(p+1)$, we can estimate δ and α by solving the following system of equations:

$$r(p-1)\pi(p-1-\delta) = \alpha \sin(\pi(p-1-\delta)), \quad (3.26)$$

$$r(p+1)\pi(p+1-\delta) = \alpha \sin(\pi(p+1-\delta)), \quad (3.27)$$

$$r(p)\pi(p-\delta) = \alpha \sin(\pi(p-\delta)). \quad (3.28)$$

We can solve the system by, first, adding Eq. 3.26 and Eq. 3.27, and, second, replacing the right side with Eq. 3.28 as follows

$$\pi(r(p+1) - r(p-1) - (\delta - p)(r(p+1) + r(p-1))) = -2\alpha \sin(\pi(\delta - p)), \quad (3.29)$$

$$\pi(r(p+1) - r(p-1) - (\delta - p)(r(p+1) - r(p-1))) = -2r(p)\pi(p - \delta). \quad (3.30)$$

Then, we have δ from 3.30 and α from 3.28 as follows:

$$\delta = p + \frac{r(p+1) - r(p-1)}{r(p+1) + 2r(p) + r(p-1)}, \quad (3.31)$$

$$\alpha = \frac{r(p)\pi(\delta - p)}{\sin(\pi(\delta - p))}. \quad (3.32)$$

Careful considerations must be made when using these equations regarding the denominator in Eq. 3.31 and the value of δ in Eq. 3.32. The denominator should not be close to zero. This can happen when dissimilar images are compared since the values of the POC function becomes noise.

3.5.2 Two values estimation

One simpler and robust estimation for δ , can be derived from the two highest consecutive values of $r(n)$, i.e., $r(p)$ and the greatest value among $r(p+1)$ and $r(p-1)$. In this way, Eq. 3.28 can be combined with Eq. 3.27 or Eq. 3.26, accordingly.

$$(r(p+1) + r(p))\pi(p - \delta) + r(p+1)\pi = 0, \text{ if } r(p+1) \geq r(p-1), \quad (3.33)$$

$$(r(p-1) + r(p))\pi(p - \delta) - r(p-1)\pi = 0, \text{ if } r(p+1) < r(p-1). \quad (3.34)$$

The resulting estimation of δ is as follows:

$$\delta = \begin{cases} p + \frac{r(p+1)}{r(p+1) + r(p)}, & \text{if } r(p+1) \geq r(p-1), \\ p - \frac{r(p-1)}{r(p-1) + r(p)}, & \text{if } r(p+1) < r(p-1). \end{cases} \quad (3.35)$$

Special care must be taken regarding the denominator since the second highest value ($r(p+1)$ or $r(p-1)$) must be positive. Otherwise, the estimation of α according Eq. 3.32 can be misleading.

3.5.3 Subpixel estimation for BLPOC

We can translate Eq. 3.31, Eq. 3.32 and Eq. 3.35 to the case of 1D-BLPOC $r^{\text{BL}}(n)$ by simply changing $r(n)$ and δ by $r^{\text{BL}}(n)$ and δ^{BL} , respectively.

$$\alpha = \frac{r^{\text{BL}}(p)\pi(\delta^{\text{BL}} - p)}{\sin(\pi(\delta^{\text{BL}} - p))}. \quad (3.36)$$

We have to consider the change in resolution of the BLPOC function for the displacement estimation.

$$\delta = \frac{L}{B}\delta^{\text{BL}} \quad (3.37)$$

Three values estimation:

$$\delta^{\text{BL}} = p + \frac{r^{\text{BL}}(p+1) - r^{\text{BL}}(p-1)}{r^{\text{BL}}(p+1) + 2r^{\text{BL}}(p) + r^{\text{BL}}(p-1)}, \quad (3.38)$$

Two values estimation:

$$\delta^{\text{BL}} = \begin{cases} p + \frac{r^{\text{BL}}(p+1)}{r^{\text{BL}}(p+1) + r^{\text{BL}}(p)}, & \text{if } r^{\text{BL}}(p+1) \geq r^{\text{BL}}(p-1), \\ p - \frac{r^{\text{BL}}(p-1)}{r^{\text{BL}}(p-1) + r^{\text{BL}}(p)}, & \text{if } r^{\text{BL}}(p+1) < r^{\text{BL}}(p-1), \end{cases} \quad (3.39)$$

where B is the bandwidth used for BLPOC. We refer to these estimations as the Peak Evaluation Formula PEF [82].

3.5.4 Subpixel estimation for images

For the 2D-case, let us denote the peak height estimation by α_{2D} and the displacement estimations by δ_1 and δ_2 . In order to compute δ_1 and δ_2 , we make two independent 1D estimations around the peak value $r(p_1, p_2)$, i.e., $r(n) \leftarrow r(n, p_2)$ and $r(n) \leftarrow r(p_1, n)$. Then, PEFs for α_{2D} are

POC case:

$$\alpha_{2D} = r(p_1, p_2) \frac{\pi(\delta_1 - p_1)}{\sin(\pi(\delta_1 - p_1))} \frac{\pi(\delta_2 - p_2)}{\sin(\pi(\delta_2 - p_2))}, \quad (3.40)$$

BLPOC case:

$$\alpha_{2D} = r^{\text{BL}}(p_1, p_2) \frac{\pi(\delta_1^{\text{BL}} - p_1)}{\sin(\pi(\delta_1^{\text{BL}} - p_1))} \frac{\pi(\delta_2^{\text{BL}} - p_2)}{\sin(\pi(\delta_2^{\text{BL}} - p_2))}. \quad (3.41)$$

We employ Eq. 3.41 in the similarity evaluation of local blocks in Ch. 4 and Ch. 5. An alternative to PEF is to fit a 2D model to the neighborhood of the peak. This is useful in accurate registration in stereo vision [20].

3.6 Phase feature vectors

Here, we define the phase feature vector as a step towards describing a biometric recognition algorithm using phase-based image matching. So far, we presented (BL)POC as a technique to compare two images. Now, we take a look into the question – which are the feature vectors being compared during (BL)POC matching? Each value of the $r(n_1, n_2)$ or $r^{\text{POC}}(n_1, n_2)$ corresponds to the inner product between two vectors that are derived from the original functions and their circular translated versions. The BLPOC function as defined in Eq. 3.17, for example, can be written as follows:

$$r(n_1, n_2) = \frac{1}{B_1 B_2} \mathbf{X}^\dagger \mathbf{Y}_{n_1, n_2}, \quad (3.42)$$

where \mathbf{X}^\dagger denotes the Hermitian transpose of \mathbf{X} . The vectors \mathbf{X} and \mathbf{Y}_{n_1, n_2} consist of the values for the respective functions, $X(k_1, k_2)$ and $Y(k_1, k_2)$, as follows:

$$\mathbf{X} = \begin{bmatrix} X(-B_1^-, -B_2^-) \\ \vdots \\ X(B_1^+, -B_2^-) \\ X(-B_1^-, -B_2^- + 1) \\ \vdots \\ X(B_1^+, -B_2^- + 1) \\ \vdots \\ X(B_1^+, B_2^+) \end{bmatrix}; \mathbf{Y}_{n_1, n_2} = \begin{bmatrix} Y(-B_1^-, -B_2^-) W_{B_1}^{B_1^- n_1} W_{B_2}^{B_2^- n_2} \\ \vdots \\ Y(B_1^+, -B_2^-) W_{B_1}^{-B_1^+ n_1} W_{B_2}^{B_2^- n_2} \\ Y(-B_1^-, -B_2^- + 1) W_{B_1}^{B_1^- n_1} W_{B_2}^{(B_2^- - 1)n_2} \\ \vdots \\ Y(B_1^+, -B_2^- + 1) W_{B_1}^{-B_1^+ n_1} W_{B_2}^{(B_2^- - 1)n_2} \\ \vdots \\ Y(B_1^+, B_2^+) W_{B_1}^{-B_1^+ n_1} W_{B_2}^{-B_2^+ n_2} \end{bmatrix}. \quad (3.43)$$

The definition of vectors above apply for POC by changing the support from $B_1 \times B_2$ to $L_1 \times L_2$, respectively. From Eq. 3.17, we see that the functions $X(k_1, k_2)$ and $Y(k_1, k_2)$ are

$$X(k_1, k_2) = \frac{F(k_1, k_2)}{|F(k_1, k_2)|}, \quad (3.44)$$

$$Y(k_1, k_2) = \frac{G(k_1, k_2)}{|G(k_1, k_2)|}. \quad (3.45)$$

These expressions are conceptually equivalent to the phase only filter POF described in Eq. 3.2. However, we use them here to define the *phase feature vectors* \mathbf{X} and \mathbf{Y}_{n_1, n_2} .

A characteristic of phase feature is that they are uncorrelated under translation, i.e.,

$$\frac{1}{B_1 B_2} \mathbf{Y}_{n_1, n_2}^\dagger \mathbf{Y}_{m_1, m_2} = \begin{cases} 1 & \text{if } n_1 = m_1 \text{ and } n_2 = m_2, \\ 0 & \text{otherwise.} \end{cases} \quad (3.46)$$

An equivalent property is that the auto-correlation functions of the spatial representations for \mathbf{X} or \mathbf{Y} can be expressed as a Kronecker delta.

In a biometric system, the phase features \mathbf{X} are stored in the system's gallery during enrolment and the phase features \mathbf{Y} are extracted from a test image during recognition. The number of users that can be enrolled in a system is limited by the storage capacity and the space required by phase features \mathbf{X} . Naturally, a significant storage reduction of \mathbf{X} is obtained by taking advantage of the spectrum symmetry and by including only the band required for the calculation of the BLPOC functions. Since $X(k_1, k_2)$ has a predetermined module, its phase angles $\theta_F(k_1, k_2)^1$ are better for storage. Quantization of the phase angles has been proposed, so as to effectively reduce the storage requirement [17, 83, 84].

Given N_{bits} for quantization, the function $Q^{N_{\text{bits}}}(\theta)$ that encodes the phase value of $X(k_1, k_2)$ is,

$$Q_{N_{\text{bits}}}(\theta_F(k_1, k_2)) = \left\lfloor \frac{\theta_F(k_1, k_2)2^{N_{\text{bits}}-1}}{\pi} + \frac{1}{2} \right\rfloor. \quad (3.47)$$

The quantization level $\theta_{qF}(k_1, k_2)$ for $\theta_F(k_1, k_2)$ can expressed by,

$$\theta_{qF}(k_1, k_2) = Q_{N_{\text{bits}}}(\theta_F(k_1, k_2)) \frac{2\pi}{2^{N_{\text{bits}}}}, \quad (3.48)$$

$$qX(k_1, k_2) = e^{j\theta_{qF}(k_1, k_2)}. \quad (3.49)$$

For clarity, we omit (k_1, k_2) for the angle quantities. The quantization error can be stated as follows

$$X(k_1, k_2) - qX(k_1, k_2) = 1 - \cos(\theta_F - \theta_{qF}) + j \sin(\theta_F - \theta_{qF}), \quad (3.50)$$

where j denotes the imaginary unit and $\theta_F - \theta_{qF} \in [-\pi/2^{N_{\text{bits}}}, \pi/2^{N_{\text{bits}}}]$. Then, the maximum of the magnitude of the quantization error $\text{Max}(\epsilon)$ is given by

$$\text{Max}(\epsilon) = \max(|X(k_1, k_2) - qX(k_1, k_2)|), \quad (3.51)$$

$$= \sqrt{2 - 2 \cos(\pi/2^{N_{\text{bits}}})}, \quad (3.52)$$

, where $\max()$ returns the maximum possible value. The variance of the magnitude of the quantization error $\text{Var}(\epsilon)$ is given by

$$\text{Var}(\epsilon) = \text{Var}(|X(k_1, k_2) - qX(k_1, k_2)|) \quad (3.53)$$

$$= \text{Var}(\sqrt{2 - 2 \cos(\theta_F - \theta_{qF})}), \quad (3.54)$$

$$= \frac{2^{N_{\text{bits}}+1}}{\pi} \left(\frac{\pi}{2^{N_{\text{bits}}}} - \sin\left(\frac{\pi}{2^{N_{\text{bits}}}}\right) \right), \quad (3.55)$$

$$= 2 - \frac{2^{N_{\text{bits}}+1}}{\pi} \sin\left(\frac{\pi}{2^{N_{\text{bits}}}}\right). \quad (3.56)$$

The standard deviation of the module magnitude of the quantization error $\text{Std}(\epsilon)$ is by definition $\text{Std}(\epsilon) = \sqrt{\text{Var}(\epsilon)}$. These statistics describe how the error decreases with the

Table 3.1: Relation of the quantization error with the number of bits. Maximum and variance of the magnitude of the quantization error, i.e., $\text{Max}(\epsilon)$ and $\text{Var}(\epsilon)$, respectively.

N_{bits}	1	2	3	4	5	6
$\text{Max}(\epsilon)$	1.414	0.765	0.390	0.196	0.098	0.0049
$\text{Var}(\epsilon)$	0.727	0.199	0.051	0.013	0.003	0.0008
$\text{Std}(\epsilon)$	0.853	0.447	0.226	0.113	0.057	0.0283

number of bits. Table 3.1 shows numerical values for these statistics. Particularly, the variance decreases drastically with few bits. Taking into account the standard deviation of $|X(k_1, k_2) - qX(k_1, k_2)|$ rather than the maximum value, quantization with four bits can be considered an encoding accurate enough for image matching; although, quantization with just two bits has exhibited good performance [17]. Therefore, phase quantization has little to non-detriment of the recognition performance if two or more bits are used.

Using 4-bit quantization, a phase vector \mathbf{X} of 32×32 ($B_1 \times B_2$) can be stored in $32 \times (32/2) \times 4/8 = 256$ bytes. In addition to lower storage requirements, another practical advantage of BLPOC over POC for biometric recognition is that BLPOC is faster since it requires a 2D IDFT on functions with a smaller support.

3.7 Robustness of phase-based image matching

In this section, we review a comparison of the techniques described above. We illustrate the effects of noise, and rotation and difference in scale on phase-based image matching in Fig. 3.6, Fig. 3.7 and Fig. 3.8, respectively. One way to compare the correlation functions is by comparing the peak heights. As the correlation functions are normalized, the peak heights are an indication of how robust they are to the case being evaluated.

Fig. 3.6(b) and Fig. 3.6(c) show the phase difference of cross-phase spectra for different levels of noise, Fig. 3.6(a). The linearity of the phase difference is robust against noise at low frequencies. Observe that for the worst case (SNR=1dB), there is not much difference with or without window (Fig. 3.6(d)). Fig. 3.6(e) shows that Weighted POC with window has the greatest peak heights; however, BLPOC with window has the greatest heights when PEF is used.

Fig. 3.7(b) and Fig. 3.7(c) show the phase difference of cross-phase spectra for 1° , 3° , 5° , 10° and 15° of rotation between the images, Fig. 3.7(a). Observe that windowing increases

¹Note that $\theta_X(k_1, k_2) = \theta_F(k_1, k_2)$.

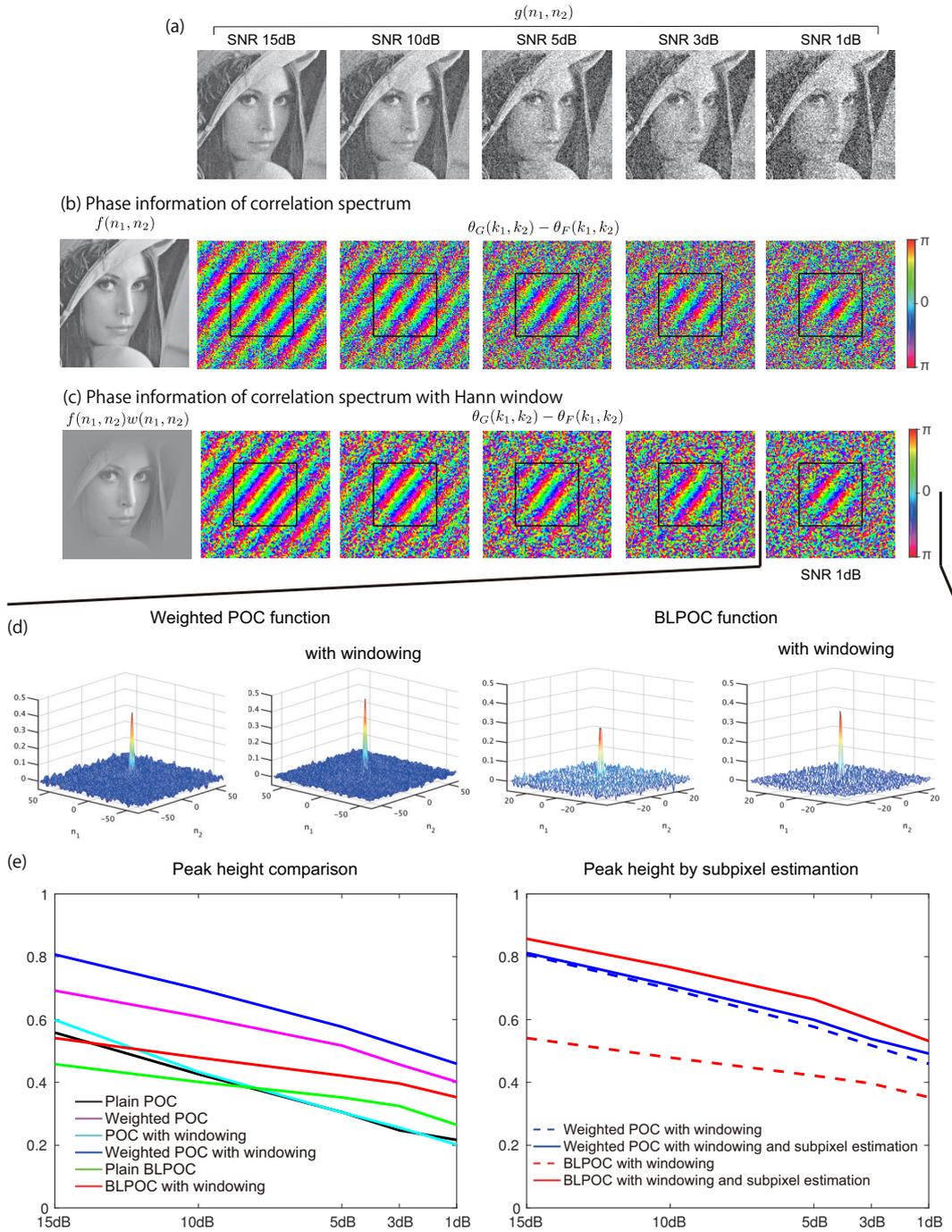


Figure 3.6: Effect of noise on POC and BLPOC functions: (a) image $g(n_1, n_2)$ for different levels of noise, (b) image $f(n_1, n_2)$ and phase difference of the image pairs, (c) image after applying Hann window $f(n_1, n_2)w(n_1, n_2)$ and phase difference of the image pairs, (d) weighted POC function and BLPOC function, (e) peak height comparison for POC and BLPOC variations.

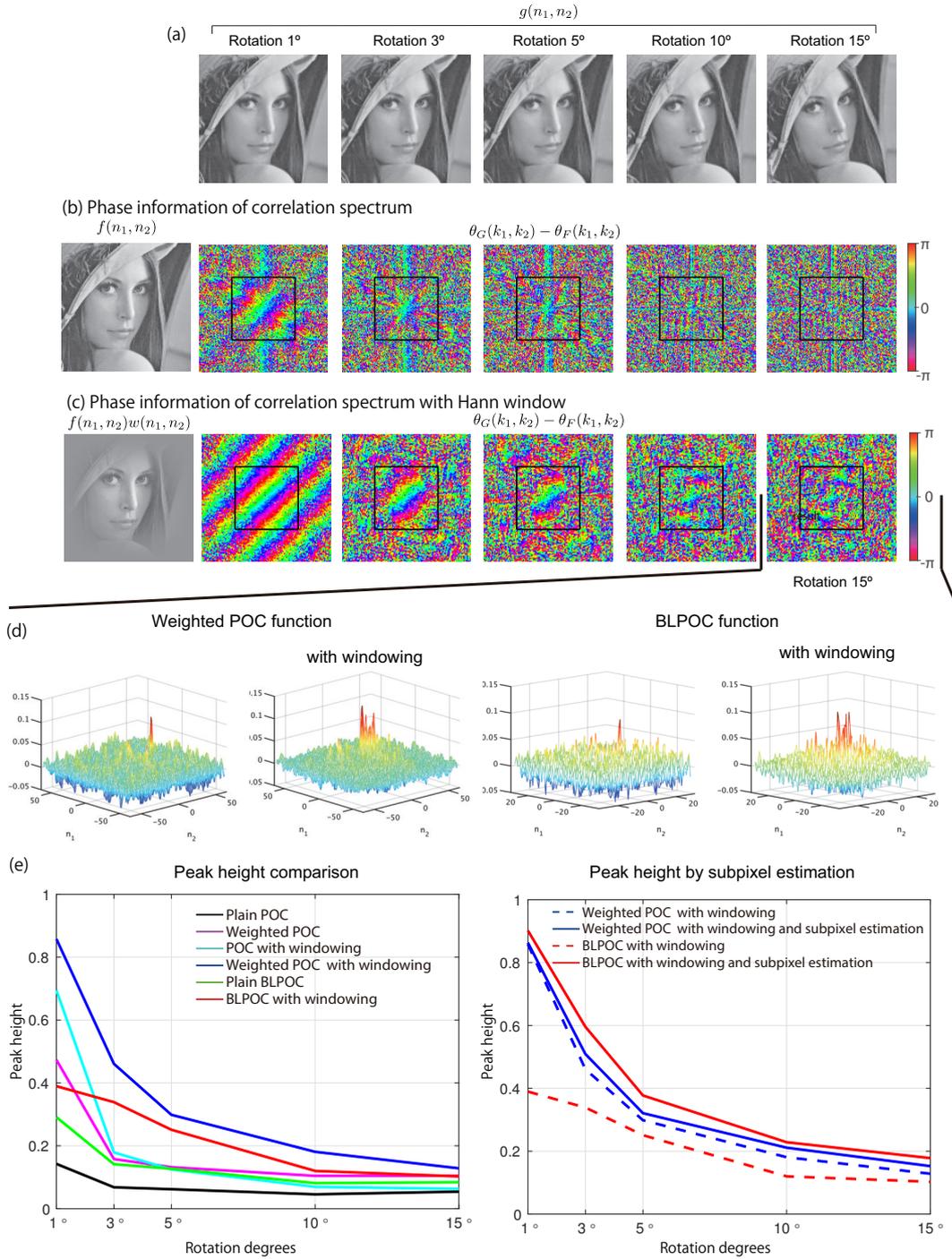


Figure 3.7: Effect of rotation on POC and BLPOC functions: (a) image $g(n_1, n_2)$ for different rotations, (b) image $f(n_1, n_2)$ and phase difference of the image pairs, (c) image after applying Hann window $f(n_1, n_2)w(n_1, n_2)$ and phase difference of the image pairs, (d) weighted POC function and BLPOC function, (e) peak height comparison for POC and BLPOC variations.

notably the linearity of the phase difference in the lower band used for BLPOC, black square in Fig. 3.7(c). This is confirmed in the peak heights presented in Fig. 3.7(e). There, we can observe that BLPOC with window and POC with window and spectral weighting are notably more robust than the other cases.

Fig. 3.8(b) and Fig. 3.8(c) show the phase difference of cross-phase spectra for the cases with the following scale factors 0.98, 0.95, 0.90, 0.85 and 0.80 in one of the images (Fig. 3.8(a)). Similar to the rotation cases, windowing increases notably the linearity of the phase difference in the lower band used for BLPOC (black square in Fig. 3.8(c)). This is confirmed in the peak heights presented in Fig. 3.8(e). Here, again, BLPOC with window and POC with window and spectral weighting are notably more robust than the other cases.

Regarding the accuracy, it is difficult to say which is better: POC or BLPOC. The tests in this section suggest similar, if not better accuracy for BLPOC. However, what actually makes one of them more accurate than the other one is the resulting SNR and its distribution. In the end, there should not be a significant difference if suitable parameters are given for both cases.

In the rest of this dissertation, we focus on BLPOC and refer to BLPOC with window and PEF as BLPOC for the sake of simplicity.

3.8 Phase-Based Correspondence Matching (PB-CM)

Let us formalize the image matching problem for biometric recognition. The problem is to accurately measure the similarity between two biometric images after preprocessing. In real situations, these images are not perfectly normalized by the preprocessing step. Thus, they vary in scale and rotation. In addition, it is natural for the biometric images to contain occlusions and global deformations. For example: in palmprint images and face images, global deformations are caused by variations in hand-pose and facial expressions, respectively; in face and periocular images, occlusions can be caused by glasses, masks, hats, scarves, etc. Accurate authentication in biometrics requires addressing these problems in the image matching task. We can deal with them by using a block-wise image matching approach based on BLPOC. The idea is to compare multiple local image blocks extracted from the given image pair since occlusions and transformations are less pronounced at the scale of small image blocks. For these comparisons to be accurate, it is necessary that local blocks are at corresponding locations in the two images. In this way, global deformations are approximated by minute translations of local block images, and the similarity between the images is assessed as the collective similarity between the block groups.

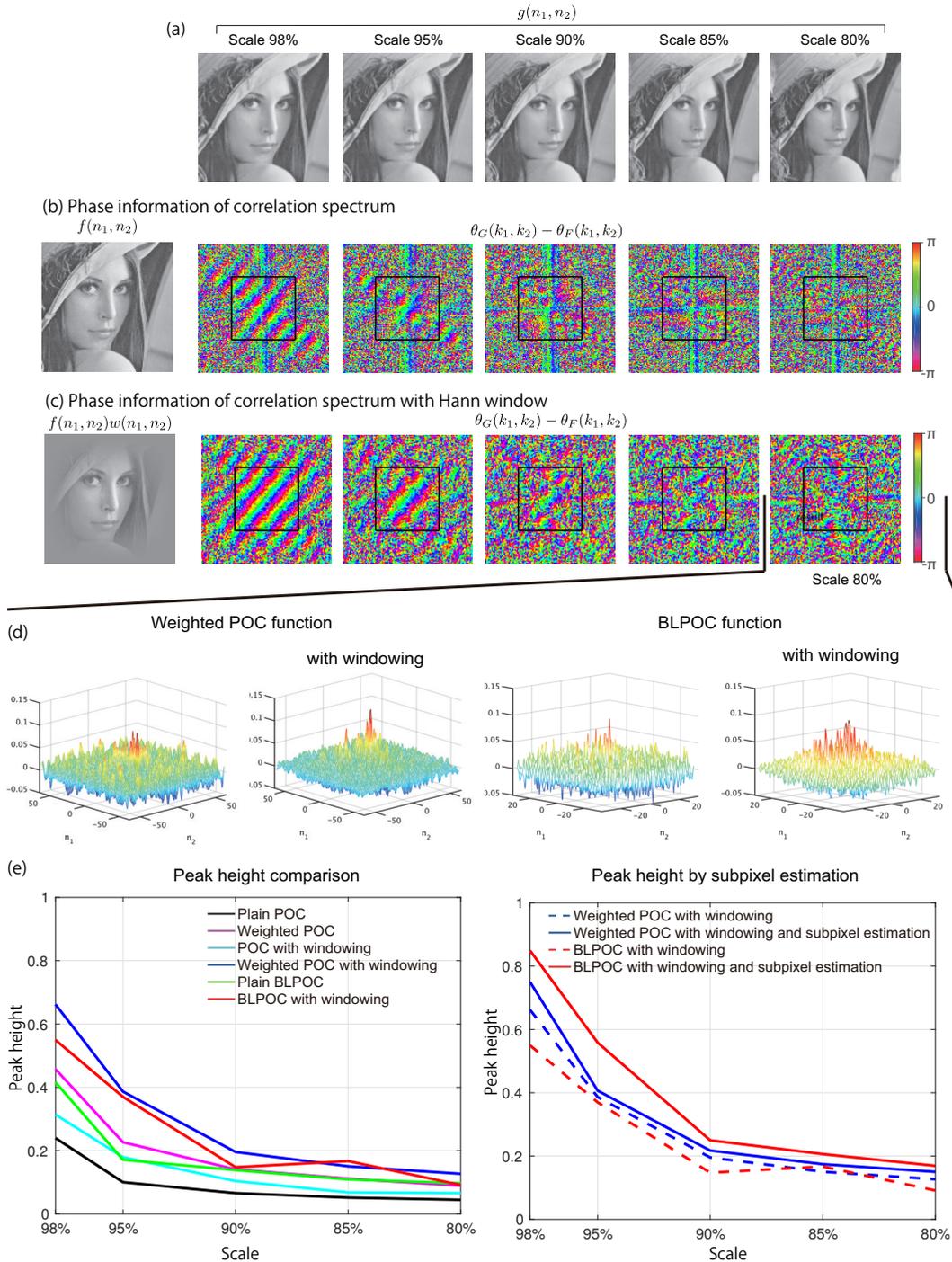


Figure 3.8: Effect of scale variation on POC and BLPOC functions: (a) image $g(n_1, n_2)$ for different scales, (b) image $f(n_1, n_2)$ and phase difference of the image pairs, (c) image after applying Hann window $f(n_1, n_2)w(n_1, n_2)$ and phase difference of the image pairs, (d) weighted POC function and BLPOC function, and (e) peak height comparison for POC and BLPOC variations.

In order to find the corresponding position for a local block using (BL)POC, it is necessary to have an initial estimate so the resulting image blocks have an overlap, e.g., examples in Fig. 3.1 and Fig. 3.2. The problem is that this estimate is not ensured by preprocessing. The naive approach is to sweep several positions until find a high correlation peak. A better approach is to consider that correspondence estimation is more reliable in low resolution, since the translation displacement is smaller. Therefore, we can combine a block-wise image matching approach with a coarse-to-fine search strategy using multi-scale image pyramids as proposed by Takita et al. 2004 [21]. Ito et. al. 2009 [28] proposed this technique for palmprint recognition and it was later refined and extended to other biometric traits by Aoyama et al. 2013 [17]. This technique is called Phase-Based Correspondence Matching (PB-CM) and it consists of two main steps: correspondence search and similarity evaluation.

3.8.1 Correspondence search

Given a pair of preprocessed images, a reference image I , which is to be registered in the system's gallery, and a probe image J , the problem considered here is to find a set of corresponding block pairs between the two images in preparation for evaluating their similarity. Let P be a set of N_b image block locations (i.e., block-center coordinates) on the reference image I , where $P = \{\mathbf{p}_1, \dots, \mathbf{p}_{N_b}\} \subseteq \mathbb{Z}^2$ determined in advance. The correspondence search problem is to find the set of corresponding block locations $Q = \{\mathbf{q}_1, \dots, \mathbf{q}_{N_b}\} \subseteq \mathbb{Z}^2$ on the probe image J . The search of Q comprises three main steps explained as follows:

(i) Generate multi-resolution image pyramids

We indicate the resolution layer of the image pyramids by the superscript l where $l = 0, \dots, l_{max}$. That is I^l for the reference image and J^l for the probe image. We generate the image pyramids by setting $I(= I^0)$ and $J(= J^0)$ and applying the following equations:

$$I^l(n_1, n_2) = \frac{1}{4} \sum_{j_1=0}^1 \sum_{j_2=0}^1 I^{l-1}(2n_1 + j_1, 2n_2 + j_2), \quad (3.57)$$

$$J^l(n_1, n_2) = \frac{1}{4} \sum_{j_1=0}^1 \sum_{j_2=0}^1 J^{l-1}(2n_1 + j_1, 2n_2 + j_2), \quad (3.58)$$

for all $l = 1, \dots, l_{max}$. In Ch. 5, image matching is improved by applying texture enhancement to the image pyramid.

(ii) Determine the set of local block locations in the reference image pyramid

Let $P^l = \{\mathbf{p}_1^l, \dots, \mathbf{p}_{N_b}^l\}$ denote the set of local block locations on the the l -th layer reference image I^l . The set of block locations $P^0 = \{\mathbf{p}_1^0, \dots, \mathbf{p}_{N_b}^0\}$ on the original reference

image I^0 are given in advance as $P^0 = P$ and $\mathbf{p}_t^0 = \mathbf{p}_t$. Then, the lower-resolution coordinates can be automatically computed as

$$\mathbf{p}_t^l = \left\lfloor \frac{1}{2^l} \mathbf{p}_t^0 \right\rfloor, \quad (3.59)$$

where $t = 1, \dots, N_b$ and $l = 1, \dots, l_{max}$.

(iii) Estimate the corresponding block locations

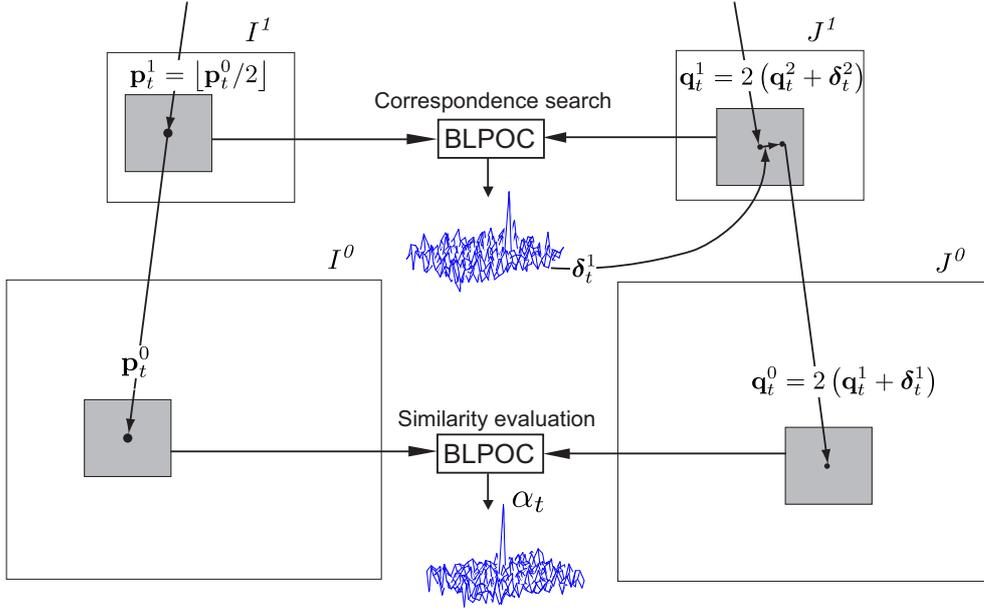


Figure 3.9: Sketch of phase-based correspondence search.

Let $Q^l = \{\mathbf{q}_1^l, \dots, \mathbf{q}_{N_b}^l\}$ denote the set of corresponding block locations on the l -th layer probe image J^l . These coordinates are estimated using a coarse-to-fine recursion. For a \mathbf{p}_t^l in the l -th layer, we find its corresponding block location $\mathbf{q}_t^l (\in Q^l)$ from the upper layer pairs $(\mathbf{p}_t^{l+1}, \mathbf{q}_t^{l+1})$, recursively, with a three steps procedure: (i) compute the BLPOC function between blocks at \mathbf{p}_t^{l+1} and \mathbf{q}_t^{l+1} , (ii) derive their displacement δ_t^{l+1} , and (iii) determine \mathbf{q}_t^l the corresponding location of \mathbf{p}_t^l by

$$\mathbf{q}_t^l = 2(\mathbf{q}_t^{l+1} + \delta_t^{l+1}), \quad (3.60)$$

$$(t = 1, \dots, N_b),$$

where $l < l_{max}$. This recurrence starts from the coarsest layer $l = l_{max}$, where we assume the simplest approximation:

$$\mathbf{q}_t^{l_{max}} = \mathbf{p}_t^{l_{max}}, \quad (3.61)$$

$$(t = 1, \dots, N_b).$$

The recurrence ends at the original resolution layer $l = l_0$. As a result, we obtain the set of coordinates $Q = Q^0$ on the probe image that corresponds to the set of coordinates P on the

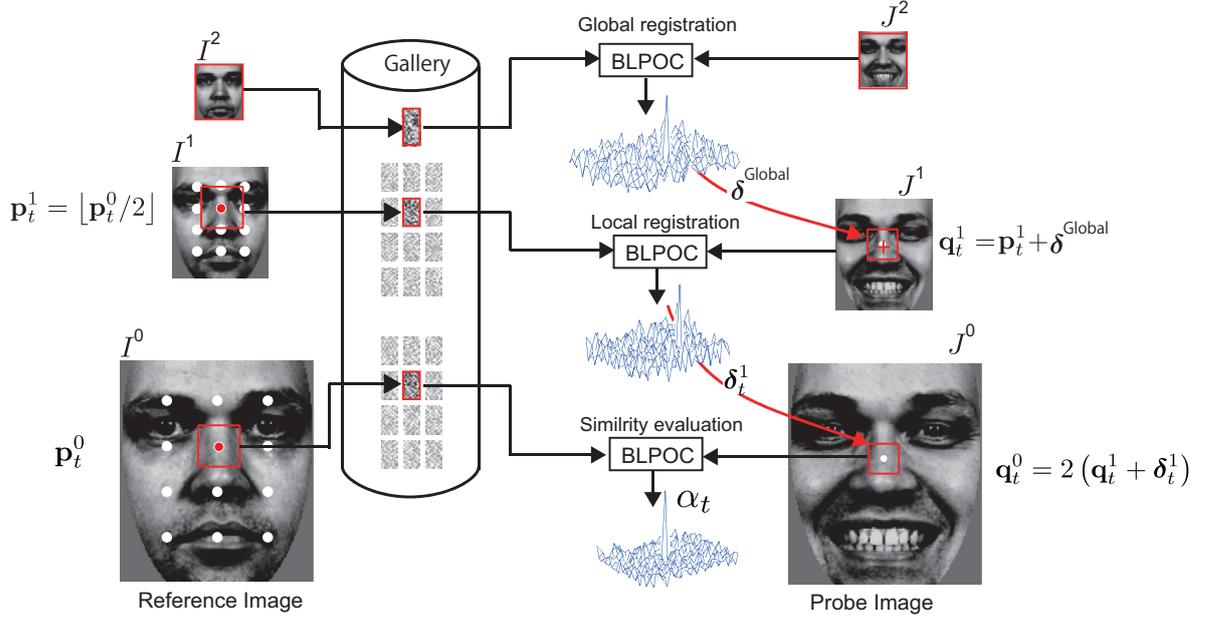


Figure 3.10: Sketch of three layers PB-CM.

reference image. Fig. 3.9 depicts the coarsest-to-fine strategy for estimating corresponding points \mathbf{q}_t^1 on J^1 and \mathbf{q}_t^0 on J^0 .

3.8.2 Similarity evaluation

Once $Q = Q^0$ is obtained, the next step is to compute the BLPOC function for the N_b block pairs $(\mathbf{p}_t^{l+1}, \mathbf{q}_t^{l+1})$. For all the block pairs $(\mathbf{p}_t^0, \mathbf{q}_t^0)$, we compute BLPOC functions and derive their block-wise similarities (i.e., BLPOC peak values α_t^0). By taking an average of these block-wise similarities, we have an overall matching score between the image pair I and J . We revise other alternatives to average in Ch. 5.3.

3.8.3 Design with local phase array

Aoyama et al. 2013 [17] proposed a design of PB-CM with three resolution layers (top layer, middle layer and bottom layer). He presented an array of hierarchical features suitable for system implementation. He also presented a similarity measure different to the average of peak heights.

The top layer (coarsest resolution) of Aoyama's design is use as a single block image to

compute a global translational displacement $\boldsymbol{\delta}^{\text{Global}}$. Then, \mathbf{q}_t^1 and \mathbf{q}_t^0 become

$$\mathbf{q}_t^1 = \mathbf{p}_t^1 + \boldsymbol{\delta}_t^{\text{Global}}, \quad (3.62)$$

$$\mathbf{q}_t^0 = 2(\mathbf{q}_t^1 + \boldsymbol{\delta}_t^1), \quad (3.63)$$

$$(t = 1, \dots, N_b). \quad (3.64)$$

Notice that Eq. (3.63) is equivalent to Eq. (3.60). Thus, the middle layer is used for a local registration and the bottom layer for similarity evaluation like in the original PB-CM. Fig. 3.10 illustrates the recognition method.

The discussion of this dissertation revolves around this algorithm, its strengths, and its drawbacks. We use Aoyama's design (although, with average of peak heights for similarity measure) in Appendix A for score-level fusion of multibiometric recognition. In Ch. 4 and Ch. 5, we consider designs with two layers which are reviewed in each chapters respectively.

3.9 Summary

In this chapter we have described the fundamentals of phase-based image matching. We described Phase-Only-Correlation (POC) and the advantages of a correlation function computed using the phase component of the Fourier transform. POC functions serve two purposes: to measure a translation-invariant similarity between images and to estimate their displacement. We reviewed two approaches to decrease the influence of low SNR in high-frequency components. These are POC with spectral weighting and Band-Limited POC (BLPOC). BLPOC has lower resolution deriving the displacement between images but this can be compensated with a subpixel estimation. We revise the robustness of the correlation functions under affine transformations and the effect of noise. We described a baseline biometric recognition method, namely, Phase-Based Correspondence Matching (PB-CM). This method yields high accuracy and robustness against deformations because it combines block-wise phase-based image matching with a coarse-to-fine strategy. This recognition method is the baseline in which this dissertation is centered. Ch. 4 addresses the computational complexity in identification, i.e., one-to-many recognition. Ch. 5 addresses the degradation in recognition accuracy that arises in periocular recognition.

4

Phase-based palmprint identification with sparse representation

This chapter presents a phase-based palmprint identification method that addresses the computational complexity of phase-based image matching in the one-to-many comparisons required for identification. Sec. 4.3 tackles the problem at the block level while Sec. 4.5 tackles the problem for full block-wise comparisons. The remaining sections are dedicated to experiments and discussions.

4.1 Introduction

In biometric recognition, the identification mode is intended to answer the question, *who is this person?*, whereas the verification mode answer, *is this person who he or she claims to be?*. Identification mode has an important role in law enforcement and security applications where suspects are tested against a black list of subjects. Equally important is its role in industrial and military applications where access is granted according to a white list of users. Another important role of the identification mode is in the enrolment stage of systems that perform verification. During enrolment, it is necessary to confirm that the user is not enrolled already (i.e., de-duplication checks).

The biometric traits commonly used for identification mode are iris, face, fingerprint and palmprint. Among these, the palmprint is a very appealing trait for commercial applications due to its balance between person recognition capability and simplicity of image data collection. The palmprint comprises a wide area, rich in stable and discriminative features

including principal lines, wrinkles and ridges. A palmprint image can be captured by using a camera under an unconstrained environment while causing little to no discomfort to users [85]. Moreover, the palm of the hand can be placed with high ease. This is because the hand is the part of the upper body with the largest feasible range of motion in space, and all tasks that require dexterity are performed using the hand. Relative to face and iris identification, palmprint identification increases privacy by granting access to users without requiring them to clear their faces from hats, scarves, eyeglasses, or burkas.

On large-scale databases, identification requires an exhaustive one-to-many comparison. Hence, the response time of the system is expected to increase significantly with large number of enrolled users, especially when computational power is limited [34]. In order to circumvent the bottleneck of an exhaustive one-to-many comparison of biometric identification systems, approaches such as coarse classification or ‘binning’ have been proposed. By binning a biometric database into classes, the overall workload for an identification attempt can be reduced [86]. However, it is difficult that the traits registered are equally distributed among these classes. For example, there is a natural classification of six category of palms, but the largest class account for 78% of the palms [87]. This is not a significant reduction of the workload. Also, binning is equivalent to combining recognition systems. Hence, classification errors might significantly increase the FRR of the system [86]. As a consequence, it is crucial to choose a suitable or fast identification method in order to deal with the computational cost derived from large number of users.

Inspired by the success in iris recognition [7], Gabor-coding based methods have been proposed and successfully applied to palmprint recognition [8, 9, 61, 72]. Representative methods describe the palm texture by encoding the orientation of line features or ordinal features across the palm surface, to have compact codes for identifying individuals, such as Competitive Code [8], Double-Orientation Code [72] and Ordinal Code [9]. Compared with the other conventional methods, these filter-coding based methods exhibit short computation times and low storage requirements [40]. These characteristics make them suitable for one-to-many comparison required in identification mode. However, for advanced palmprint recognition systems with contactless image acquisition, Gabor-coding based methods are not necessarily adequate since they lack the ability to cope with nonlinear palmprint deformations and imperfect preprocessing that results from hand pose variations.

Phase-based image matching techniques [17, 28, 84] have demonstrated efficient verification performances for palmprint images by combining hierarchical correspondence search [21] with Band-Limited Phase-Only Correlation (BLPOC) [88]. As described in Sec. 3.6 and Sec. 3.8, phase-based correspondence matching utilizes phase features extracted from local image blocks through 2D DFT. These local phase features are used to compute BLPOC functions that provide us accurate and robust similarity evaluation between images. In a

real system, local phase features of users are to be enrolled in the system's gallery (which in this chapter we refer as *phase templates*). However, these phase templates are large in size and the BLPOC functions are computationally expensive, especially for one-to-many comparisons. This results in a long computation time which is prohibitive to identification with a large number of users. We can conclude that the phase-based matching approaches have excellent palmprint recognition accuracy at the cost of computational complexity and template size.

In this chapter, we propose a palmprint identification method which employs Convolutional Sparse Coding (CSC) [89] of phase templates for computationally efficient phase-based image matching. By introducing a compact CSC representation of phase templates, we can significantly reduce the computational complexity of BLPOC. We found that the proposed method can reduce computation time and storage requirements without significant degradation of recognition performance. By means of experiments on the PolyU palmprint database [90] and the CASIA palmprint database [91], we demonstrate the effectiveness of the proposed method and its advantages over conventional methods.

This chapter comprises three main contributions:

- **(i)** A novel phase-based hierarchical block matching algorithm with reduced computational complexity using CSC of phase templates: given a compact representation of phase templates in sparse codes, this technique efficiently computes the BLPOC functions for one-to-many comparisons. It vastly reduces the number of 2D-IDFTs by applying convolution filters to phase features instead of BLPOC matching.
- **(ii)** An efficient CSC optimization algorithm that accomplishes the required compact representation of phase templates. Effective optimization is made possible with an approach based on the Matching Pursuit algorithm.
- **(iii)** An experimental evaluation of the proposed method in comparison with conventional Gabor-coding methods.

Sec. 4.2 presents the configuration of the basic setting of phase correlation used in this chapter. Sec. 4.3 introduces a basic framework for one-to-many image matching based on phase correlation with sparse representation of templates. Sec. 4.4 is dedicated to a CSC optimization algorithm. Sec. 4.5 describes an overall palmprint identification algorithm. Sec. 4.6 presents the experimental evaluation, and Sec. 4.7 summarizes this chapter.

4.2 Review of BLPOC matching

In Ch. 3, we presented the fundamentals of phase-based image matching which is effective to assess similarity in various biometric images, including palmprint images. Particularly, the block-wise approach of phase-based correspondence matching described therein is suitable to address nonlinear image distortions caused by hand pose variations. Here, we revisit the phase features described in Sec. 3.6 so as to introduce the concept of phase templates and the respective correlation functions. This is a necessary modification for the sparse representation to be proposed in the next section.

Consider two preprocessed palmprint images, a reference image I and a probe image J . We extract two block images of size $L_1 \times L_2$; one from I and one from J as in Sec. 3.8. Denote $f(n_1, n_2)$ as a block centered at a point \mathbf{p} on I , and denote $g(n_1, n_2)$ (called a *probe block*) as a block centered at a point \mathbf{q} on the reference image. The ranges of image coordinates are given by $n_1 = -L_1^-, \dots, L_1^+$ and $n_2 = -L_2^-, \dots, L_2^+$. Hereon, $f(n_1, n_2)$ is called *reference block* and $g(n_1, n_2)$ is called *probe block*. $F(k_1, k_2)$ and $G(k_1, k_2)$ are the Fourier representation given by 3.9 and 3.10.

We define the *phase template* $X(k_1, k_2)$ of a reference block $f(n_1, n_2)$ as

$$X(k_1, k_2) = H(k_1, k_2) \frac{F(k_1, k_2)}{|F(k_1, k_2)|}, \quad (4.1)$$

where $H(k_1, k_2)$ is a spectral weighting function like the one defined for Eq. 3.14. On the other hand, the *phase feature* $Y(k_1, k_2)$ of the probe block $g(n_1, n_2)$ is given by 3.45, which we repeat here for clarity,

$$Y(k_1, k_2) = \frac{G(k_1, k_2)}{|G(k_1, k_2)|}. \quad (4.2)$$

In addition to applying spectral weighting, we restricts for $X(k_1, k_2)$ and $Y(k_1, k_2)$ the range of K_1 and K_2 to a limited band $B_1 \times B_2$, where $B_1 < L_1$ and $B_2 < L_2$. We can compute a band-limited cross-phase spectrum with spectral weighting $R(k_1, k_2)$ as

$$R(k_1, k_2) = \overline{X(k_1, k_2)} Y(k_1, k_2), \quad (4.3)$$

where $k_1 = -B_1^-, \dots, B_1^+$ and $k_2 = -B_2^-, \dots, B_2^+$. Thus, the resulting BLPOC function (with spectral weighting) is

$$r(n_1, n_2) = \frac{1}{B_1 B_2} \sum'_{k_1, k_2} R(k_1, k_2) W_{B_1}^{-k_1 n_1} W_{B_2}^{-k_2 n_2}, \quad (4.4)$$

which stays equivalent to the vector form expressed in Eq. 3.42. Remember that $r(n_1, n_2)$ is defined for $n_1 = -B_1^-, \dots, B_1^+$ and $n_2 = -B_2^-, \dots, B_2^+$. As explained in Ch. 3, the

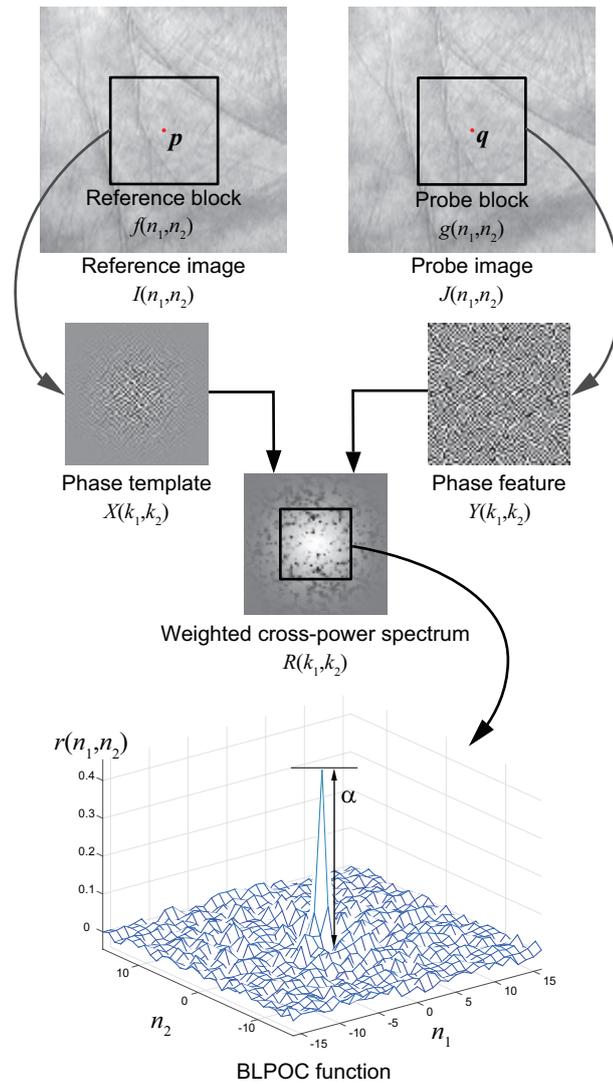


Figure 4.1: Computation flow of BLPOC function between two image blocks extracted from the same person. Note that only real parts of $X(k_1, k_2)$, $Y(k_1, k_2)$ and $R(k_1, k_2)$ are visualized in order to demonstrate the effects on spectral weighting in Fourier domain.

BLPOC function exhibits a sharp peak when the images are similar. The peak height measures the similarity between blocks, and the location of the correlation peak $[\delta_1^{\text{BL}}, \delta_2^{\text{BL}}]$ indicates the translational shift between block images but with a factor, $\delta_1 = \delta_1^{\text{BL}}L_1/B_1$ and $\delta_2 = \delta_2^{\text{BL}}L_2/B_2$. Fig. 4.1 shows a computation flow of BLPOC function between two block images extracted from the same person.

Let $x(n_1, n_2)$ and $y(n_1, n_2)$ be the (band-limited) 2D IDFTs of $X(K_1, K_2)$ and $X(K_1, K_2)$, respectively. Unlike $f(n_1, n_2)$ and $g(n_1, n_2)$, these spatial representations have size $B_1 \times B_2$, thus, for $x(n_1, n_2)$ and $y(n_1, n_2)$, $n_1 = -B_1^-, \dots, B_1^+$ and $n_2 = -B_2^-, \dots, B_2^+$. Then, note that the BLPOC function $r(n_1, n_2)$ can be expressed in the spatial domain as the following cross-correlation:

$$r(n_1, n_2) = x(n_1, n_2) \otimes y(n_1, n_2). \quad (4.5)$$

In general, however, it is unnecessary to calculate $x(n_1, n_2)$ and $y(n_1, n_2)$ explicitly since the BLPOC function $r(n_1, n_2)$ is efficiently calculated by the band-limited 2D-IDFT of $R(k_1, k_2)$ as seen in Eq. (4.4). Hereon we will only use the band-limited versions of phase templates, phase features and cross-phase spectrum, i.e., $X(k_1, k_2)$, $Y(k_1, k_2)$ and $R(k_1, k_2)$, with $k_1 = -B_1^-, \dots, B_1^+$ and $k_2 = -B_2^-, \dots, B_2^+$.

4.3 Phase-based image matching with CSC

So far, we described BLPOC matching between one probe block and one reference block. However, biometric identification, unlike verification, requires one-to-many comparison, in which the same probe block must be compared with a set of reference blocks from different reference images registered in the gallery database.

Let $Y(k_1, k_2)$ denote a phase feature extracted from the probe image J , and let us denote by $\{X_1(k_1, k_2), \dots, X_{N_u}(k_1, k_2)\}$ a collection of N_u phase templates in the frequency domain. These phase templates are extracted from the reference images $\{I_1, \dots, I_{N_u}\}$ registered into the gallery database, and N_u indicates the number of users registered in the system. The problem considered here is to find a computationally efficient way of calculating the N_u BLPOC functions $\{r_1(n_1, n_2), \dots, r_{N_u}(n_1, n_2)\}$ for evaluating the similarities between the probe's phase feature Y and the reference's phase templates $\{X_1(k_1, k_2), \dots, X_{N_u}(k_1, k_2)\}$. This section describes a basic idea of reducing the computational complexity of BLPOC functions as well as the data amount to be stored into the gallery database by introducing sparse representation of phase templates.

4.3.1 CSC representation of phase templates

Consider here the spatial-domain representations $\{x_1(n_1, n_2), \dots, x_{N_u}(n_1, n_2)\}$ of phase templates $\{X_1(k_1, k_2), \dots, X_{N_u}(k_1, k_2)\}$. They contain a form of enhanced palmprint textures in a normalized form, which are highly discriminative as revised in Sec. 3.6. Our experimental observation shows that these enhanced textures can be decomposed into a fewer number of primitive patterns, to be later used as atoms for sparse representation [89]. These primitive patterns repeat across registered phase templates at different locations. As a result, the i -th phase template $x_i(n_1, n_2)$ can be approximated by combining atoms with specific positions and intensities in the form of a sum of circular convolutions between atoms and sparse codes:

$$x_i(n_1, n_2) \approx \sum_{j=1}^{N_d} z_{i,j}(n_1, n_2) * d_j(n_1, n_2), \quad (4.6)$$

where $n_1 = -B_1^-, \dots, B_1^+$ and $n_2 = -B_2^-, \dots, B_2^+$. Note that we use $i = 1, \dots, N_u$ and $j = 1, \dots, N_d$ if not otherwise specified. Being $d_j(n_1, n_2)$ the impulse response of the j -th convolution filter (or the j -th *atom*), the set of filters $\{d_1(n_1, n_2), \dots, d_{N_d}(n_1, n_2)\}$ is called the *convolutional dictionary* for the sparse representation, where N_d indicates the number of filters. The function $z_{i,j}(n_1, n_2)$ is the j -th code for the i -th phase template. In our design, the codes are sparse so that a collection of code functions $\{z_{i,1}(n_1, n_2), \dots, z_{i,N_d}(n_1, n_2)\}$ has only N_c non-zero coefficients as

$$\sum_{j=1}^{N_d} \sum'_{n_1, n_2} \lim_{p \rightarrow 0} |z_{i,j}(n_1, n_2)|^p = \sum_{j=1}^{N_d} \|\mathbf{z}_{i,j}\|_0 = N_c, \quad (4.7)$$

where \sum'_{n_1, n_2} denotes $\sum_{n_1=-B_1^-}^{B_1^+} \sum_{n_2=-B_2^-}^{B_2^+}$. In Eq. (4.7), $\mathbf{z}_{i,j}$ is a vector whose elements are the values $z_{i,j}(n_1, n_2)$ and $\|\mathbf{z}_{i,j}\|_0$ is the L_0 -norm¹ of the vector, i.e., the number of non-zero elements of the vector [92]. The synthesis of phase templates (Eq. (4.6)) can be rewritten in the frequency domain as

$$X_i(k_1, k_2) \approx \sum_{j=1}^{N_d} Z_{i,j}(k_1, k_2) D_j(k_1, k_2), \quad (4.8)$$

where $D_j(k_1, k_2)$ and $Z_{i,j}(k_1, k_2)$ are the 2D DFTs of $d_j(n_1, n_2)$ and $z_{i,j}(n_1, n_2)$, respectively.

In the following, we describe how BLPOC functions can be represented using CSC formulas (Eqs. (4.6) and (4.8)). We call this CSC-based approximation as CSC-BLPOC.

¹Standard notations are l_0 -norm and l_1 -norm, but they can be confused with the pyramid layer l .

4.3.2 CSC-based approximation of BLPOC functions

Here we write equations in both domains: spatial domain and frequency domain for clarity purposes. Substituting the approximation Eq. (4.6) of $x_i(n_1, n_2)$ into the definition of BLPOC function Eq. (4.5) (or equivalently, Eq. (4.8) into Eq. (4.3)), we have

$$r_i(n_1, n_2) \approx \left[\sum_{j=1}^{N_d} z_{i,j}(n_1, n_2) * d_j(n_1, n_2) \right] \otimes y(n_1, n_2), \quad (4.9)$$

$$R_i(k_1, k_2) \approx \left[\sum_{j=1}^{N_d} \overline{Z_{i,j}(k_1, k_2)} D_j(k_1, k_2) \right] Y(k_1, k_2). \quad (4.10)$$

These equations can be rewritten as

$$r_i(n_1, n_2) \approx \hat{r}_i(n_1, n_2) = \sum_{j=1}^{N_d} z_{i,j}(n_1, n_2) \otimes c_j(n_1, n_2), \quad (4.11)$$

$$R_i(k_1, k_2) \approx \hat{R}_i(k_1, k_2) = \sum_{j=1}^{N_d} \overline{Z_{i,j}(k_1, k_2)} C_j(k_1, k_2), \quad (4.12)$$

where $c_j(n_1, n_2)$ and $C_j(k_1, k_2)$ are defined as

$$c_j(n_1, n_2) = d_j(n_1, n_2) \otimes y(n_1, n_2), \quad (4.13)$$

$$C_j(k_1, k_2) = \overline{D_j(k_1, k_2)} Y(k_1, k_2). \quad (4.14)$$

The function $\hat{r}_i(n_1, n_2)$ is an approximation of $r_i(n_1, n_2)$ using CSC, which we call CSC-BLPOC function. We use the symbol $\hat{\alpha}$ and $[\hat{\delta}_1^{\text{BL}}, \hat{\delta}_2^{\text{BL}}]$ to denote the height and location of the correlation peak for the CSC-BLPOC function. Since $c_j(n_1, n_2)$ is the cross-correlation function between the j -th atom $d_j(n_1, n_2)$ and $y(n_1, n_2)$, we call $c_j(n_1, n_2)$ *atomic correlation function*. As a result, we can understand Eq. (4.11) as a decomposition of a BLPOC function into a set of N_d atomic correlation functions $\{c_1(n_1, n_2), \dots, c_{N_d}(n_1, n_2)\}$.

4.3.3 Computational advantage of CSC-BLPOC

The purpose of using CSC-BLPOC is to reduce the computational cost of evaluating N_u BLPOC functions $\{r_1(n_1, n_2), \dots, r_{N_u}(n_1, n_2)\}$ corresponding to N_u phase templates $\{X_1(k_1, k_2), \dots, X_{N_u}(k_1, k_2)\}$. CSC-BLPOC functions can be computed from only N_d ($\ll N_u$) atomic correlations regardless of number of phase templates N_u . Furthermore, such computation can be very fast by taking advantage of the codes' sparsity. To confirm the above, we rewrite Eq. (4.11) as

$$\hat{r}_i(n_1, n_2) = \sum_{(j, m_1, m_2) \in \zeta_i} z_{i,j}(m_1, m_2) \tilde{c}_j(n_1 + m_1, n_2 + m_2), \quad (4.15)$$

Algorithm 1 Computation of BLPOC functions

```

for  $i \in \{1, \dots, N_u\}$  do
  (i) Compute  $R_i(k_1, k_2)$  by Eq. (4.3)
  (ii) Compute  $r_i(n_1, n_2)$  by Eq. (4.4)
end for

```

Algorithm 2 Computation of CSC-BLPOC functions

```

for  $j \in \{1, \dots, N_d\}$  do
  (i) Compute  $C_j(k_1, k_2)$  by Eq. (4.14)
  (ii) Compute  $c_j(n_1, n_2)$  in the same way as Eq. (4.4)
end for
for  $i \in \{1, \dots, N_u\}$  do
  (iii) Compute  $\hat{r}_i(n_1, n_2)$  by Eq. (4.15)
end for

```

where $\tilde{c}_j(n_1, n_2)$ is a 2D periodic extension of $c_j(n_1, n_2)$ (centered at $(0,0)$ and with the 2D period of (B_1, B_2)). Note that the set $\zeta_i = \{(j, m_1, m_2) | z_{i,j}(m_1, m_2) \neq 0\}$ contains only N_c triplets having nonzero values of $z_{i,j}(n_1, n_2)$. For better understanding of the computational advantage of CSC-BLPOC, we summarize the computational flow for both BLPOC and CSC-BLPOC cases. Given a phase feature $Y(k_1, k_2)$ extracted from a probe image and a set of phase templates $\{X_1(k_1, k_2), \dots, X_{N_u}(k_1, k_2)\}$ extracted from N_u reference images in the gallery, BLPOC and CSC-BLPOC are computed as shown in Algorithm 1 and 2, respectively.

In the case of CSC-BLPOC, we assume that the filters and the sparse codes are computed in advance through the CSC optimization algorithm described in the next section. Thus, the filters in frequency domain $\{D_1(k_1, k_2), \dots, D_{N_d}(k_1, k_2)\}$ are given, and N_u sets of sparse codes $\{z_{i,1}(n_1, n_2), \dots, z_{i,N_d}(n_1, n_2)\}$ are given as shown in Algorithm 2. Additionally, if we assume $|\hat{\delta}_1^{\text{BL}}| < M_1$ and $|\hat{\delta}_2^{\text{BL}}| < M_2$ for the BLPOC peak positions, we can further reduce the computation of $\hat{r}_i(n_1, n_2)$ to a limited search range: $n_1 = -M_1, \dots, M_1$ and $n_2 = -M_2, \dots, M_2$.

In order to clarify the computational saving by CSC-BLPOC, we give a rough comparison of the computation cost between BLPOC and CSC-BLPOC as shown in Table 4.1. For simplicity, we use a square image block size and a square search range, i.e., $B = B_1 = B_2$ and $M = M_1 = M_2$. Also, we assume that one complex multiplication corresponds to four real multiplications. We can see a clear similarity between the computation of BLPOC functions and the computation of atomic correlation functions (i.e., the step (i) and (ii) in CSC-BLPOC). If $N_d \ll N_u$, the computation of atomic correlation functions is much faster

Table 4.1: Computational cost of BLPOC functions and CSC-BLPOC functions.

Method	# of operations
BLPOC	(i) $4N_u B^2$ real multiplications
	(ii) $8N_u B^2 \log_2 B$ real multiplications
CSC-BLPOC	(i) $4N_d B^2$ real multiplications
	(ii) $8N_d B^2 \log_2 B$ real multiplications
	(iii) $N_u N_c (2M + 1)^2$ real multiplications

than the computation of BLPOC functions, where N_d is the number of atoms, N_u is the number of phase templates and the speed-up factor is N_u/N_d . The remaining step (iii) in CSC-BLPOC can be, in a sense, regarded as an overhead of CSC-based computation. For instance, assuming $N_u = 360$, $B = 32$, $M = 8$, $N_d = 16$ and $N_c = 32$, the number of real multiplications for computing the BLPOC functions is 16,220,160, while that for computing CSC-BLPOC functions is 4,050,176, and hence CSC-BLPOC is more than 4 times faster than BLPOC.

The concept of CSC-BLPOC computation can be extended to other correlation methods such as [93, 94] in the field of correlation pattern recognition [33].

4.4 CSC optimization algorithm

4.4.1 Problem statement

In this section, we propose an efficient optimization algorithm for deriving compact CSC representation of phase templates to be registered in the gallery. Let $\hat{x}_i(n_1, n_2)$ denote the CSC approximation of the spatial-domain phase template $x_i(n_1, n_2)$ given by

$$\hat{x}_i(n_1, n_2) = \sum_{j=1}^{N_d} z_{i,j}(n_1, n_2) * d_j(n_1, n_2). \quad (4.16)$$

The optimization problem considered in this section is to find adequate codes and filters to minimize reconstruction error subject to two constraints: (i) the number of non-zero coefficients in the code functions $\{z_{i,1}(n_1, n_2), \dots, z_{i,N_d}(n_1, n_2)\}$ is small constant N_c and (ii) the energy of filter's impulse response $d_j(n_1, n_2)$ is normalized to 1. Formally, the problem

is described as follows:

$$\begin{aligned}
& \arg \min_{d_j(n_1, n_2), z_{i,j}(n_1, n_2)} \sum_{i=1}^{N_u} \sum'_{n_1, n_2} |x_i(n_1, n_2) - \hat{x}_i(n_1, n_2)|^2 \\
& \text{subject to } \sum_{j=1}^{N_d} \|z_{i,j}\|_0 = N_c \text{ for } i = 1, \dots, N_u, \\
& \sum'_{n_1, n_2} |d_j(n_1, n_2)|^2 = 1 \text{ for } j = 1, \dots, N_d.
\end{aligned} \tag{4.17}$$

The sparsity constraint, i.e., $\sum_{j=1}^{N_d} \|z_{i,j}\|_0 = N_c$, is a distinct characteristic in our biometric problem. In order to achieve a regular data structure for phase templates in the gallery, we force the code to have the same size N_c for every template. Since this kind of L^0 -norm constraint is difficult to solve, most of the researches on CSC relax this constraint to L^1 -norm constraint [95, 96, 97, 98, 99]. However, we found that, for a small N_c as is required here, the ideal L^0 -norm constraint can be efficiently addressed through the use of a specially designed CSC optimization algorithm based on Matching Pursuit (MP) [100] technique, which is described next.

4.4.2 Basic strategy for optimization

If we fix the codes in Eq. (4.17), the filter optimization becomes a convex problem. Conversely, if we fix the filters, the code optimization is a combinatorial NP-hard problem due to the L^0 -norm. This code optimization is usually relaxed into a convex problem by changing the L^0 -norm constraint to the L^1 -norm constraint such as $\sum_{j=1}^{N_d} \|z_{i,j}\|_1 \leq N_c$. The relaxed code optimization is known in the field of statistics as Least Absolute Shrinkage and Selection Operator (LASSO) regression and in the field of signal processing as basis pursuit. Taking advantage of biconvex nature of the relaxed problem, a common practice in CSC optimization is to alternate between code optimization for fixed filters and filter optimization for fixed codes. Following this strategy, Bristow et al. 2013 [95] advocate for applying Alternating Direction Method of Multipliers (ADMMs) [101] to both code optimization and filter optimization. Following, Bristow et al. [98] reviewed the development of the CSC algorithms, and presented a framework for ADMM and proximal gradient methods such as Fast Iterative Shinkage-Thresholding (FISTA) [102]. Related works using ADMM follow this lead, [96, 97], and more recently Heide et al. [99] presented a framework with proximal operators.

In these algorithms, code optimization starts from a *dense solution* and iteratively prune coefficients from the codes until a certain degree of sparsity is obtained. They do not control the exact number of coefficients. These algorithms can attain good solutions for

low sparsity but not for high sparsity. Therefore, they can not be applied with success to our problem defined by Eq. (4.17), where strict L^0 -norm constraint with small N_c , i.e., highly sparse condition, must be addressed. Clearly, it is quite difficult to address the general class of L^0 -norm problems directly. But, for problems with small N_c , we found that MP technique can be applied efficiently to code optimization. For smaller N_c , MP ensures better reconstruction accuracy close to the global optimum, as is theoretically analyzed in [103].

From the above mentioned point of views, we propose a new optimization method that alternately performs MP for code optimization and Ordinary Least Squares (OLS) for filter optimization. The proposed approach can be regarded, in a sense, as an extension of the Method of Optimal Directions (MOD) [104] to convolutional version of dictionary learning.

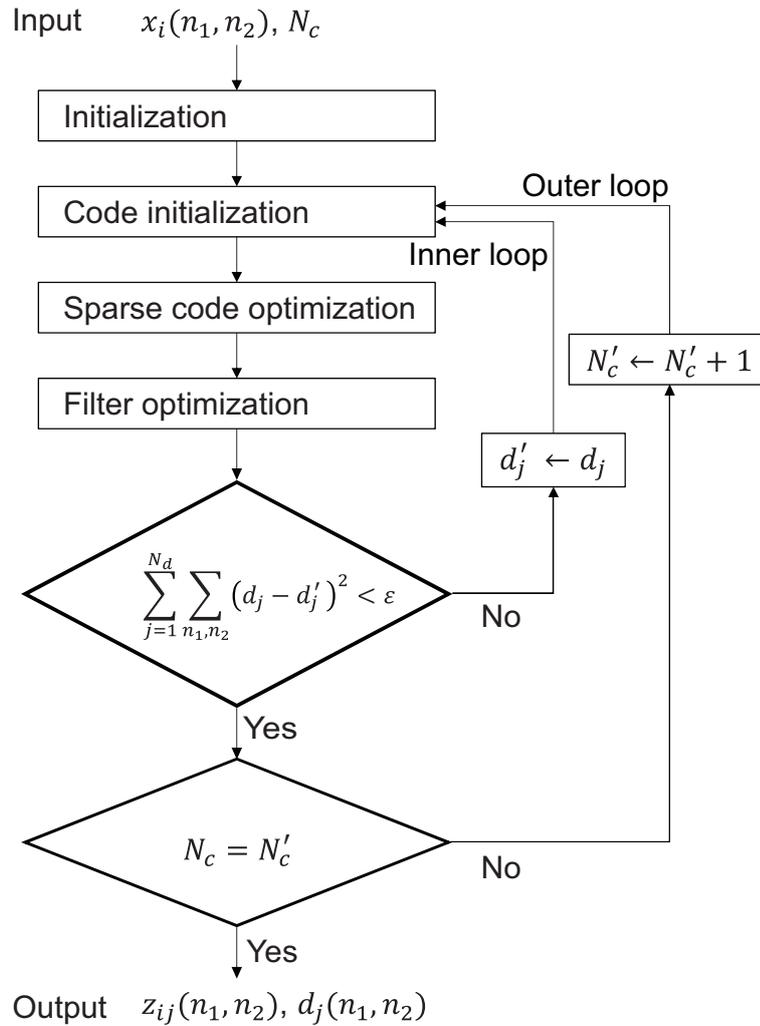


Figure 4.2: Flow diagram of the CSC optimization algorithm.

Algorithm 3 Initialization

Set N'_c to 2

for $j \in \{1, \dots, N_d\}$ **do**

Initialize $d_j(n_1, n_2)$ by random values generated from the normal distribution with $\mu = 0$ and $\sigma^2 = 1$

Apply a Hanning window to $d_j(n_1, n_2)$

Compute the filter energy e_j by

$$e_j = \sum'_{n_1, n_2} |d_j(n_1, n_2)|^2$$

Normalize $d_j(n_1, n_2)$ by $d_j(n_1, n_2)/\sqrt{e_j}$
end for

4.4.3 Incremental CSC optimization algorithm

The overall structure of our proposed algorithm is depicted in Fig. 4.2. The basic procedures for the inner loop are code optimization based on MP and filter optimization using OLS as described above. Although relatively good performance is expected by alternating between MP and OLS, the quality of the solution significantly depends on the initial setting of filters $d_j(n_1, n_2)$. We address this drawback by adding an outer loop to start the optimization with the simplified sparsity constraint and to repeat the optimization by gradually increasing the number of non-zero elements until it reaches N_c . For this purpose, we introduce a new counter N'_c for controlling the number of non-zero elements for the inner loop. We found that the result of optimization with the L^0 -norm sparsity N'_c can be used as a good initial condition for the case of $N'_c + 1$. This incremental approach drastically improves the quality of the final solution. The initialization of N'_c and filters $d_j(n_1, n_2)$ is shown in Algorithm 3, where $d_j(n_1, n_2)$ are initialized only once when the algorithm starts.

At the beginning of every outer iteration, the codes $z_{i,j}(n_1, n_2)$ are to be initialized to 0 in ordinary MP algorithm. However, we observe through a set of preliminary experiments that reconstruction accuracy can be improved by skipping the code initialization and refining the code values after the counter N'_c reaches a certain value. We determine such value of N'_c by empirical observations. The code initialization is shown in Algorithm 4.

4.4.4 Sparse code optimization

Let us define the residual atomic correlation $c_{i,j}^{\text{res}}(n_1, n_2)$ as follows:

$$c_{i,j}^{\text{res}}(n_1, n_2) = d_j(n_1, n_2) \otimes \{x_i(n_1, n_2) - \hat{x}_i(n_1, n_2)\}, \quad (4.18)$$

Algorithm 4 Code initialization

Require: N'_c and $z_{i,j}(n_1, n_2)$

Ensure: $z_{i,j}(n_1, n_2)$

if $N'_c \leq 10$ and # of inner-loop iterations ≤ 8 **then**

for $i \in \{1, \dots, N_u\}$ and $j \in \{1, \dots, N_d\}$ **do**

$z_{i,j}(n_1, n_2) \leftarrow 0$

end for

end if

where $\hat{x}_i(n_1, n_2)$ is computed by (4.16). Then, Sparse code optimization step in Fig. 4.2 is shown in Algorithm 5.

4.4.5 Filter optimization

The filter optimization problem in Fig. 4.2 (derived from (4.17)) is solved easily in the frequency domain, similar to the recent CSC studies [95, 96, 97, 99]:

$$\arg \min_{D_j(k_1, k_2)} \sum_{i=1}^{N_u} \left| X_i(k_1, k_2) - \sum_{j=1}^{N_d} D_j(k_1, k_2) Z_{i,j}(k_1, k_2) \right|^2. \quad (4.19)$$

Notice that Eq. (4.19) can be solved independently for each frequency component with OLS. For simplicity, we use a vector form for a given frequency defined as follows:

$$\mathbf{X}_{k_1, k_2} = \begin{bmatrix} X_1(k_1, k_2) \\ X_2(k_1, k_2) \\ \vdots \\ X_{N_u}(k_1, k_2) \end{bmatrix}, \quad (4.20)$$

$$\mathbf{D}_{k_1, k_2} = \begin{bmatrix} D_1(k_1, k_2) \\ D_2(k_1, k_2) \\ \vdots \\ D_{N_d}(k_1, k_2) \end{bmatrix}, \quad (4.21)$$

$$\mathbf{Z}_{k_1, k_2} = \begin{bmatrix} Z_{1,1}(k_1, k_2) & \cdots & Z_{1,N_d}(k_1, k_2) \\ Z_{2,1}(k_1, k_2) & \cdots & Z_{2,N_d}(k_1, k_2) \\ \vdots & \ddots & \vdots \\ Z_{N_u,1}(k_1, k_2) & \cdots & Z_{N_u,N_d}(k_1, k_2) \end{bmatrix}. \quad (4.22)$$

In this way, the problem becomes $B_1 \times B_2$ OLS sub-problems of size $N_d \times N_d$ as follows:

$$\arg \min_{\mathbf{D}_{k_1, k_2}} \|\mathbf{X}_{k_1, k_2} - \mathbf{Z}_{k_1, k_2} \mathbf{D}_{k_1, k_2}\|_2^2. \quad (4.23)$$

Algorithm 5 Sparse code optimization

Require: N'_c , $x_i(n_1, n_2)$, $z_{i,j}(n_1, n_2)$ and $d_j(n_1, n_2)$ **Ensure:** $z_{i,j}(n_1, n_2)$ **for** $i \in \{1, \dots, N_u\}$ **do**Count the number of non-zero elements N''_c by $\sum_{j=1}^{N_d} \|z_{i,j}\|_0$ **if** $N''_c < N'_c$ **then****while** $N''_c < N'_c$ **do**Compute $c_{i,j}^{\text{res}}(n_1, n_2)$ by Eq. (4.18)Find $\{j', n'_1, n'_2\}$ maximizing the absolute value of the residual atomic correlation functions by

$$\{j', n'_1, n'_2\} \leftarrow \arg \max_{j, n_1, n_2} |c_{i,j}^{\text{res}}(n_1, n_2)|$$

Update $z_{i,j'}(n'_1, n'_2)$ by

$$z_{i,j'}(n'_1, n'_2) \leftarrow c_{i,j'}^{\text{res}}(n'_1, n'_2)$$

$$N''_c \leftarrow N''_c + 1$$

end while**else**Compute $c_{i,j}^{\text{res}}(n_1, n_2)$ by Eq. (4.18)Update all the non-zero elements in $z_{i,j}(n_1, n_2)$ by

$$z_{i,j}(n_1, n_2) \leftarrow z_{i,j}(n_1, n_2) + \eta c_{i,j}^{\text{res}}(n_1, n_2) \quad (0 < \eta < 1)$$

end if**end for**

The OLS solution is as follows:

$$\mathbf{D}_{k_1, k_2} = \left(\mathbf{Z}_{k_1, k_2}^\dagger \mathbf{Z}_{k_1, k_2} \right)^{-1} \mathbf{Z}_{k_1, k_2}^\dagger \mathbf{X}_{k_1, k_2}, \quad (4.24)$$

where \dagger is the Hermitian transpose.

This OLS solution differs from [95, 96, 97, 99] in that no spatial constraint was imposed and thus, the terms related to spatial constraint are avoided. However, as the filters resulting from OLS are not normalized or bounded, their values might diverge after consecutive optimization steps; nonetheless, this is addressed by re-scaling the sparse codes accordingly. Thus, filter optimization, shown in Algorithm 6, produces the optimized filters $d_j(n_1, n_2)$ and scaled codes $z_{i,j}$. Note that the outputs of the CSC optimization algorithm are available both in the spatial and frequency domain, i.e., $z_{i,j}(n_1, n_2)$ and $D_j(k_1, k_2)$.

Algorithm 6 Filter optimization

Require: $x_i(n_1, n_2)$ and $z_{i,j}(n_1, n_2)$

Ensure: $d_j(n_1, n_2)$ and $z_{i,j}(n_1, n_2)$

 Compute 2D DFT of $x_i(n_1, n_2)$ and $z_{i,j}(n_1, n_2)$

 Obtain $D_j(k_1, k_2)$ by Eq. (4.24)

for $j \in \{1, \dots, N_d\}$ **do**

 Compute the filter energy e_j by

$$e_j = \sum'_{k_1, k_2} |D_j(k_1, k_2)|^2 / (B_1 B_2)$$

 Normalize $D_j(k_1, k_2)$ by $D_j(k_1, k_2) / \sqrt{e_j}$

 Scale $z_{i,j}(n_1, n_2)$ by $z_{i,j}(n_1, n_2) \sqrt{e_j}$

end for

 Compute 2D DFT of $D_j(k_1, k_2)$

4.5 Palmprint identification with Phase-Based Hierarchical Block Matching

Now we are ready to design a complete palmprint identification algorithm on the basis of fundamental discussions of the previous sections. In the previous section, we discussed a computationally efficient way of calculating phase-based similarities between a probe block extracted from the probe image J and a set of N_u reference blocks extracted from the reference images $\{I_1, \dots, I_{N_u}\}$ in the gallery. This one-to-many block-wise image

comparison makes an implicit assumption that all palmprint images are aligned in position with perfectly normalized hand pose. However, in real situation, palmprint images contain global translations, rotations and nonlinear deformations.

Hence, for given a small local block on a full-size palmprint image, the problem is to find the corresponding block location on another palmprint image. In real-world palmprint identification, this problem of block-wise correspondence search must be addressed before block-wise similarity evaluation.

We described in Sec. 3.8 a Phase-Based Correspondence Matching algorithm (PB-CM) [17, 28] that involve a coarse-to-fine search strategy. PB-CM has achieve very high recognition accuracy in palmprint and finger-knuckle biometrics. However, the correspondence matching algorithm is suitable for one-to-one comparison required in verification mode and not for one-to-many comparison required in identification mode. The goal of the next section is to modify the correspondence matching algorithm so as to perform one-to-many comparison with reduced computational complexity using the proposed CSC-based correlation technique proposed in Sec. 4.3.

4.5.1 Phase-Based Correspondence Matching

Let us revisit the correspondence matching algorithm described in Sec. 3.8. Given a reference image I (registered in the gallery) and a probe image J , the algorithm comprises a correspondence search step (Sec. 3.8.1) and a similarity evaluation step (Sec. 3.8.2). The execution of this algorithm involves two types of phase feature computation, (i) one for a set of image blocks extracted from I at predetermined locations and (ii) one for a set of image blocks extracted from J at corresponding block locations. The locations of the reference blocks on the reference images I_1, \dots, I_{N_u} are predetermined when we enroll the reference blocks (or, more precisely, register the phase templates (Eq. (4.1)) of the blocks) in the gallery. Hence, the phase templates of every reference block are computed only once during enrolment, i.e., they are computed off-line. A major problem regarding the computational efficiency arises when we apply this correspondence matching algorithm to the one-to-many comparisons required for identification. The problem stems from the fact that the corresponding locations of the probe blocks on J vary depending on the reference images I_1, \dots, I_{N_u} . Thus, in order to compare a single probe image J with a set N_u reference images, we must repeat the feature computation from J for the corresponding locations of each reference image. This means that the total number of different probe blocks is very large, as much as the total number of reference blocks enrolled in the gallery. As a consequence, the number of 2D DFT computations to be performed on-line for generating the phase features of probe blocks (Eq. (4.2)) is significantly large in identification mode.

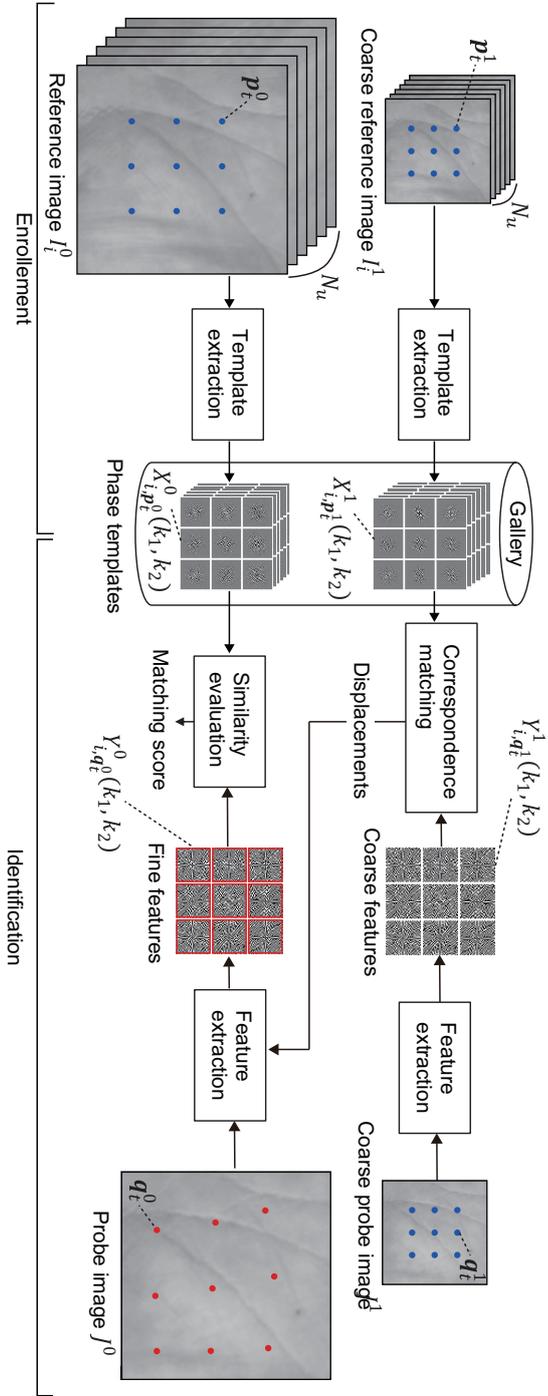


Figure 4.3: Overall of palmprint identification using Phase-Based Correspondence Matching (PB-CM) with two-layer resolution pyramid. Left: enrollment process. Right: identification process.

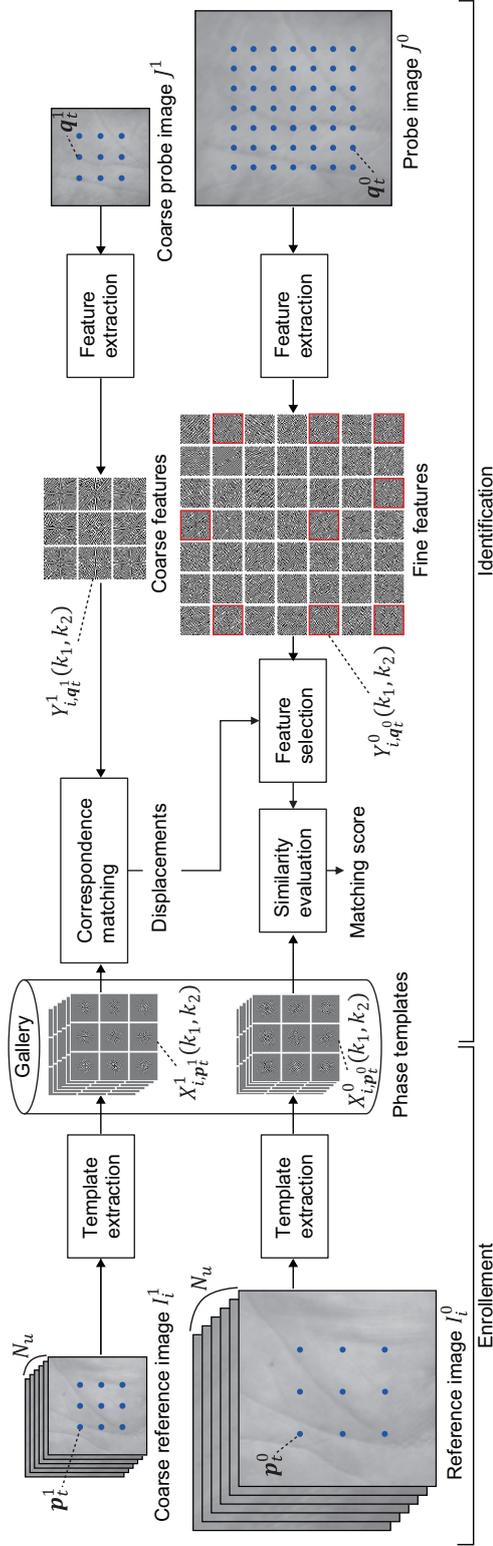


Figure 4.4: Overall of palmprint identification using Phase-Based Hierarchical Block Matching (PB-HBM) algorithm with two-layer resolution pyramid. Red boxes on fine features indicate the selected features for similarity evaluation. Left: enrollment process. Right: identification process.

4.5.2 Phase-Based Hierarchical Block Matching (PB-HBM)

We address the aforementioned problem by modifying the original correspondence matching algorithm to have fixed locations also for the probe blocks. The modified algorithm, which we call Phase-Based Hierarchical Block Matching (PB-HBM) algorithm, has reduced computational complexity compared with the correspondence matching algorithm [17, 28]. In addition, since the PB-HBM algorithm is based on one-to-many block comparisons, we can apply CSC-BLPOC technique proposed in Sec. 4.3 for further reducing computational cost and gallery data size. For simplicity, we first explain how to evaluate the similarity between a single image pair, a probe image J and a reference image I . PB-HBM starts with defining N_b reference blocks on I , finds their best match (corresponding) blocks on J , and evaluates their block-wise similarity. The corresponding blocks are selected from predetermined candidate blocks on J .

Let $P = \{\mathbf{p}_1, \dots, \mathbf{p}_{N_b}\} \subseteq \mathbb{Z}^2$ denote the set of locations (i.e., coordinates) for the N_b reference blocks on I , and let $Q_{cand} \subseteq \mathbb{Z}^2$ denote the set of locations for candidate blocks on J , where we assume the number of candidate locations in Q_{cand} is sufficiently large, i.e., $|Q_{cand}| \geq N_b$. Given the set P on I , the problem is to find a set of corresponding block locations $Q (\subseteq Q_{cand})$ on J .

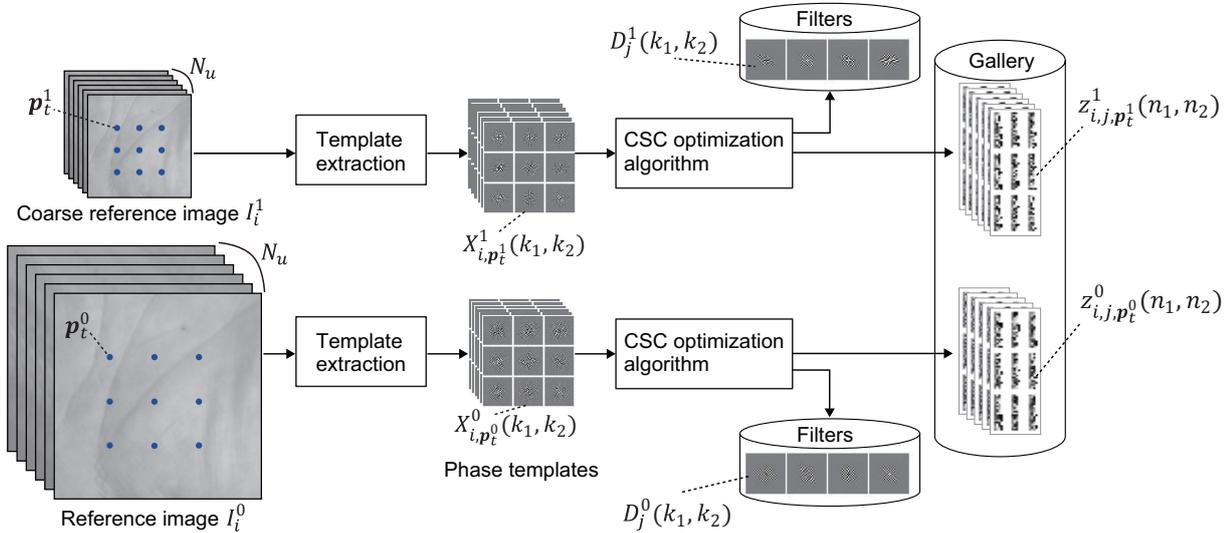


Figure 4.5: Enrolment for palmprint identification using two-layer Phase-Based Hierarchical Block Matching with Convolutional Sparse Coding (PB-HBM-CSC)

To solve this problem, we shall adopt a coarse-to-fine strategy similar to that of the steps (i) and (ii) of the correspondence search of PB-CM in Sec.3.8.1. This is two set of images I^l and J^l for layers $l = 0, \dots, l_{max}$ where $I^0 = I$ and $J^0 = J$. I^l and J^l for $l > 0$ is given by Eq. (3.57) and Eq. (3.58) from Sec. 3.8.1. Likewise, the reference block locations

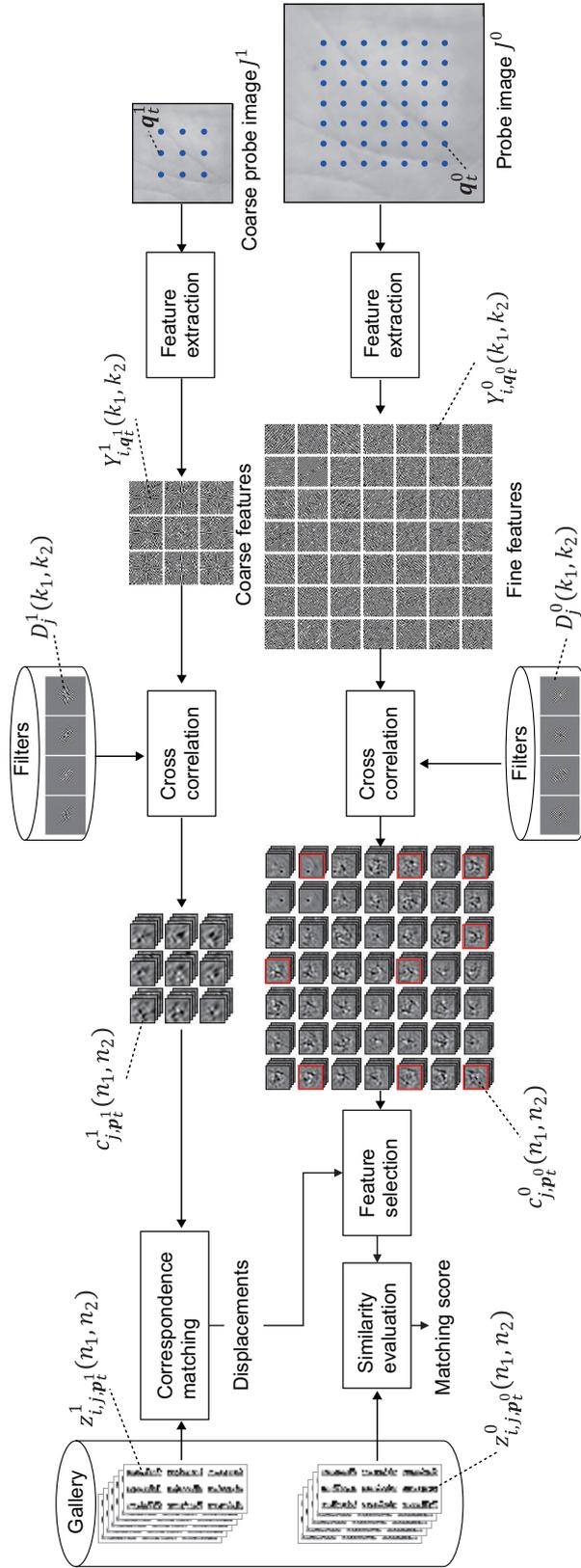


Figure 4.6: Overall palmprint identification using two-layer Phase-Based Hierarchical Block Matching with Convolutional Sparse Coding (PB-HBM-CSC)

$P^l = \{\mathbf{p}_1^l, \dots, \mathbf{p}_{N_b}^l\}$ on the l -th layer reference image I^l are given in advance by Eq. (3.59). Remember that for the original image layer $l = 0$, we have $\mathbf{p}_t^0 = \mathbf{p}_t$.

Given the reference block \mathbf{p}_t^l on I^l , the problem considered here is to find the corresponding block \mathbf{q}_t^l on J^l , for all $t = 1, \dots, N_b$, with a coarse-to-fine search starting from $l = l_{max}$ to $l = 0$. We reduce the computational complexity of the coarse-to-fine search by restricting the corresponding blocks to a set of predetermined locations Q_{cand}^l on J^l .

The procedure to recursively find the corresponding block pairs $(\mathbf{p}_t^l, \mathbf{q}_t^l)$ from the upper layer pairs $(\mathbf{p}_t^{l+1}, \mathbf{q}_t^{l+1})$ consists of the following steps: (i) compute BLPOC function between the blocks at \mathbf{p}_t^{l+1} and \mathbf{q}_t^{l+1} , (ii) derive their displacement δ_t^{l+1} , and (iii) update their positions to have $(\mathbf{p}_t^l, \mathbf{q}_t^l)$ with higher resolution, where \mathbf{p}_t^l is given by Eq. (3.59) and \mathbf{q}_t^l is derived as

$$\mathbf{q}_t^l = \arg \min_{\mathbf{q} \in Q_{\text{cand}}^l} \|\mathbf{q} - 2(\mathbf{q}_t^{l+1} + \delta_t^{l+1})\|_2. \quad (4.25)$$

For simplicity, we start the above recursion with the initial setting:

$$\mathbf{q}_t^{l_{max}} = \mathbf{p}_t^{l_{max}}. \quad (4.26)$$

As a result of this procedure, we can obtain the set of N_b pairs $(\mathbf{p}_t^0, \mathbf{q}_t^0)$ of corresponding blocks at the original image layer. For all the block pairs $(\mathbf{p}_t^0, \mathbf{q}_t^0)$, we compute BLPOC functions and derive their block-wise similarities (i.e., BLPOC peak values α_t^0). By taking an average of these block-wise similarities, we have an overall matching score between the image pair I and J .

Algorithm 7 Enrolment procedure for PB-HBM

Require: N_u reference images

Ensure: $2N_u N_b$ phase templates $X_{i, \mathbf{p}_t^l}^l(k_1, k_2)$

for $i \in \{1, \dots, N_u\}$ **do**

 Generate an image pyramid, I_i^1 and $I_i^0 (= I_i)$, by Eq. (3.57)

for $l \in \{0, 1\}$ **do**

for $t \in \{1, \dots, N_b\}$ **do**

 Obtain a reference block location \mathbf{p}_t^l on I_i^l by Eq. (3.59)

 Compute the local phase template $X_{i, \mathbf{p}_t^l}^l(k_1, k_2)$ by Eq. (4.1)

 Store the phase template into the gallery

end for

end for

end for

We must address the trade-off between recognition accuracy and computational cost, since a higher number of reference blocks N_b implies not only higher recognition accuracy

but also higher computational cost in PB-HBM. Another key parameter to be considered in PB-HBM is l_{max} , where higher l_{max} implies higher robustness in recognition, but at an increased computational cost. In the experiments presented in this section, we found that two resolution layers (coarse $l = 1$ and original $l = 0$) are sufficient, and hence $l_{max} = 1$.

We describe how to apply PB-HBM to the identification mode where a probe image J is compared with N_u reference images $\{I_1, I_2, \dots, I_{N_u}\}$ enrolled in the gallery. In identification, the block-wise comparisons within PB-HBM are performed in the Fourier domain for computational efficiency. As discussed in Sec. 5.3, this involves the comparisons of phase templates with phase features computed from the reference blocks and probe blocks, respectively. In this way, phase templates can be prepared in advance and phase features are computed once before the comparisons. Fig. 4.4 illustrates the overall flow of palmprint identification using two-layer PB-HBM and for contrast two-layer PB-CM is illustrated as well in Fig. 4.3. The enrolment process, which is common for PB-CM and PB-HBM, is depicted at the left side of the gallery database while the identification process depicted at the right side. In the enrolment process, we construct an image pyramid for each reference image I_i , extract the phase templates $X_{i,p_t}^l(k_1, k_2)$ from I_i^l at fixed locations p_t^l and store them into the gallery, where $i = 1, \dots, N_u$, $l = 0, 1$ and $t = 1, \dots, N_b$. A detailed description of the enrolment procedure is shown in Algorithm 7.

In the identification process, we construct an image pyramid of J , extract the phase features $Y_q^l(k_1, k_2)$ from J^l at fixed locations q in Q_{cand}^l and compare them with the phase templates stored into the gallery. These are a total of $2N_u N_b$ comparisons which are carried out by computing BLPOC functions between phase features and phase templates. A detailed description of the identification procedure is shown in Algorithm 8.

PB-HBM computes only a phase feature for each candidate location which is a vast reduction compared with correspondence matching [17, 28], where a phase feature is computed for each phase template. Finally, we combine PB-HBM and CSC-BLPOC into our final proposed algorithm, namely, PB-HBM-CSC.

4.5.3 Phase-Based Hierarchical Block Matching with Convolutional Sparse Coding (PB-HBM-CSC)

Fig. 4.5 and Fig. 4.6 illustrate the overall process of PB-HBM-CSC. Fig. 4.5 illustrates the enrolment process where we store the sparse representation of phase templates into the gallery. During enrolment, first, we extract all the phase templates $X_{i,p_t}^l(k_1, k_2)$. Next, for each layer, we apply the CSC optimization algorithm described in Sec. 4.4 to all the phase templates of that layer. Then, we store the obtained convolution filters $D_j^l(k_1, k_2)$ and sparse codes $z_{i,j,p_t}^l(n_1, n_2)$ into the gallery. A detailed description of the enrolment

Algorithm 8 Identification procedure for PB-HBM

Require: A probe image J

Ensure: Matching scores for reference images I_i

Generate an image pyramid, J^1 and $J^0(= J)$, by Eq. (3.58)

for $l \in \{0, 1\}$ **do**

for each candidate block location $\mathbf{q}(\in Q_{\text{cand}}^l)$ on J^l **do**

 Compute the phase feature $Y_{\mathbf{q}}^l(k_1, k_2)$ by Eq. (4.2)

end for

end for

for $i \in \{1, \dots, N_u\}$ **do**

for $t \in \{1, \dots, N_b\}$ **do**

 Obtain \mathbf{p}_t^0 and \mathbf{p}_t^1 by Eq. (3.59)

 Find the corresponding block location \mathbf{q}_t^1 on J^1 be equal to \mathbf{p}_t^1 on I_i^1 by Eq. (4.26)

 Compute the BLPOC function for the block pair $(\mathbf{p}_t^1, \mathbf{q}_t^1)$ to derive δ_t^1 , where we use Eq. (4.3) and (4.4) with $X_{i, \mathbf{p}_t^1}^l(k_1, k_2)$ stored in the gallery and the $Y_{\mathbf{q}_t^1}^l(k_1, k_2)$ prepared above

 Determine \mathbf{q}_t^0 from \mathbf{q}_t^1 and δ_t^1 using Eq. (4.25)

 Compute the BLPOC function for the block pair $(\mathbf{p}_t^0, \mathbf{q}_t^0)$ to derive the peak value α_t^0 , where we use Eq. (4.3) and (4.4) with $X_{i, \mathbf{p}_t^0}^0(k_1, k_2)$ and $Y_{\mathbf{q}_t^0}^0(k_1, k_2)$

end for

 Compute the matching score between I_i and J as an average of α_t^0 for $t = 1, \dots, N_b$

end for

process is shown in Algorithm 9. Fig. 4.6 illustrates the identification process. At the beginning of this process, we extract the phase features $Y_{\mathbf{q}}^l(k_1, k_2)$ and compute their atomic correlation functions $c_{j, \mathbf{q}}^l(k_1, k_2)$ for all \mathbf{q} in Q_{cand}^l . Then, we compute the CSC-BLPOC functions between atomic correlation functions and sparse codes. A detailed description of the identification process is shown in Algorithm 10.

By applying CSC-BLPOC to PB-HBM, we reduce the number of 2D-IDFTs required for identification. While PB-HBM requires $2N_u N_b$ 2D-IDFTs for the computation of BLPOC functions, PB-HBM-CSC only requires $N_d(N_b + |Q_{\text{cand}}|)$ (where $N_d \ll N_u$) for the computation of atomic correlation functions.

As a final remark, the coarse-to-fine strategy allows us to limit the search range of CSC-BLPOC function (defined by M_1 and M_2 in Sec. 4.3.3) which is important for the total computational saving of CSC-BLPOC.

Algorithm 9 Enrolment procedure for PB-HBM-CSC

Require: N_u reference images I_i **Ensure:** $D_j^l(k_1, k_2)$ and $z_{i,j,p_t^l}^l(n_1, n_2)$ Compute the local phase templates $X_{i,p_t^l}^l(k_1, k_2)$ following the enrolment procedure for PB-HBM**for** $l \in \{0, 1\}$ **do** Compute sparse codes $z_{i,j,p_t^l}^l(n_1, n_2)$ and convolution filters $D_j^l(k_1, k_2)$ by applying the CSC optimization algorithm in Sec. 4.4 to the phase templates $X_{i,p_t^l}^l(k_1, k_2)$

Store convolution filters and sparse codes into the gallery

end for

Algorithm 10 Identification procedure for PB-HBM-CSC

Require: A probe image J **Ensure:** Matching scores for reference images I_i Generate an image pyramid, J^1 and $J^0(= J)$, by Eq. (3.58)**for** $l \in \{0, 1\}$ **do** **for** each candidate block location $\mathbf{q}(\in Q_{cand}^l)$ on J^l **do** Compute the phase feature $Y_{\mathbf{q}}(k_1, k_2)$ by Eq. (4.2) **for** each convolution filter D_j^l **do** Compute the atomic correlation functions $c_{j,\mathbf{q}}(k_1, k_2)$ using $Y_{\mathbf{q}}(k_1, k_2)$ by Eq. (4.14) **end for** **end for****end for****for** $i \in \{1, \dots, N_u\}$ **do** **for** $t \in \{1, \dots, N_b\}$ **do** Obtain \mathbf{p}_t^0 and \mathbf{p}_t^1 by Eq. (3.59) Find the corresponding block location \mathbf{q}_t^1 on J^1 be equal to \mathbf{p}_t^1 on I_i^1 by Eq. (4.26) Compute the CSC-BLPOC function for the block pair $(\mathbf{p}_t^1, \mathbf{q}_t^1)$ to derive δ_t^1 , where we use Eq. (4.15) with $z_{i,j,p_t^1}^1(n_1, n_2)$ stored in the gallery and the $c_{j,\mathbf{q}_t^1}^1(k_1, k_2)$ prepared above Determine \mathbf{q}_t^0 from \mathbf{q}_t^1 and δ_t^1 by Eq. (4.25) Compute the CSC-BLPOC function for the block pair $(\mathbf{p}_t^0, \mathbf{q}_t^0)$ to derive the peak value $\hat{\alpha}_t^0$, where we use Eq. (4.15) with $z_{i,j,p_t^0}^0(n_1, n_2)$ and $c_{j,\mathbf{q}_t^0}^0(k_1, k_2)$ **end for** Compute the matching score between I_i and J as an average of $\hat{\alpha}_t^0$ for $t = 1, \dots, N_b$ **end for**

4.6 Experiments and discussion

This section presents experimental evaluation of the proposed and conventional methods for palmprint identification and clarifies the major advantages of the proposed method.

4.6.1 Methods

The following is a list of palmprint identification methods to be evaluated in the experiments:

- (i) Competitive Code (*CompCode*) [8],
- (ii) Ordinal Code (*OrdiCode*) [9],
- (iii) Double Orientation Code (*DOC*) [72].
- (iv) Phase-Based Correspondence Matching (*PB-CM*) [28],
- (v) Phase-Based Hierarchical Block Matching (*PB-HBM*) described in Sec. 4.5.2,
- (vi) Phase-Based Hierarchical Block Matching with Convolutional Sparse Coding (*PB-HBM-CSC*) described in Sec. 4.5.3,

The methods (i)–(iii) are based on features extracted using Gabor and ordinal filters, while the methods (iv)–(vi) are based on local phase features. As for the conventional methods to be compared, we selected (i)–(iii) (Sec. 2.4), since they are very fast (which is essential for identification mode) and have favorable recognition accuracy. The method (iv) can be regarded as a baseline for the methods (v) and (vi). The method (v) is a modified version of method (iv), where correspondence matching is carried out on a predetermined set of candidate blocks with fixed locations. The method (vi) reduces the computational complexity of method (v) by introducing CSC for phase template representation.

4.6.2 Palmprint databases

We employ two public databases for our experiments: the PolyU palmprint database [90] and the CASIA palmprint database [91]. The images in the PolyU database show only minute displacements and deformations, since the subjects' hands are being fixed during the image acquisition. Conversely, the images in the CASIA database show various kind of transformations due to contactless image acquisition. Although these databases have a variable number of images per palm, we use a fixed number of images per palm in our experiments. For the PolyU database, we use 3,740 images from 374 palms (hence 10 images per palm) while for the CASIA database, we use 4,800 images from 600 palms (hence 8

Table 4.2: Parameters for Gabor-coding based methods.

(i) CompCode	
Parameter	Value
Number of orientations	6
Radial frequency μ	0.1833 radians per pixel
Elliptical Gaussian envelope with standard deviations (σ, β)	(2.8090, 1.4045)
(ii) OrdiCode	
Parameter	Vale
Number of orientations	3
Filter's horizontal scale δ_x	0.9363
Filter's vertical scale δ_y	2.8090
(iii) DOC	
Parameter	Value
Number of orientations	6
Radial frequency μ	0.0916 radians per pixel
Gaussian envelope with standard deviation σ	5.6179

images per palm). Within both databases, one image per palm is used for enrolment and the rest for testing. We utilize the preprocessing method presented in [61] to extract and normalize the region of interest from these palmprint images as mentioned in Sec. 2.3.2, where the image size after preprocessing is 160×160 pixels.

4.6.3 Parameters

Table 4.2 summarizes the parameters for the conventional algorithms (i)–(iii) according to [72, 105]. Table 4.3, on the other hand, summarizes the parameters for the phase-based methods (iv)–(vi). In addition, we discard texture-less reference/probe blocks in matching operation to improve the recognition accuracy of methods (iv)–(vi). The AC energy¹ of a image block $f(n_1, n_2)$ is calculated by

$$e^{AC} = \sum_{n_1, n_2} (f(n_1, n_2) - f^{DC})^2, \quad (4.27)$$

¹equivalent to pixel variance times block size

Table 4.3: Parameters for phase-based methods.

(iv) PB-CM, (v) PB-HBM and (vi) PB-HBM-CSC	
Parameter	Value
Number of reference blocks N_b	49 blocks
Spacing between adjacent reference blocks	11 pixels
Image block size $L_1 \times L_2$	64×64 pixels
Phase template/feature size $B_1 \times B_2$	32×32 coefficients
Spectral weighting function $H(k_1, k_2)$	$\sigma^2 = 0.0796$ for the coarse resolution
is a Gauss function with variance σ^2	$\sigma^2 = 0.1137$ for the original resolution
Search-range for the BLPOC function (M_1, M_2)	(8, 8) for the coarse resolution (5, 5) for the original resolution
(v) PB-HBM and (vi) PB-HBM-CSC	
Parameter	Value
Number of candidate blocks $ Q_{\text{cand}} $	169 blocks
Spacing between adjacent candidate blocks	8 pixels
(vi) PB-HBM-CSC	
Parameter	Value
Number of filters N_d for CSC	16 filters
Number of coefficients N_c for CSC	32 coefficients
ϵ	$0.67/N'_c$

where f^{DC} indicates the DC component of $f(n_1, n_2)$ as well as Eq. (3.9). If an AC energy is below a quality threshold $thr_{AC} = 1.2$, i.e., textureless, we do not use this image block in correspondence matching.

4.6.4 Evaluation of identification performance

The performance of the above methods is evaluated by the rank-1 identification error and by the average computation time (excluding preprocessing time). All the methods are implemented by using MATLAB 7.14 and executed on a system with Intel Xeon E5-2680 2.70 GHz, 128 GB and CentOS 6.8.

Table 4.4 summarizes the results of our experiments together with the results reported in [72] for conventional methods. The difference between our results for (i)–(iii) and the reported values in [72] can be due to the difference in the preprocessing step. For

Table 4.4: Experimental results of identification errors and computation times.

Method	Identification error rate [%]		Computation time [sec.]	
	PolyU	CASIA	PolyU	CASIA
(i) CompCode	2.16 (3.57 in [72])	9.86	0.21	0.31
(ii) OrdiCode	0.74 (4.81 in [72])	8.07	0.21	0.32
(iii) DOC	0.71 (2.55 in [72])	11.71	0.79	1.50
(iv) PB-CM	0	0.0238	9.21	14.9
(v) PB-HBM	0	0.0476	1.37	2.20
(vi) PB-HBM-CSC	0	0.0476	0.75	0.95

both databases, the phase-based methods exhibit significantly lower identification errors compared with the conventional methods. PB-HBM and PB-HBM-CSC have the same identification errors, which are slightly higher than that of PB-CM. This difference in error rate is attributed to the use of predetermined block locations in PB-HBM and PB-HBM-CSC. The computation time of PB-HBM is more than six times faster than that of PB-CM, since the number of phase feature extractions is vastly reduced. The introduction of CSC further reduces the computation time by half with respect to the computation time of PB-HBM. This demonstrates that PB-HBM-CSC is effectively faster than PB-CM.

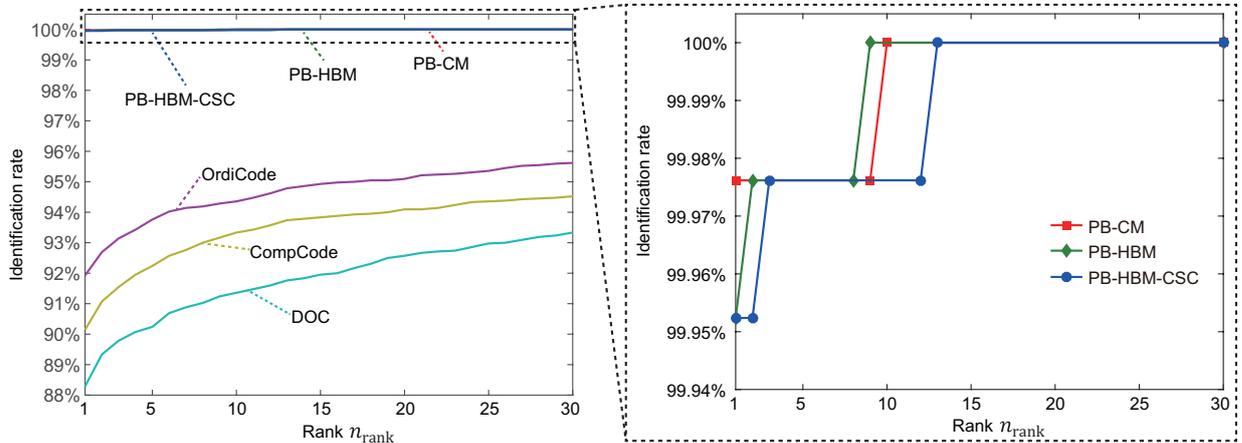
**Figure 4.7:** CMC curve evaluated using the CASIA database for all methods and magnified view of CMC curves for identification rates higher than 99.94%.

Fig. 4.7 shows the Cumulative Match Characteristic (CMC) curves for our experiments with CASIA database, where the CMC curve presents the probability of observing the correct identity within the top n_{rank} ranks (Sec. 2.5). As for the conventional methods, OrdiCode outperforms CompCode and DOC as is observed in the left plot of Fig. 4.7.

Table 4.5: Registered data per image for each method.

	Method	Bit length of element	# of elements	Total size [bytes]
(i)	CompCode	3	32×32	384
(ii)	OrdiCode	3	32×32	384
(iii)	DOC	3	$32 \times 32 \times 2$	768
(iv)	PB-CM	4	$32 \times 16 \times 49 \times 2$	25,088
(v)	PB-HBM	4	$32 \times 16 \times 49 \times 2$	25,088
(vi)	PB-HBM-CSC	32+16	$32 \times 49 \times 2$	18,816

However, none of them reach 100% identification rate within the top 30 ranks. The right plot of Fig. 4.7 shows a magnified view of the CMC curves for identification rates higher than 99.94%. We observe that the phase-based methods have similar accuracy and reach 100% identification rate within the top 14 ranks.

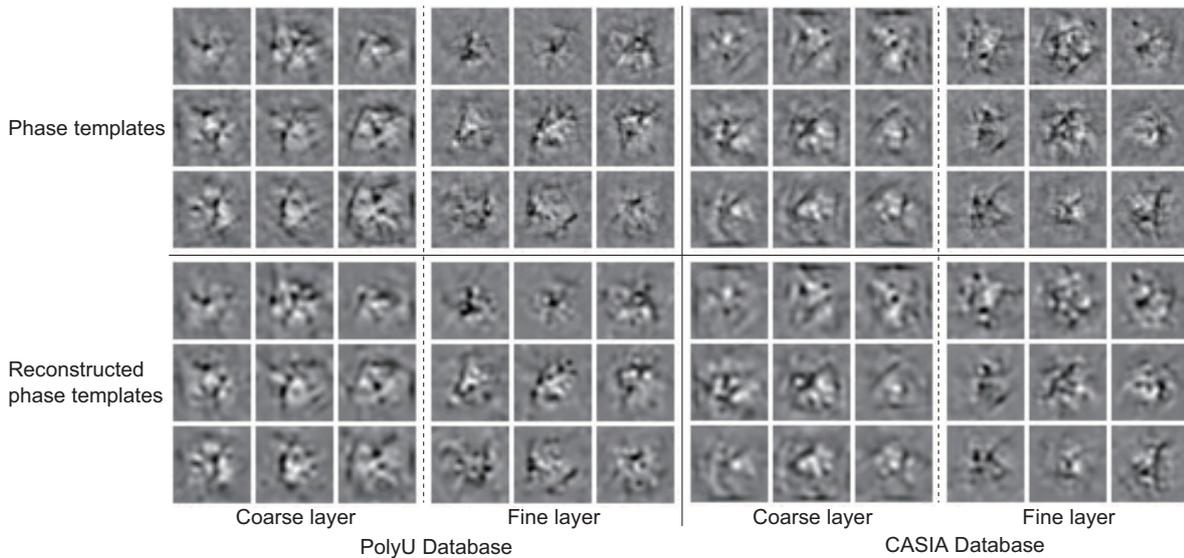


Figure 4.8: Phase templates in spatial domain and its reconstructed versions with $N_d = 16$ and $N_c = 32$.

Table 4.5 compares the data size required to enroll a reference image in the gallery for each method. The phase-based methods (iv)–(vi) enroll a reference image as $N_b \times (\# \text{ of layers}) = 49 \times 2$ phase templates. In the case of PB-HBM and PB-CM, we adopt 4-bit quantization of phase components for storage efficiency [17, 60, 84]. In the case of PB-HBM-CSC, a non-zero element of the sparse codes can be stored using 6 bytes, i.e., 32 bits for a coefficient value and 16 bits for its position and filter index. As is observed in Table 4.5, the data size required to enroll a reference image for PB-HBM-CSC is smaller than those required for PB-HBM and PB-CM. This data size is still considerable higher than those for conventional

methods. Even so, we can say that 1,000 users can be easily enrolled in an embedded system storage since it requires only 19 Megabytes.

4.6.5 CSC representation

We applied the CSC algorithm to the phase templates in order to obtain a compact CSC approximation and proposed PB-HBM-CSC, which is a computationally efficient phase-based palmprint identification algorithm. Here, we take a look into the CSC approximation of phase templates.

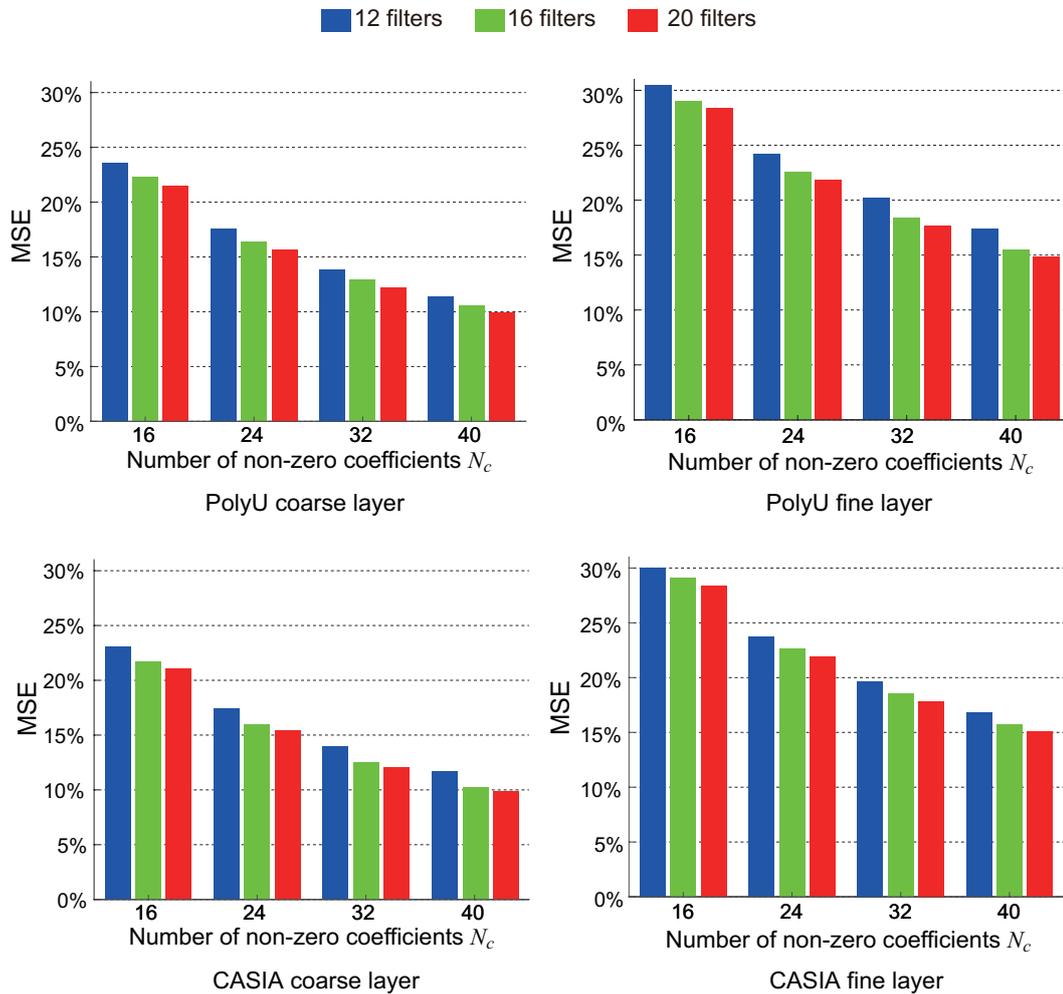


Figure 4.9: Mean square error (%) of the CSC reconstruction of phase templates for number of coefficients $N_c = 16, 24, 32, 40$ and number of filters $N_d = 12, 16, 20$.

In general, we have to select the parameter (N_c, N_d) taking into account the balance between computation time and identification accuracy. An interesting observation is that our setting, which is $(N_d, N_c) = (16, 32)$, exhibits good identification accuracy as shown in Table 4.4 in spite the fact that this setting does not correspond to the lowest reconstruction

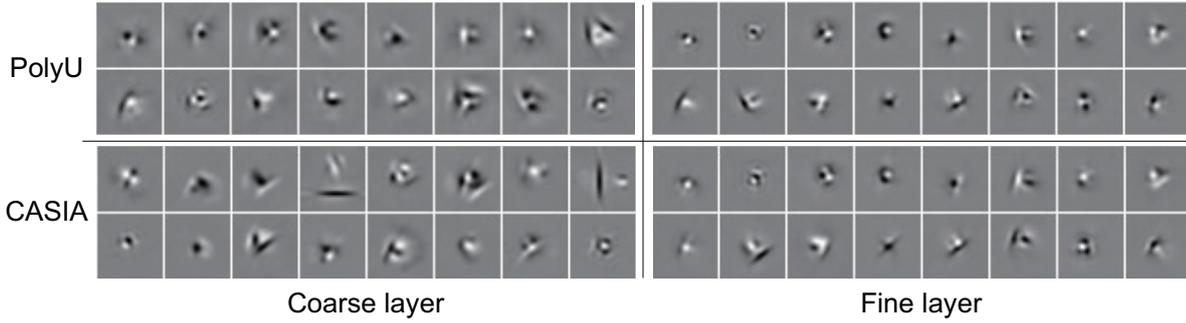


Figure 4.10: Impulse response of filters used in PB-HBM-CSC with $N_d = 16$ and $N_c = 32$.

error in Fig. 4.9. Therefore, a highly accurate CSC reconstruction is not necessary to obtain high identification accuracy.

Fig. 4.8 shows examples of phase templates and corresponding phase templates reconstructed from CSC representation, where $N_c = 32$ and $N_d = 16$. The CSC approximation is similar to apply a low-pass filter to phase templates. The reconstruction error is evaluated using Mean Squared Error (MSE), which is calculated by

$$\text{MSE}(\%) = \frac{1}{N_u} \sum_{i=1}^{N_u} \sum_{n_1, n_2}' |x_i(n_1, n_2) - \hat{x}_i(n_1, n_2)|^2 \times 100\%, \quad (4.28)$$

where $\hat{x}_i(n_1, n_2)$ is the phase template reconstructed by Eq. (4.16). Fig. 4.9 shows reconstruction errors for a variety of (N_c, N_d) .

Fig. 4.10 shows the impulse response of filters used in PB-HBM-CSC. Note here that our CSC algorithm attained a reduced spatial support for the patterns in the impulse responses. This confirms that the structure of phase templates is composed of localized primitive patterns.

4.7 Summary

Motivated by the high accuracy of phase-based palmprint recognition, we proposed a palmprint identification method that employs a sparse representation of local phase templates. This method addresses the large computational complexity of phase-based image matching in the one-to-many comparison required for identification mode by taking advantage of a compact representation of phase templates in convolutional codes. We also presented a convolutional sparse coding algorithm based on matching pursuit that provides such a compact representation. Experimental evaluation demonstrated that the proposed identification method effectively reduces the computation time of phase-based palmprint identification in one order of magnitude without significant degradation of the identification accuracy. In

future work, we will extend the identification framework of the proposed method to other biometric traits, such as face, finger-knuckle-print, periocular region, and iris.

This chapter contains pre-prints of an article submitted on October 3rd 2018 for publication to the IEEE Transactions on Image Processing.

5

Phase-based periocular recognition with texture enhancement

5.1 Introduction

Reliable authentication of individuals in unconstrained scenarios is increasingly being required for applications, such as immigration control, entrance-exit management, surveillance, law enforcement, forensics, etc. [106]. Capturing high-resolution images in less-constrained environments with relaxed cooperation is crucial for user experience in terms of convenience and acceptability [107], and it is relatively easy using state-of-the-art imaging technology. Traditionally, iris and face have been intended for unconstrained scenarios. Face recognition performance has greatly improved in the last decades [13], and iris recognition has arguably the highest performance in controlled settings [70]. However, face recognition applicability is limited since face recognition methods have to deal with factors such as facial expressions, lighting variations, and occlusions in order to achieve accurate authentication. Likewise, iris recognition methods have to deal with different factors, especially, partial occlusions due to specular reflections and eyelashes, non-frontal gaze, motion blur, and defocus blur. Such impairments degrade recognition performance or sometimes prevent recognition at all.

On the other hand, over the last years, the *periocular region* – the extended region around the eye – has received considerable attention [53, 54, 55]. As mentioned in Sec. 2.2, periocular region includes many distinctive components, such as iris, sclera, skin, eyefolds, eyelashes and eyebrows [54]. These components are indicated in Fig. 5.1 for an image sample from the UBIPr database [69]. They allow highly accurate recognition comparable

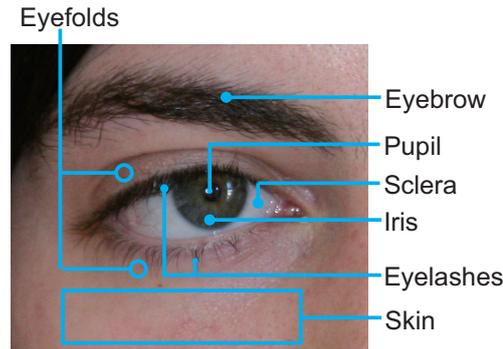


Figure 5.1: Components in the periocular region. Image sample from the UBIPr database.

with iris recognition in non-controlled scenarios [67, 108, 109]. In addition, existing sensing setups for face recognition and iris recognition can be used to perform periocular recognition as well.

Images captured from the periocular region usually exhibit nonlinear deformations due to variations in facial orientation and expression as well as partial occlusion by eyeglasses, hair, etc. These problems have to be addressed through highly accurate image matching techniques in order to achieve an efficient performance. This chapter proposes a periocular recognition algorithm based on the Phase-Based Correspondence Matching (PB-CM) described in Sec. 3.8, which has demonstrated efficient performance in face, palmprint, and finger knuckle recognition [17]. PB-CM employs local block matching using *phase features* obtained from 2D discrete Fourier transform of image blocks. It combines phase features with image resolution pyramids to deal with deformations caused by variations in facial expressions and head pose.

A major problem of PB-CM is that its performance is significantly degraded when it is applied to regions with poor texture such as the skin under the eye. Addressing this problem, we combine phase-based correspondence matching with a texture enhancement technique to make a highly robust recognition algorithm for periocular images. Experimental evaluation using three public databases demonstrates an efficient performance of the proposed algorithm in periocular recognition compared with conventional algorithms.

Contributions of this chapter are summarized as follows: (i) a new periocular image recognition algorithm using phase-based correspondence matching, (ii) a technique for improving its recognition performance through variance normalization, and (iii) systematic experimental evaluation of the algorithms using three public databases. In the next section, we present an overview of related works and the motivation of the proposed method.

Table 5.1: Overview of periocular recognition works with traditional features.

Work	Feature	Database
Miller et al. 2010 [110]	LBP	FGRC[111], FERET[112]
Woodard et al. 2010 [113]	LBP, color histograms	FGRC, MBGC[114]
Park et al. 2011 [115]	LBP, HOG, SIFT	FGRC
Padole et al. 2012 [69]	LBP, SIFT	UBIPr[69]
Ross et al. 2012 [74]	HOG, m-SIFT, PDM(FOSTDF)	FOCS[116]
Uzair et al. 2015 [117]	LBP, PCA, raw pixel	UBIPr, MBGC v2 [114]
Behera et al. 2017 [118]	LBP, HOG	Cross-Eyed 2016 [119]

5.2 Conventional methods for periocular recognition

As explained in 2.3, the recognition process comprises sensing, preprocessing, feature extraction and feature matching. During preprocessing, the system extracts the periocular region using eye or corner-of-eye detection, and normalizes the periocular image, as explained in Sec. 2.3.2. From the preprocessed periocular images, the system extracts discriminative feature vectors which are used in the later stage for comparison.

Previous works on periocular recognition applied traditional features used in biometric recognition, specially in face recognition. See Table 5.1 for a list of previous works. Examples of such works employ Histograms of Oriented Gradients (HOG) [74, 115, 118], Local Binary Patterns (LBP) [69, 110, 113, 115, 117, 118], Principal Component Analysis (PCA) [117] and Scale-Invariant Feature Transform (SIFT) [69, 74, 115], among others. Recognition methods that employ these features are relatively robust to imperfect alignment and changes in facial expression. However, their performance is limited since they do not fully exploit the texture information within the periocular region.

Among recent periocular recognition methods, we find those based on Convolutional Neural Networks (CNNs) [120, 121, 122] and those based on a correlation filter known as Fusion Optimal-Trade-off Synthetic Discriminant Function (FOTSDF) [48, 74, 123, 124]. For comparison purposes, we put our attention on two of these methods, Semantic-Assisted CNN (SCNN) [120] and Periocular Probabilistic Deformation Model (PPDM) [48]. One major disadvantage of these state-of-the-art methods is that they require training data. CNN approaches require large training datasets for generalization which are not available, and the approaches that employ FOTSDF rely on image samples of target users for training and parameter selection.

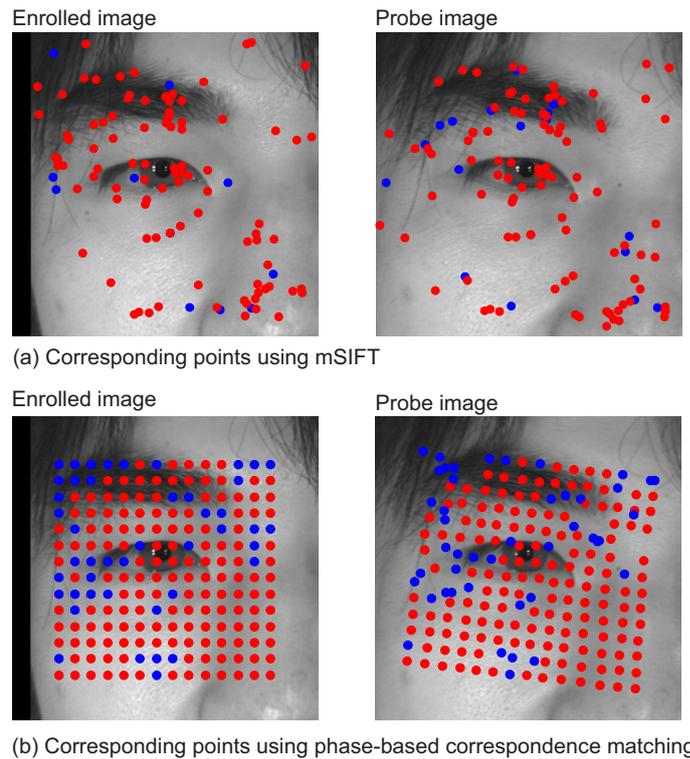


Figure 5.2: Correspondence matching examples for two periocular images of the same person (genuine pair): (a) using m-SIFT features matching and (b) using PB-CM. Red dots indicate corresponding locations successfully estimated, and blue dots indicate failed estimations.

In this chapter, we propose Phase-Based Correspondence Matching (PB-CM) with texture enhancement for periocular recognition. As explained in 3.8, PB-CM can handle various nonlinear transformations by comparing local blocks at their corresponding location in a similar way to SIFT feature matching. Compared with SIFT feature matching, Fig. 5.2(a), PB-CM utilizes precise corresponding locations for an accurate similarity evaluation, Fig. 5.2(b). PB-CM employs phase features which have shown to be effective for representing various biometric textures, such as fingerprint, iris, face, palmprint, and finger-knuckle-print [81]. However, the skin below the eyes has usually a weak texture and does not yield the same recognition performance as other parts of the periocular region. We found out that we can enhance the skin texture with variance normalization and improve the discriminative capacity of the phase features in periocular recognition. In this manner, the novel combination of PB-CM with texture enhancement makes intensive use of the available periocular texture. This allows our method to compete favorably against advanced periocular recognition approaches such as FOTSDF-based [48] and CNN-based [120] methods.

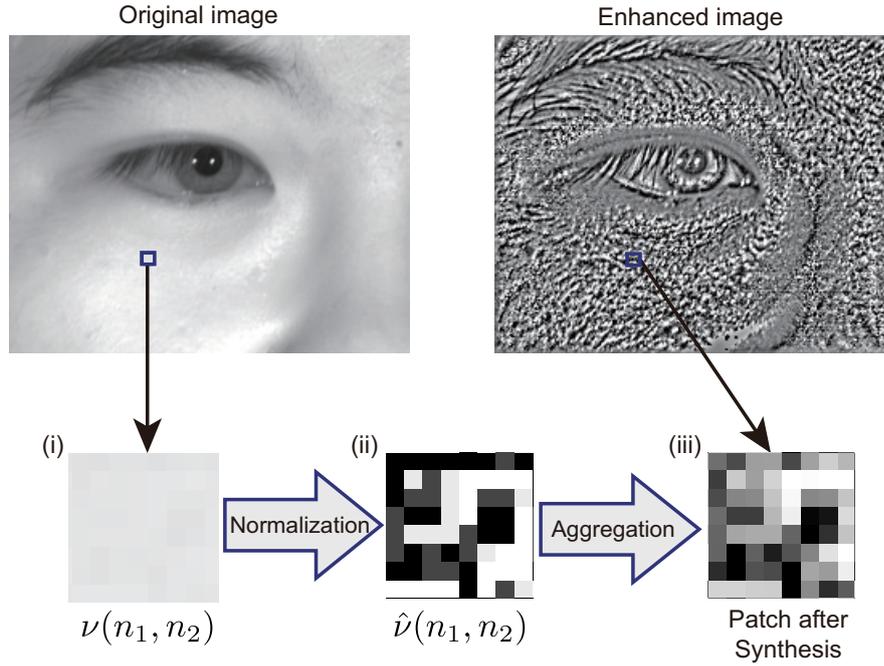


Figure 5.3: Texture enhancement with variance normalization. Consists of three steps: (i) extract one patch per pixel, (ii) normalize the variance of each patch and (iii) combine them into a new image.

5.3 Periocular image matching based on phase features

In this section we describe the proposed texture enhancement strategy and our modifications to PB-CM. We present three similarity measures as well.

5.3.1 Texture enhancement

Phase-based image matching has shown to be effective for recognizing popular biometric traits, such as fingerprint [24], iris [83], palmprint [28, 84], [58, 125] and finger knuckle [58, 125], which contain homogeneous texture components. When we apply phase-based image matching to periocular recognition, however, we have to deal with heterogeneous texture components contained in periocular images. For example, while the appearances of eyebrows have very distinctive textures, the skin around the eyes might have poor textures, which deteriorate the recognition performance. It is necessary to enhance the periocular images so as to make the poor texture more visible and make the whole image more homogeneous. Therefore, we employ variance normalization, also known as contrast normalization [89], which adjusts the local variance of pixel intensity across the whole image

to the same level.

Variance normalization consists of three steps as illustrated in Figure 5.3.

(i) Extract local patches: Extract a small (e.g., 7×7) local patch from the original image for every pixel.

(ii) Normalize the patches: For all the patches, subtract their DC components and normalize their variance. A given patch $\nu(n_1, n_2)$ is normalized as follows:

$$\hat{\nu}(n_1, n_2) = \frac{\nu(n_1, n_2) - \nu^{DC}}{\max\left(\eta, \sqrt{\frac{1}{|C|} \sum_{(n'_1, n'_2) \in C} (\nu(n'_1, n'_2) - \nu^{DC})^2}\right)}, \quad (5.1)$$

where C is the set of patch coordinates, η is a small constant to cancel noise, $\max(., .)$ is a function that returns the maximum value among the arguments, and ν^{DC} is the DC component which is given by

$$\nu^{DC} = \sum_{(n_1, n_2) \in C} \frac{\nu(n_1, n_2)}{|C|}. \quad (5.2)$$

Notice that as long as the variance of $\nu(n_1, n_2)$ is greater than η , the variance of $\hat{\nu}$ is equal to 1, i.e.,

$$\frac{1}{|C|} \sum_{(n_1, n_2) \in C} \left(\hat{\nu}(n_1, n_2) - \sum_{(n_1, n_2) \in C} \frac{\hat{\nu}(n_1, n_2)}{|C|} \right)^2 = 1. \quad (5.3)$$

(iii) Synthesize a new image from normalized patches: construct a new image by aggregating the patches using their original pixel coordinates. The intensity of each pixel in the new image is the addition of overlapping neighbor patches.

The general formulation of the steps (i) and (ii) is called Divisive Normalization Transform (DNT), and it is effective to reduce the statistical dependencies [126]. DNT has been applied to image processing tasks such as image compression [127] and contrast enhancement [128] among others.

In order to illustrate the effect of texture enhancement on the BLPOC function, Fig. 5.4 depicts an example of BLPOC matching without texture enhancement (left side) and another example with texture enhancement (right side). Both examples show the image blocks ($f(n_1, n_2)$ and $g(n_1, n_2)$), the phase features ($X(k_1, k_2)$ and $Y(k_1, k_2)$), and the respective BLPOC functions ($r(n_1, n_2)$). The BLPOC function without enhancement exhibits multiple peaks, and the location of the highest peak does not correspond to the correct translational displacement between the two image blocks. On the other hand, the BLPOC function with texture enhancement produces a single peak, which location indicates the correct translational displacement between the two image blocks.

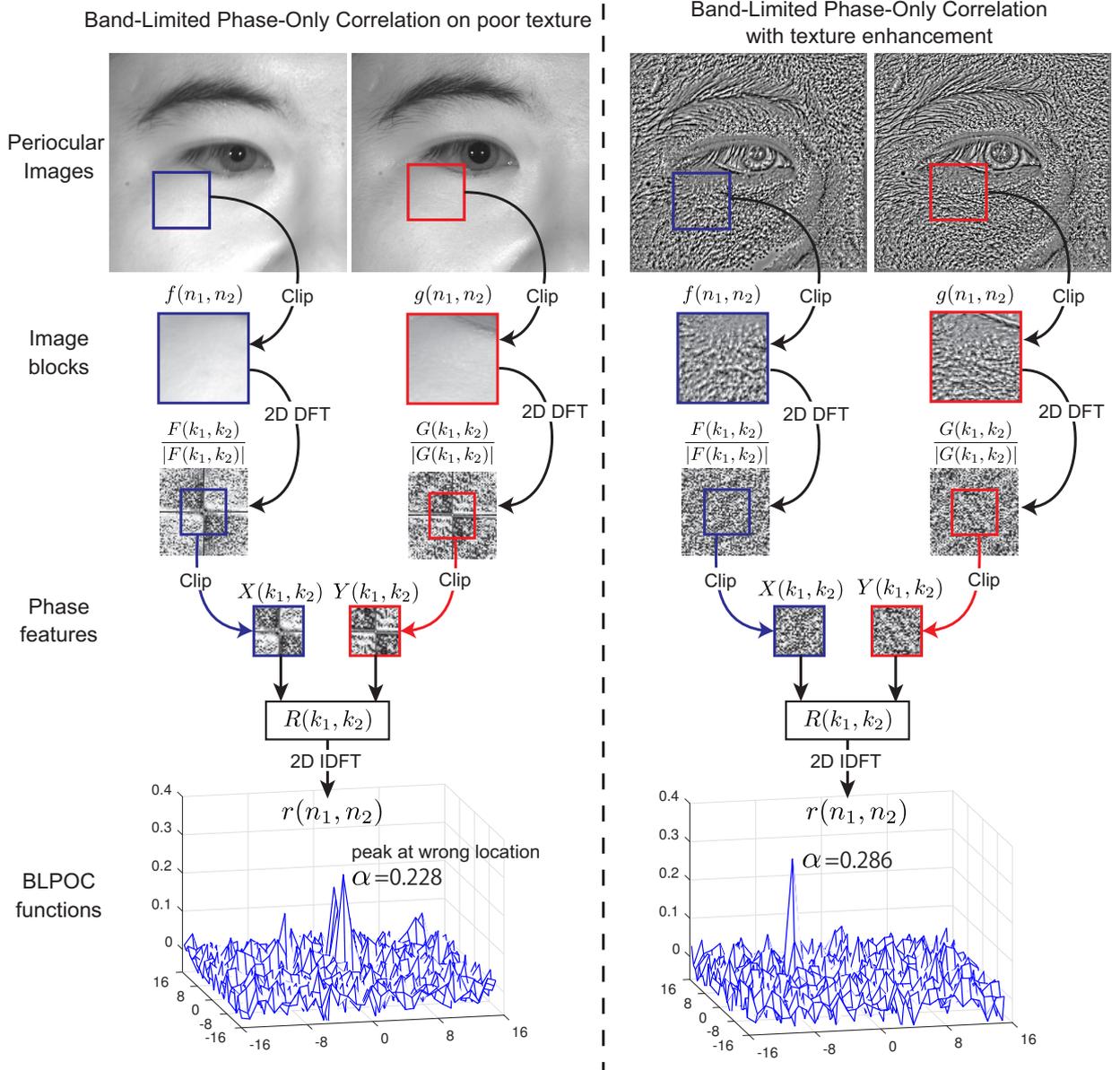


Figure 5.4: Effect of texture enhancement on the BLPOC function for comparison of a genuine block pair with poor texture (left: without texture enhancement, right: with texture enhancement).

The spatial representation of phase feature can be regarded as an enhanced texture. While variance normalization equalizes the pixel variance across the image, phase feature computation equalizes the frequency components of the Fourier spectrum. The combination of both enhancements is the main novelty of this chapter. Variance normalization is important for phase-based image matching because it improves the representation of the texture appearance across the spectrum. Thus, phase features that are computed after variance normalization have high discrimination (for small texture details) at a wider range

of frequency bands than those computed without it.

In order to assess the similarity of periocular images with phase-based image matching, we introduce the texture enhancement step into PB-CM as follows.

5.3.2 Correspondence search

We modify the PB-CM algorithm described in section 3.8 in order to dedicate it to periocular image matching. Our experimental observation shows that correspondence search with two-level resolution pyramid ($l_{max} = 1$), should suffice to a wide range of periocular recognition applications. Instead, Global or local registration in third layer resolution is challenging due to the presence of occlusions and variations in gaze. Nonetheless, detection of eyes and facial landmarks is very mature and can ensure reliable global alignment of periocular images. We include a texture enhancement step in the coarse-to-fine correspondence search described in Sec. 3.8.1. After generating the multi-resolution pyramids (step (i) in Sec. 3.8.1) of the preprocessed images I and J , we apply the texture enhancement described above to each resolution layer. Then, we place the set P of N_b locations for reference blocks and estimate the set Q of corresponding locations (step (ii) and (iii) in Sec. 3.8.1). We also introduce a new similarity evaluation metric *rank-n score* for periocular recognition described in the next subsection.

5.3.3 Similarity evaluation

We evaluate the similarity between the images, I and J , through a block-wise comparison between corresponding blocks. This is to compute the BLPOC functions between the set of blocks at locations P on I and the set of blocks at locations Q on J , so as to find the correlation peak values. So far, we assumed that all the blocks in $Q = \mathbf{q}_1, \dots, \mathbf{q}_{N_b}$ are valid in that they precisely correspond to the blocks in $P = \mathbf{p}_1, \dots, \mathbf{p}_{N_b}$.

In a real situation, however, we have to consider that some blocks in Q are meaningless since periocular images can present different kinds of occlusions, such as glasses, hair, hats and specular reflections, which significantly disturb the matching operation. In particular, specular reflections have a considerable impact on recognition performance [129], which we actually confirmed through experiments. To address this problem, we detect the specular reflections with a simple thresholding operation and discard the blocks (in P and Q) that are mostly covered by reflections. To be precise, in our experiments, a block is discarded when reflections occupies more than 50% of the block area.

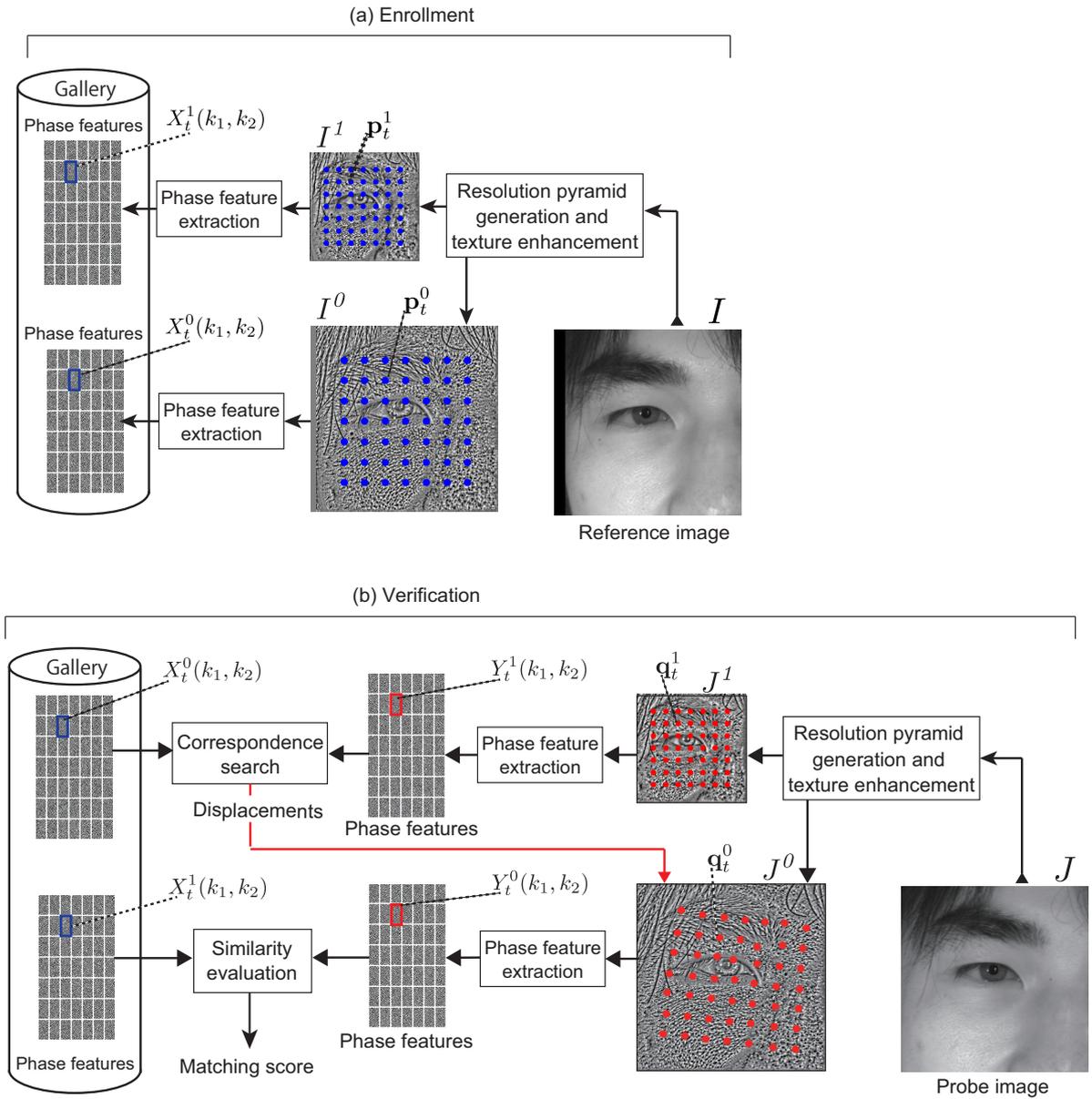


Figure 5.5: Overall of the proposed periocular recognition method. (a) Enrolment procedure of phase-based correspondence matching with texture enhancement. (b) Verification procedure of phase-based correspondence matching with texture enhancement.

We first define reflection masks M_I and M_J for the enrolled image I and the probe image J , respectively, through thresholding:

$$M_I(n_1, n_2) = \begin{cases} 1, & \text{if } I(n_1, n_2) < 252, \\ 0, & \text{otherwise,} \end{cases} \quad (5.4)$$

$$M_J(n_1, n_2) = \begin{cases} 1, & \text{if } J(n_1, n_2) < 252 \\ 0, & \text{otherwise.} \end{cases} \quad (5.5)$$

where 1 indicates valid pixels and 0 indicates possible specular reflections.

In order to weaken the effect of abrupt intensity changes caused by specular reflections, we fill the area with reflections by interpolating inward from the pixel values on the outer boundary of such area¹.

Using the reflection masks, we can select valid blocks for which more than half of the pixels are valid. Fig. 5.6 depicts the block selection for two images with considerable specular reflections(Fig. 5.6(a)). We can observe how the effect of the specular reflection on the glasses is reduced in Fig. 5.6(c). After the block selection, we can determine the valid block pairs (p_t, q_t) where both of two blocks are valid in I and J , respectively. Let V be the set of all the indices t for the valid block pairs (p_t, q_t) , i.e., $V = \{t | (\mathbf{p}_t, \mathbf{q}_t)\}$ is a valid block pair, where $\mathbf{p}_t \in P$ and $\mathbf{q}_t \in Q$. Then, the correlation peak value α_t of the BLPOC function between the block pair $(\mathbf{p}_t, \mathbf{q}_t)$ is said to be valid if and only if the block pair is valid, i.e., $t \in V$.

We consider three measures for similarity (i.e., matching scores) between I and J using the valid correlation peak values. The first measure is the straightforward average of the valid peak values α_t as

$$S_{\text{average}} = \frac{1}{|V|} \sum_{t \in V} \alpha_t. \quad (5.6)$$

The second measure is the n -rank of peak values ordered from highest to lowest:

$$S_{\text{rank}} = \alpha_{t_n}, \quad (5.7)$$

where the valid peak values are sorted as $\alpha_{t_1} \geq \dots \geq \alpha_{t_n} \geq \dots \geq \alpha_{t_{|V|}}$. The third measure is the number of peak values that are greater than a threshold value Thr as

$$S_{\text{thr}} = \sum_{t \in V} h(\alpha_t), \quad (5.8)$$

where

$$h(x) = \begin{cases} 1, & \text{if } x \geq Thr, \\ 0, & \text{otherwise.} \end{cases}$$

¹For example MATLAB provides the *regionfill* function for this purpose.

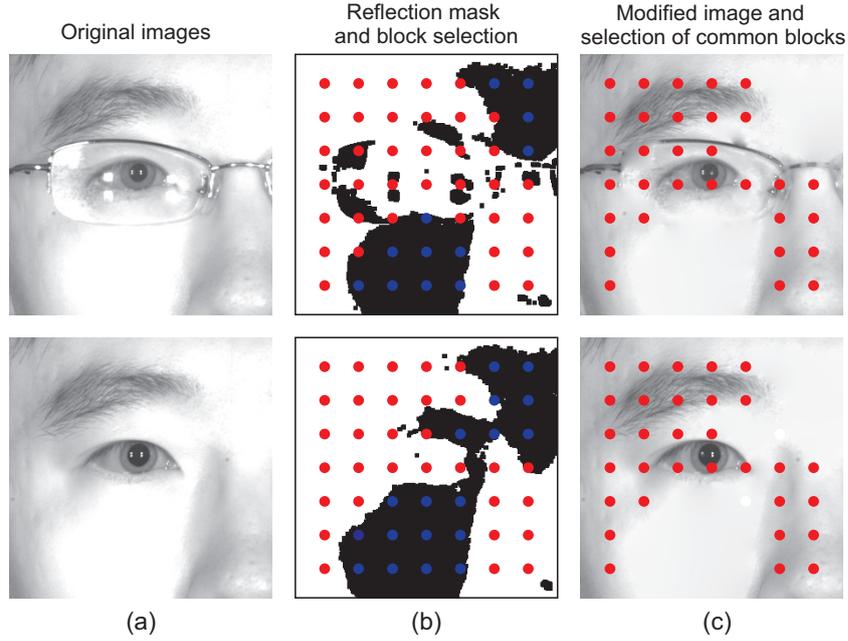


Figure 5.6: Block center selection using thresholding to detect reflections: (a) The original image pair. (b) The reflection masks and block locations with selected locations indicated by red dot and the discarded locations indicated by blue dots. (c) The image after filling the areas that have reflections and the final block locations used for matching

5.4 Overall periocular recognition algorithm

Finally, we describe our proposed algorithm for periocular recognition using PB-CM with texture enhancement. Fig. 5.5 shows the overall of the recognition method. The enrolment procedure is depicted in Fig. 5.5(b) and described by Algorithm 11. During enrolment, we register a given periocular image as an array of local phase features extracted from its multi-resolution image pyramid with texture enhancement. A phase feature extracted from the enhanced enrolled image I^l at position \mathbf{p}_t^l is denoted by $X_t^l(k_1, k_2)$.

The verification procedure is depicted in Fig. 5.5(a) and described by Algorithm 12. During verification, given a probe image J , we extract a set of phase features from its multi-resolution image pyramid with texture enhancement. We compare these phase features with the phase features in the gallery using correspondence search and similarity evaluation. Here, $Y_t^l(k_1, k_2)$ denotes a phase feature extracted from the enhanced probe image J^l at a position \mathbf{q}_t .

Regarding the data size of phase features stored in the gallery, we stored only a reduced portion by taking advantage of the spectrum symmetry and including only the band required for BLPOC computation. In addition, we applied quantization to the phase angles (see [17, 84] and Sec. 3.6 for details). Phase quantization has shown little to non-detriment of

Algorithm 11 Enrolment procedure

Require: An image I to be enrolled**Ensure:** A set of phase features $X_t^l(k_1, k_2)$ for I Generate the reflection mask $M_I(n_1, n_2)$ by Eq. (5.4)Fill the area with specular reflections on I as described in Sec. 5.3.3Generate a two-layer resolution pyramid and apply texture enhancement, so to obtain I^0 and I^1 , as described in Sec. 3.8.1 and Sec. 5.3.1**for** $t \in \{1, 2, \dots, N_b\}$ **do** **if** the block at location \mathbf{p}_t^0 has more than 50% of valid pixels according to $M_I(n_1, n_2)$ **then** Extract a block from I^0 at \mathbf{p}_t^0 Compute the phase feature $X_t^0(k_1, k_2)$ by (3.44) Extract a block from I^1 at \mathbf{p}_t^1 Compute the phase feature $X_t^1(k_1, k_2)$ by (3.44) **end if****end for**Store the computed phase features in the system's gallery

the recognition performance by carefully considering the number of bits used.

5.5 Experimental evaluation

This section describes a performance evaluation of the proposed method and two baseline methods, one based on LBP [73] and the other based on SIFT [11], using three publicly available databases: CASIA-Iris-Distance in the CASIA Iris Image Database Version 4.0 [130] (CASIA), UBIPr database [69] (UBIPr), and ocular still challenge of the NIST FOCS dataset [116] (FOCS). In addition, we compare the recognition performance of our method with the performances reported in the literature for advanced periocular recognition methods [48, 120].

For the comparative evaluation, we present the Receiver Operating Characteristic (ROC) curve which is a plot of the FRR against the FAR, as described in Sec. 2.5. In all of our experiments, we measured the verification performance using the Equal Error Rate (EER), which is the error rate when FAR(%) is equal to FRR(%). We compute the EER(%) values by linear interpolation of the ROC curves.

Algorithm 12 Verification procedure

Require: A probe image J **Ensure:** Matching score between J and the enrolled image I Generate the reflection mask $M_J(n_1, n_2)$ by Eq. (5.5)Fill the area with specular reflections on J as described in Sec. 5.3.3Generate a two-layer resolution pyramid and apply texture enhancement, so to obtain J^0 and J^1 , as described in Sec. 3.8.1 and Sec. 5.3.1Initialize the corresponding points \mathbf{q}_t^1 ($t = 1, 2, \dots, N_b$) on J^1 by Eq. (3.61)Initialize the set V , i.e., $V = \{\emptyset\}$ **for** $t \in \{1, 2, \dots, N_b\}$ **do** **if** there is a phase feature $X_t^1(k_1, k_2)$ in the gallery **then** Extract a corresponding block from J^1 at \mathbf{q}_t^1 Compute the phase feature $Y_t^1(k_1, k_2)$ by Eq. (3.45) Compute the BLPOC function between $X_t^1(k_1, k_2)$ and $Y_t^1(k_1, k_2)$ by Eq. (4.3) and Eq. (3.17) Derive the translational displacement δ_t^1 Determine \mathbf{q}_t^0 from \mathbf{q}_t^1 and δ_t^1 by Eq. (3.60) Extract a corresponding block from J^0 at \mathbf{q}_t^0 **if** the block at \mathbf{q}_t^0 has more than 50% of valid pixels according to $M_J(n_1, n_2)$ **then** Compute the phase feature $Y_t^0(k_1, k_2)$ by Eq. (3.45) Compute the BLPOC function between $X_t^0(k_1, k_2)$ and $Y_t^0(k_1, k_2)$ by Eq. (4.3) and Eq. (3.17) Derive the peak value α_t and add t to the set V **end if** **end if****end for**Compute the matching score between I and J from the values α_t ($t \in V$) using one of the similarity measures described in Sec. 5.3.3

5.5.1 Periocular databases

The images of the CASIA database were collected by the Chinese Academy of Sciences' Institute of Automation [130]. This database consists of 2,567 partial face images taken from 142 subjects under near infrared illumination, where the subjects stand at a distance around 3m from the camera. The size of this images is $2,352 \times 1,728$ pixels (width \times height) and they cover from the mouth to the forehead of the subjects. The images contain small variations in head pose and occlusions due to hair, eyeglass, and specular reflections. When the tilt (or rotation of the face) is more than six degrees, we correct it by aligning the eyes position horizontally. We scaled the images to one-fourth of the conventional image resolution and cropped two periocular regions of 300×300 pixels for each eye.

The University of Beira Interior Periocular (UBIPr) database [69] consists of 10,950 periocular images captured in the visible spectrum from 342 subjects. Each subject has images at five different resolutions: 501×401 pixels at 8m (stand-off distance), 561×441 pixels at 7m, 651×501 pixels at 6m, 801×651 pixels at 5m and 1001×801 pixels at 4m. We scale all these images to a common size of 240×300 pixels and transform them into gray-scale.

The Face and Ocular Challenge Series (FOCS) database [116] consist of 9,581 periocular images of 136 subjects where 4,792 images are from left eyes and 4,789 images are from right eyes. captured with a near-infrared camera at an image resolution of 750×600 pixels. These periocular images were extracted from video sequences of subjects while walking. This database contains images with blur, occlusion and gaze deviation. The images also exhibit drastic variations in illumination and sensor noise. We scale these images to a size of 300×240 pixels.

5.5.2 Baseline methods

In order to compare the recognition performance of the proposed method, we implemented two baseline recognition methods, which employ well known image descriptors: Local Binary Patterns (LBP) [10] and modified Scale-Invariant Feature Transform (m-SIFT) [74]. The first method employs the LBP operator [131], which assigns a label to every pixel of an image by thresholding a circular neighborhoods of each pixel with the center pixel value and considering the result as a binary number 2.4. Then, the histogram of the labels is used as a texture descriptor [131]. Periocular recognition is performed by block-wise comparison of the local histograms. We tested two block sizes for the histogram computation, which are 30×30 pixels for small size and 50×50 pixels for large size. In the experiments with the CASIA database, images are divided into 10×10 (column \times row) for small blocks and

6×6 for large blocks. In the case of FOCS and UBIPr databases, we divided images into 10×8 for small blocks and 6×5 for large blocks. We use MATLAB implementation of LBP¹. The second method is the modified SIFT (m-SIFT) method [74], a biometric recognition method based on SIFT features [11]. In our experiments, m-SIFT is implemented using the VLfeat library² like Ross et al. [74]. For a fair comparison, we also applied the reflection mask described in Sec. 5.3.3 to select or discard features for the two baseline methods. If the circular region around a SIFT key-point has less than 70% of valid area, that key-point is discarded, and if an LBP block has less than 70% of valid area, that block is discarded.

5.5.3 Effect of texture enhancement on the recognition accuracy

We employed the CASIA database to assess the effect of texture enhancement since CASIA images are larger than those of the other two databases and display richer skin textures. The parameters for PB-CM and the texture enhancement are presented in Table 5.2. Texture enhancement affects the appearance of the images with a whitening-like effect, which intensifies the texture representation in the higher frequency bands. Hence, the use of wider BLPOC bandwidth compared with the usual 50% bandwidth [17] can improve verification performance. For this reason, we consider two band-limitation setups for BLPOC and three patch size for texture enhancement. Table 5.3 presents the EERs for the aforementioned configurations. We observe consistent improvement using texture enhancement for both eyes. The introduction of texture enhancement and 67% BLPOC bandwidth can reduce EER to less than one-third of the original case (50% bandwidth) and no enhancement. Considering these results, in the following experiments, we employ a patch size of 5×5 pixels and a bandwidth of 67%. For this configuration, the similarity measures S_{thr} and S_{rank} consistently outperform S_{ave} , and hence we omit S_{ave} in the following experiments.

5.5.4 Quantization of phase features

We evaluated the effect of phase quantization on the recognition performance using the CASIA database. We used the parameters shown in Table 5.2 and selected a patch size of 5×5 pixels for texture enhancement and 67% bandwidth for BLPOC. Table 5.4 shows the EERs for different quantization levels of phase angles: 2bits, 3bits, and 4bits. We did not observe significant degradation of recognition performance due to quantization. However, in the case of 2-bit quantization, the results are not consistent for both eyes. We observe little degradation of the recognition performance for the left eye, while slight improvement

¹ <http://www.cse.oulu.fi/CMV/Downloads/LBPSoftware>

² <http://www.vlfeat.org/>

Table 5.2: Parameters for PB-CM and texture enhancement. Parameters indicated with * are used in the comparative evaluations

PB-CM	
Parameter	Value
Block size	48×48 pixels*
Horizontal/vertical spacing between blocks	16 pixels*
Number of blocks	(14 \times 14) blocks (where we removed 5 blocks corresponding to the upper eyelid)
Band limitation for BLPOC (two different setups)	50% bandwidth: 24×24 phase components 67% bandwidth: 32×32 phase components
Texture enhancement	
Parameter	Value
Noise rejection parameter η	$0.0004 \times C \times 255^*$ where $ C $ is the size of the patch
Patch sizes (three different setups)	3×3 pixels, 5×5 pixels* and 7×7 pixels
Similarity measure	
Parameter	Value
<i>Threshold</i> of $S_{\text{threshold}}$ (varies depending on the BLPOC bandwidth)	0.30 for 50% bandwidth 0.28* for 67% bandwidth
Rank of S_{rank} (varies depending on the BLPOC bandwidth)	12 for 50% bandwidth 8* for 67% bandwidth

Table 5.3: Verification performances EERs(%) of phase-based image matching with and without texture enhancement. Bold fonts indicate the best EER(%) per similarity measure.

Bandwidth	Texture enhancement	Left eye		Right eye	
		S_{average}	S_{rank}	S_{average}	S_{rank}
50%	without enhancement	0.415	0.451	0.472	0.313
	3×3 -pixel patch	0.237	0.197	0.210	0.116
	5×5 -pixel patch	0.237	0.179	0.227	0.098
	7×7 -pixel patch	0.250	0.205	0.237	0.102
67%	without enhancement	0.442	0.624	0.484	0.487
	3×3 -pixel patch	0.235	0.170	0.180	0.103
	5×5 -pixel patch	0.223	0.157	0.145	0.080
	7×7 -pixel patch	0.250	0.165	0.178	0.081
				S_{average}	$S_{\text{threshold}}$
				0.085	0.351
				0.085	0.128
				0.085	0.108
				0.085	0.123
				0.085	0.272
				0.085	0.109
				0.085	0.077
				0.085	0.075

Table 5.4: EERs (%) of phase-based image matching with and without quantization of phase angles. Bold fonts indicate the best EER per similarity measure.

Phase quantization	Left eye			Right eye		
	S_{ave}	S_{rank}	S_{thr}	S_{ave}	S_{rank}	S_{thr}
No	0.223	0.157	0.145	0.085	0.080	0.077
4 bits	0.219	0.143	0.150	0.087	0.075	0.078
3 bits	0.223	0.143	0.165	0.089	0.089	0.084
2 bits	0.257	0.179	0.210	0.080	0.071	0.088

is observed for the right eye. For the rest of our experiments, we choose 4-bit quantization because it is a good balance between the degree of compression and recognition performance. Also, 3-bit quantization is a valid consideration when the registered data size is the main concern.

5.5.5 Comparative performance evaluation

We compared our proposed method with the baseline methods for the three databases. We adopt the parameters marked with * in Table 5.2. The number of blocks for PB-CM are changed depending on the database used, i.e., 14×14 for CASIA and 14×11 for FOCS and UBIPr. We also compare the performance of the proposed method without texture enhancement in order to demonstrate the effectiveness of texture enhancement in periocular recognition. Table 5.5 (a), (b) and (c) summarize the verification performance for the CASIA, UBIPr, and FOCS databases, respectively. Fig. 5.7 compares the ROC curves of the proposed method with S_{rank} , the proposed method with texture enhancement and S_{rank} and the conventional methods using m-SIFT and LBP (the best case) for the three databases. The proposed method with texture enhancement outperforms m-SIFT, while m-SIFT outperforms LBP for the three databases. The proposed method with texture enhancement outperforms that without texture enhancement in all the cases. Hence, we discuss the experimental results only for the proposed method with texture enhancement in the following.

As for the CASIA database shown in Table 5.5(a), the proposed method exhibits EERs one order of magnitude lower than those of m-SIFT. As observed at FAR=0.01% in Fig. 5.7 (a) and (b), the proposed method has a significantly low FRR, i.e., 0.23% for left eye and 0.09% for right eye, compared with m-SIFT, i.e., 9.6% for left eye and 6.4% for right eye. The CASIA database has an advantage for the proposed method due to its relatively

good quality images with some skin texture. This database is also acceptable for m-SIFT, since the images have around 400 m-SIFT key-points on average.

As for the UBIPr database shown in Table 5.5(b), the proposed method and m-SIFT performed relatively close, while LBP performed poorly. Unlike the CASIA database, the UBIPr database contains images with significant large head-pose variation specially in yaw rotations, which are difficult to address. This is a relatively advantageous scenario for m-SIFT due to its robustness against image deformation. Also, the UBIPr images contain a significant number of SIFT key-points, where around 1,200 key-points in average per image are detected. Nonetheless, as observed at FAR=0.1% in Fig. 5.7 (c) and (d), the proposed method exhibits significantly low FRRs, i.e., 9.1% for left eye and 6.5% for right eye, with respect to m-SIFT, i.e., 16.7% for left eye and 10.29% for right eye.

As for the FOCS database shown in Table 5.5 (c), all the methods performed poorly. The proposed method performs slightly better than m-SIFT in terms of EER. This database is highly challenging, since it contains images with substantial blur and noise.

5.5.6 Comparison with methods based on trained features

A major difficulty of periocular recognition is that the skin under the eyes has considerably weak texture and hence conventional methods of biometric feature representation could not capture its inherent features. Another possibility to address this problem would be to use state-of-the-art machine learning techniques with a training data set. In order to enhance the credibility of this chapter’s experimental evaluation, in this section, we additionally consider two methods: one is based on Convolutional Neural Networks (CNNs), which are gaining traction also in biometric applications [15, 16], and the other is based on correlation filters which have been studied on periocular recognition [48, 67, 74, 123, 124, 133]. Specifically, we compared with “Semantic assisted CNN (SCNN)” proposed by Zhao and Kumar [120] and “Periocular Probabilistic Deformation Model (PPDM)” proposed by Smereka et al. [48].

We prepared a subset of the CASIA database that imitates the setup used in [120] for SCNN, although we used a different image segmentation. Table 5.6 presents the parameters of this setup, and Table 5.7 shows the resulting EERs(%) of our method and SCNN implementation¹. The error rates of our method are one order of magnitude lower than those of SCNN. We assumed that our segmentation was favorable to SCNN since in our experiments SCNN yielded an EER=4.32% which is lower than the 6.61% reported in the literature [120]. Therefore, the comparison in Table 5.7 should be fair. Note that the EERs

¹<http://www4.comp.polyu.edu.hk/~csajaykr/scnn.rar>

Table 5.5: EERs (%) of the proposed method and conventional methods on CASIA database, UBIPr database and FOCS database, where TE indicates texture enhancement. Bold fonts indicate the best EER.

(a) CASIA database		
Method	Left eye	Right eye
LBP (Block size: 30×30 pixels)	6.067	4.877
LBP (Block size: 50×50 pixels)	4.595	3.920
m-SIFT	2.065	1.710
Proposed with S_{rank}	0.640	0.510
Proposed with S_{thr}	0.610	0.290
Proposed with TE and S_{rank}	0.143	0.075
Proposed with TE and S_{thr}	0.150	0.078

(b) UBIPr database		
Method	Left eye	Right eye
LBP (Block size: 30×30 pixels)	19.59	17.70
LBP (Block size: 50×50 pixels)	30.10	29.88
m-SIFT	5.57	4.15
Proposed with S_{rank}	6.55	5.25
Proposed with S_{thr}	5.19	4.05
Proposed with TE and S_{rank}	3.16	2.87
Proposed with TE and S_{thr}	3.47	3.17

(c) FOCS database		
Method	Left eye	Right eye
LBP (Block size: 30×30 pixels)	35.30	35.45
LBP (Block size: 50×50 pixels)	46.60	43.03
m-SIFT	24.69	25.26
Proposed with S_{rank}	32.64	32.79
Proposed with S_{thr}	32.02	32.46
Proposed with TE and S_{rank}	22.46	25.08
Proposed with TE and S_{thr}	24.67	26.46

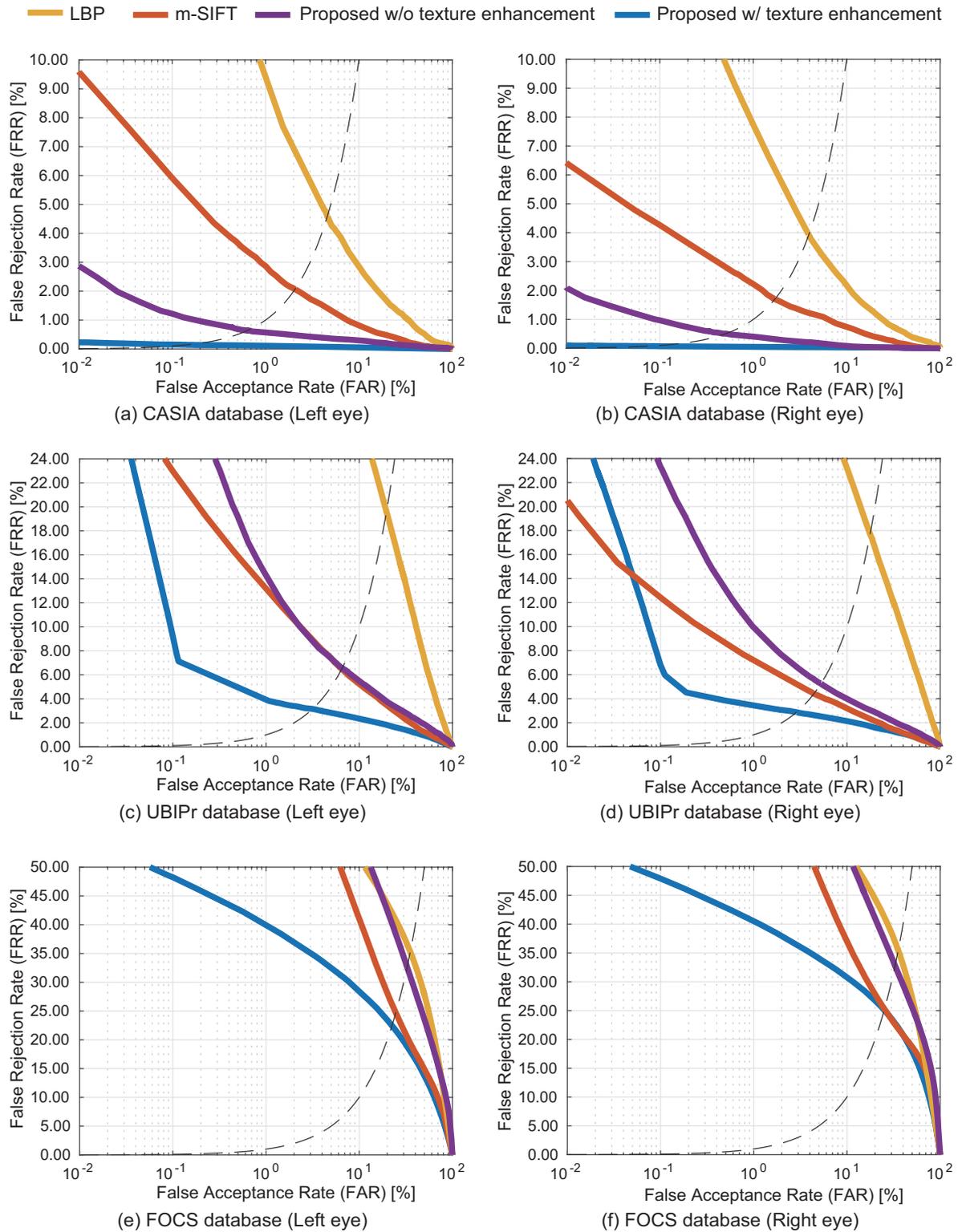


Figure 5.7: Comparison of ROC curves of the proposed method and conventional methods.

Table 5.6: Experimental setup for comparative evaluation with SCNN.

Parameter	Value
Image selection	As provided in [132]
Image resolution	0.294 times the original resolution (estimated from images samples provided in [132])
Image Size	240×240 pixels
About right eye images	Right eye images are flipped horizontally since SCNN is trained for left eye images
Number of blocks for PB-CM	10×10 blocks

Table 5.7: Verification performances EERs(%) on a subset of CASIA database (comparison with SCNN). Bold fonts indicate the best EER.

Method	Left eye	Right eye
SCNN	4.32	4.25
proposed with S_{rank}	0.682	0.237
proposed with $S_{\text{threshold}}$	0.735	0.253

of the proposed method in Table 5.7 differ from those in Table 5.5 due to the differences in the experimental setup.

We also compare our method with PPDM considering the EERs reported in the literature for left eye images. The EERs(%) of PPDM are 10.47% and 7.67% for CASIA database [120] and UBIPr database [48], respectively. These EERs are more than double of those of our method for the respective databases, i.e., 0.68% and 3.16%. We do not consider that the differences in test protocols can be the reason for such a pronounced difference in performance. Therefore, we conclude that our method has a clear advantage over PPDM in periocular image matching.

In the case of FOCS database, for left eye images, the EER of SCNN reported in [120] is 10.47%, and the EER of PPDM reported in [48] is 22.44%. Their experimental evaluation differs in term of preprocessing and selection of test images. However, we can infer that the performance of our method, i.e., EER=22.19% is close to PPDM, while it is clearly worse than the performance of SCNN. This is because our method did not deal with such a high level of noise and blur as is observed in the FOCS images, while CNNs are able to manage a certain degree of feature ambiguity due to image quality degradation. For future works, we will consider introducing deblurring and denoising based on spatio-temporal analysis of video sequences in addition to texture enhancement so as to realize accurate periocular recognition for walking persons in future work.

5.6 Summary

In this chapter, we proposed a periocular recognition algorithm using Phase-Based Correspondence Matching (PB-CM) with texture enhancement. By using a specially selected texture enhancement, we improved the recognition performance of phase-based image matching on periocular images. Experimental evaluation on three public databases demonstrated a clear advantage of the proposed method matching periocular images over conventional methods. To the best of the author knowledge, at the date of this dissertation, the proposed method accomplished the highest performance for the CASIA iris-Distance dataset and the UBIPr database.

This chapter contains pre-prints of an article to be published on the IEICE Transactions on Fundamentals, Vol. E102-A, No.10, Oct. 2019.

6

Conclusion

We have developed two high-performance recognition methods that employ phase-based image matching. The first method is a palmprint identification method that employs an efficient one-to-many matching strategy of phase features. This strategy is derived from convolutional sparse coding of phase templates. In our experimental evaluation, the proposed identification method exhibits a computation time competitive with fast methods and no detriment in the high recognition accuracy of phase-based methods. Our method can accomplish palmprint identification under one-tenth of the time yielded by conventional phase-based recognition methods. The second method is a periocular recognition method that improves the recognition accuracy of conventional phase-based methods by applying texture enhancement. To the best of the author knowledge, at the date of this dissertation, the proposed method exhibits the highest recognition accuracy reported for two public databases.

The contributions presented in this dissertation expand the applicability of recognition using phase-based image matching in two scenarios. They are the following: scenarios for identification (one-to-many recognition) and scenarios that demand facial recognition with high accuracy and low constrained setups. Considering the high performance of the presented methods, this dissertation makes a solid contribution to the state of the art in the field of biometric recognition.

6.1 Future work

The aforementioned contributions of this dissertation expand the range of applicability of biometric recognition using phase-based image matching and opened new directions for research.

For future works, the author considers developing an image restoration method for biometric recognition. This is sought for face, iris, and periocular recognition. The aim is to attain deblur and denoising sufficiently fast for practical biometric applications.



Score-level fusion of multi-biometrics

A.1 Introduction

Biometric systems that utilize one modality (one form of recognition) fail to exhibit high performance in real-life application, specially, those with a large number of users. This is because different problems. Imperfect acquisition increases the noise level; preprocessing can fail on images acquired with low constrained setups; population of users have difficult providing biometric data (e.g., texture-less fingerprints in elder population); and some people can not provided it due to lack of universality. Moreover, it is easier to target one modality for spoofing attacks than multiples. Some of these problems can be addressed by using multiple modalities such as multiple sensors, multiple image samples, multiple recognition algorithms, or multiple traits. Particularly, in order to overcome most of the limitations within unimodal systems, person authentication systems that make use of multiple biometric traits have attracted considerable interest in the last decade [32].

Multibiometric systems improving performance by the complementary use of the information provided by multiple traits. This approach has multiple advantages: (A) It address limited population coverage, (B) hinder spoofing by impostors, and (C) assess noise in sensed data, which are previously unmanageable with unibiometric systems. Fusion levels for multibiometric systems can be classified into five categories: (i) sensor level, (ii) feature level, (iii) score level, (iv) rank level, and (v) decision level. In this chapter, we focus on score level fusion of multiple biometric traits, since matching scores are accessible and relatively simple to combine regardless of the algorithms or traits used.

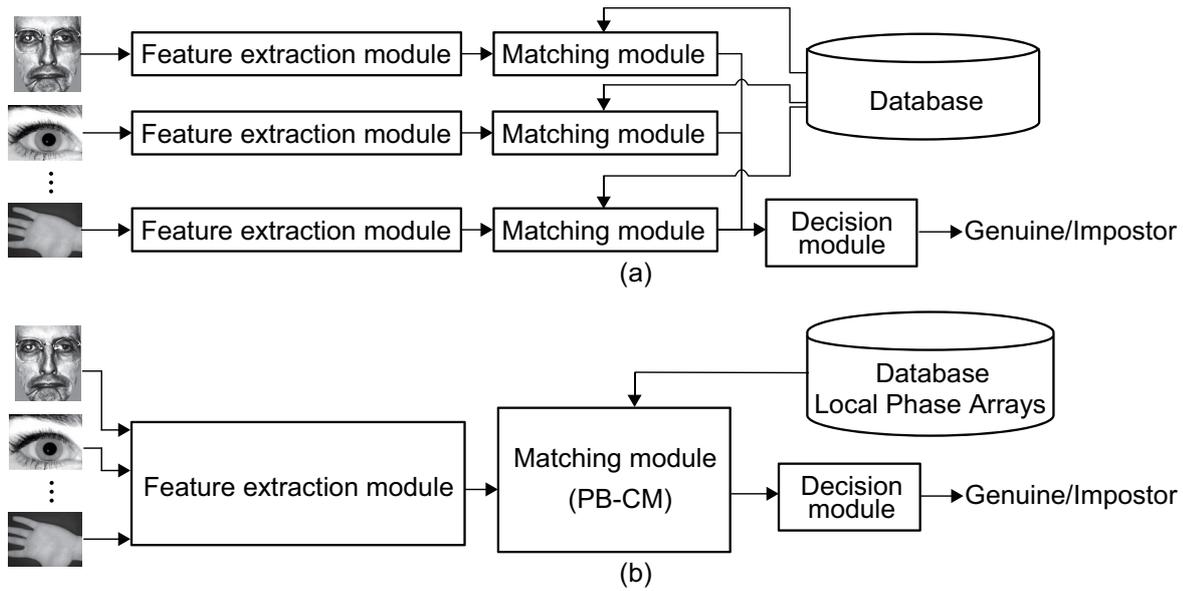


Figure A.1: Multibiometric systems: (a) conventional system using multiple matchers, and (b) proposed system using PB-CM (local phase array).

In Ch. 2 we covered the fundamentals of biometric systems, the modules of their operation flow with a single trait: an acquisition module, a preprocessing module, a feature extraction module, and a matching module, which is also referred as matcher. In general for multibiometric systems with score-level fusion, the matching score for each trait is calculated using a trait-dedicated recognition algorithm. This case requires a feature extraction module and matching module for each trait. Therefore, the increase in the number of traits results in a large-scale system as shown in Fig. A.1(a). In contrast, multibiometric recognition using a unified recognition algorithm such as PB-CM is expected to realize a compact system as shown in Fig. A.1(b).

In chapter 3 we described the Phase-Based Correspondence Matching (PB-CM) recognition method which has demonstrated its efficiency for face, palmprint and finger knuckle [17]. If the matching score for each trait is calculated by one same matcher using PB-CM, the size of the system can be reduced and the simple score level fusion can be utilized to exhibit high quality performance for person authentication. In this appendix, we consider the score level fusion of face, iris, palmprint and finger knuckle whose matching scores are calculated using PB-CM method described in Sec. 3.8. We already review the biometric traits face, iris, and palmprint in Sec. 2.2.

Through a set of experiments using public databases, we demonstrate the effectiveness of PB-CM for multibiometric recognition compared with the combination of the state-of-the-art recognition algorithms for each trait.

After a review of the PB-CM recognition method in the next section, we describe the score-level fusion approaches considered in this study. We present in Sec. A.3 an extensive experimental evaluation and its respective discussions. Then, we summarize this appendix in Sec. A.4.

A.1.1 Biometric recognition using phase-based image matching

Phase features are suitable to represent various biometric texture which can be compared by the means of the BLPOC function to measure similarity. As we explained in Sec. 3.8, PB-CM combines block-wise comparisons with a coarse-to-fine strategy. In this method, biometric images are represented by an hierarchical array of local phase features, i.e., a Local Phase Array (LPA). We employ a three layers PB-CM and the matching score computed by Eq. 5.6.

A.2 Score fusion approaches

This section describes score fusion approaches considered in this work. Score fusion approaches are broadly classified into 3 approaches: (i) density-based approach, (ii) classifier-based approach, and (iii) transformation-based approach [32]. The density-based approach estimates the Probability Density Function (PDF) of the matching scores for both genuine and impostor pairs of each trait. Then, this approach calculates the combined matching score according to the relation between genuine and impostor PDFs. Given an accurate estimation of these PDFs, the density-based approach can exhibit the best performance in score fusion approaches; however, an accurate estimation is not always possible under practical situations where the amount of training data is limited. Compared with classifier-based approach, transformation-based approach is a simpler way to approximate the relation between the PDFs. We employ the density-based and transformation-based approaches in this dissertation. In the following, the matching scores of face, iris, palmprint and finger knuckle are denoted by S_x ($x \in T = \{F, E, P, K\}$), and the matching score vector is denoted by $\mathbf{S} = [S_F, S_E, S_P, S_K]$, where the high value of S_x indicates the high possibility of genuine match. The set of biometric traits to be fused is indicated by T' as

$$T' \in \mathfrak{P}(T) \setminus \{\phi, \{F\}, \{E\}, \{P\}, \{K\}\}, \quad (\text{A.1})$$

where $\mathfrak{P}(T)$ is a power set of T .

A.2.1 Density-based approach

This approach uses the PDF of matching score S to combine matching scores calculated from different traits. In the training stage, the PDFs $p_x(S | \omega)$ of each $x \in T$ for $\omega \in \{\text{genuine}, \text{impostor}\}$ are estimated from the training data set. The conventional approach is kernel density estimation, also known as Parzen window. The model of this approach depends on all the training data, so it is not convenient for our case. For this reason, we choose instead a powerful parametric method called Gaussian mixture model for representing the PDF $p_x(S | \omega)$, i.e.,

$$p_x(S | \omega) = \sum_{i=1}^{N_g} w_i \exp\left(\frac{-(S - \mu_i)^2}{2\sigma_i^2}\right). \quad (\text{A.2})$$

The parameters $w_i, \sigma_i, \mu_i, N_g$ are learned with a fitting/optimization algorithm. The standard method to fit mixture model of Gaussian functions is expectation-maximization (EM). However, EM cannot estimate the complexity of the model, i.e., the number of components N_g . For this purpose, we use the algorithm proposed by Zivkovic et al. 2004 [134]. This algorithm starts with a high number N_g and searches for the maximum a posteriori solution. During this search a prior discards irrelevant components. In order to solve a convergence problem that arose in our experiments, we add a restriction to the algorithm to the parameters. This restriction truncates the values of w_i , if they become very large, and it truncates the values of σ_i , if they become very small.

In the testing stage, the values of $p_x(S_x | \text{genuine})$ and $p_x(S_x | \text{impostor})$ for \mathbf{S} of the input data. Then, we calculate the combined matching score S_{fusion} fusion as a Likelihood Ratio (LR) between genuine and impostor distributions, i.e.,

$$S_{\text{fusion}} = \frac{p(\mathbf{S} | \text{genuine})}{p(\mathbf{S} | \text{impostor})} = \frac{\prod_{x \in T} p_x(S_x | \text{genuine})}{\prod_{x \in T} p_x(S_x | \text{impostor})}. \quad (\text{A.3})$$

A.2.2 Transformation-based approach

This approach employs simple fusion rules to calculate the combined matching scores by transforming input matching scores of different traits into a common domain. The parameters for score transformation, i.e., score normalization, are calculated from the training data set. In this work, we employ three normalization techniques [135]: (i) Min-max, (ii) Double sigmoid, and (iii) tanh-estimators. Let S'_x denote a transformed element S_x of the matching score vector \mathbf{S} . Then, the combined matching score S_{fusion} is calculated from the normalized matching score vector $\mathbf{S}' = [S'_F, S'_E, S'_P, S'_K]$ using the following simple fusion rules:

- **Average**

$$S_{\text{fusion}} = \frac{\sum_{x \in T'} S'_x}{|T'|}, \quad (\text{A.4})$$

- **Mean Square (MS)**

$$S_{\text{fusion}} = \frac{\sum_{x \in T'} S_x'^2}{|T'|}, \quad (\text{A.5})$$

- **Residuals**

$$S_{\text{fusion}} = 1 - \prod_{x \in T'} (1 - S'_x). \quad (\text{A.6})$$

Score fusion rules MS and (product of) residuals rules are novelties of this study. If the elements of matching score vector \mathbf{S} are in the common domain, score normalization with the training data set should not be required. This is expected if use an unified matching modules with a robust feature vector, such as local phase features.

A.3 Experimental evaluation

This section describes a set of experiments to evaluate the recognition performance of the proposed multibiometric recognition system using PB-CM. We construct a set of virtual multibiometric databases which we utilize together with a statistical technique for a comparative experimental evaluation.

A.3.1 Virtual multibiometric datababases

In order to evaluate recognition accuracy of score level fusion for face, iris, palmprint and finger knuckle recognition algorithms, we utilize a virtual multibiometric database. We generate this database by combining public unimodal databases. In this way one identity of each database is attributed to one virtual subject identity. Performance evaluation with virtual databases is considered to be valid under the assumption that the biometric data of sufficiently distinct traits is statistically independent.

The virtual subjects, i.e., chimera subjects is created by joining together face, iris, palmprint and finger knuckle images from unimodal databases. In the experiments, the number of chimera subjects is 100 with 4 images for each biometric trait. These images are a subset of the initially considered datasets. As for face images, we use 144 subjects with 4 images from FERET database [112]. As for iris images, we use 175 subject with 4 images from Iris Challenge Evaluation 2005 (ICE 2005) database [136], where we have

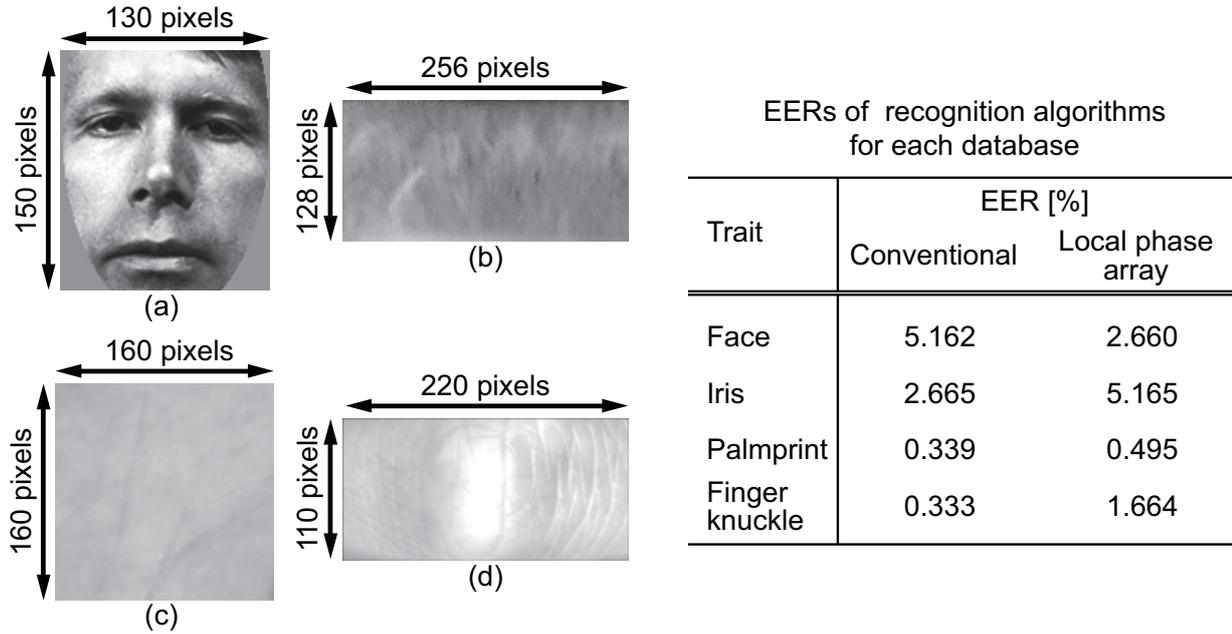


Figure A.2: Examples of ROI images in each database and EERs(%) of each recognition algorithm: (a) face, (b) iris, (c) palmprint, and (d) finger knuckle.

assumed the left eye and right eye of the same person as different subjects. As for palmprint images, we use 600 subjects with 4 images from CASIA Palmprint database [91], where we have assumed the left and right hand of the same person as different subjects. As for finger knuckle images, we employ PolyU FKP database [137] which consists of 7920 images with 165 subjects and 6 different images for each of the left index finger, the left middle finger, the right index finger and the right middle finger in 2 separate sessions. We assume each finger knuckle of the same person as different subjects, i.e., a total of 660 subjects ($= 165 \text{ subjects} \times 4 \text{ fingers}$). We use 660 subjects with 4 images from PolyU FKP database, where 2 images are from the first session and the remaining 2 images are from the second session. Subsequently, for all these subjects, we calculate matching scores of all the possible combinations of genuine pairs using PB-CM. Then, we made a database for each trait by selecting 100 subjects per trait. We select those subjects with the lowest averages of matching scores. This selection eliminates any bias in the choice of subjects that might favor PB-CM. In the following experiments, we combine these 100 subjects to make chimera subjects of virtual multibiometric databases. Fig. A.2 shows example images for each database.

For performance comparison, we employ the following conventional algorithms: Local Phase Quantization [138] for face recognition, Ordinal Code [9] for iris recognition, SIFT [12] for palmprint recognition, and Local-Global Information Combination [139] for finger

knuckle recognition. These algorithms are known to belong to the state-of-the-art algorithms for the corresponding biometric trait. Hereon, we denote the number of 2-combinations from a set of n elements as ${}_nC_2$ such that ${}_nC_2 = \binom{n}{2}$. For each database, we evaluate Equal Error Rate (EER) of conventional and local phase array algorithms using $100 \times {}_4C_2 = 600$ genuine pairs and ${}_{100}C_2 \times 4 \times 4 = 79,200$ impostor pairs. Figure A.2 shows a summary of EERs for conventional algorithms and local phase array for each database. The proposed algorithm exhibits comparable performance with the conventional algorithms specialized for each biometric trait despite selecting worst subjects for PB-CM.

A.3.2 Comparative evaluation of the score level fusion

We evaluate the error rates statistically by using the bootstrap technique, which is a non-parametric method to estimate the confidence interval by random data sampling. We create 100 chimera subjects with 4 biometric traits by randomly combining subjects from each database generated in Sec. A.3.1 For each trait, the number of all the possible combinations of genuine matching scores is ${}_4C_2 = 6$, while the number of all the possible combinations of impostor matching scores between different 2 chimeras is $4 \times 4 = 16$, since one chimera subject has 4 images of each trait. We use 2 genuine scores and 6 impostor scores in the training step and leave the remaining scores for the testing step. We then generate a set of virtual score vectors using the same procedure in [135]. Thus, the total number of genuine combinations of score vectors \mathbf{S} for one chimera subject is $4^4 = 256$, since there are 4 genuine matching scores for each trait. On the other hand, the total number of impostor combinations of score vectors \mathbf{S} between 2 different chimeras is $10^4 = 10,000$, since there are 10 impostor matching scores for each trait. Among the above score vectors, we randomly select 128 genuine pairs and 512 impostor pairs to generate a set of virtual score vectors. Then, for $|T'| = 2$, we apply the score fusion approaches described in Sec. A.2 to $128 \times 100 = 12,800$ genuine pairs and $512 \times {}_{100}C_2 = 2,534,400$ impostor pairs and evaluate EERs calculated from combined matching scores. For consistency, we repeat this experimental evaluation for 100 different combinations of subjects and scores in the virtual database.

Table 1 summarizes EERs for each fusion rule when using all the possible combinations of biometric traits, where EER indicates an average of 100 trials. ‘‘Conventional’’ indicates fusing scores calculated by conventional algorithms, ‘‘LPA (PB-CM)’’ indicates fusing scores calculated by the method described in Sec. 3.8. ‘‘Best single modality’’ indicates lower EER of the respective two traits. ‘‘Similarity’’ indicates a simple combination of the matching score between 2 images, which is given as a value within $[0, 1]$ calculated by each recognition algorithm. From the EERs of Conventional, we can make 3 observations. First, most

of them are higher than those of single modality cases. In particular, the combination of face, whose EER is the highest, and finger knuckle, whose EER is the lowest, shows this tendency. Second, specifically “Similarity + MS” exhibits significantly high EER for face-palmprint and iris-palmprint compared with other fusion rules. This is because distribution of matching scores for palmprint is significantly different from other algorithms. Third, EERs of “Double sigmoid + MS” and “Double sigmoid + Residuals” improve the EERs of other fusion rules. These observations indicate that conventional algorithms have to employ normalization to use simple combination rules and to ensure high performance. For the result of PB-CM, we can make three observations. First, EERs are lower than those of single modality cases, except for the combination of iris and palmprint for some fusion rules. Second, “Similarity + MS” shows the best EER between simple combinations. Third, in this case also, the EERs of “Double sigmoid + MS” and “Double sigmoid + Residuals” improve the EERs of simple combination and the other normalization methods. These observations indicate that PB-CM does not always need normalization since it can employ simple combinations for almost all the combination of traits. In both Conventional and PB-CM, LR exhibits the lowest EERs for most cases. As mentioned in Sect. 3, LR is expected to show the best performance, if the PDFs for genuine and impostor pairs were estimated accurately. However, the transformation-based approaches, “Double sigmoid + MS” and “Double sigmoid + Residuals,” also exhibit efficient performances comparable to the ones of LR. Therefore, these score fusions are robust against limited training data and the diversity of their score distributions compared with LR.

Focusing on “Similarity + Average” and “Double sigmoid + MS/Residuals,” EERs of PB-CM are significantly improved compared with those of Conventional. This result indicates that fusion with different recognition methods requires a complex optimization for score normalization and combination approaches to exhibit similar efficiency of score fusion observed with PB-CM. This is because the optimal score normalization method and optimal fusion rule might be different for each recognition algorithm. As observed above, successful score fusion with PB-CM does not depend on normalization methods and fusion rules as it does with the conventional method. Hence, the use of PB-CM for multibiometric systems makes it possible to improve the performance only with simple combination.

A.4 Summary

This appendix proposed score-level fusion of multibiometric recognition using Phase-Based Correspondence Matching (PB-CM). We considered four biometric traits: face, iris, palmprint, and finger knuckle, which we combined in pairs for an extensive experimental

evaluation. Experimental evaluation demonstrates that simple score fusion approaches are enough to improve consistently recognition performance for PB-CM, whereas for conventional algorithms the improvement depends on elaborated fusion rules. PB-CM makes possible to realize simple and compact multibiometric person authentication systems, since its implementation requires only one matching module and simple score fusion.

This appendix contains reprints of a work published in LNCS Proceedings Pacific-Rim Conference on Multimedia 2015 [60].

Table A.1: EERs(%) for all combinations and fusion rules.

T'		$\{F, E\}$	$\{F, P\}$	$\{F, K\}$	$\{E, P\}$	$\{E, K\}$	$\{P, K\}$	
Conventional	Best single modality	2.665	0.339	0.333	0.339	0.333	0.333	
	LR	0.749	0.222	0.630	0.138	0.270	0.138	
	Similarity	Average	0.973	0.391	0.699	0.240	0.391	0.260
		MS	1.017	1.973	0.706	1.604	0.535	0.791
		Residuals	1.023	0.707	0.645	0.660	0.441	0.332
	Min-max	Average	1.078	0.499	0.677	0.435	0.519	0.194
		MS	0.897	1.722	1.137	1.192	0.790	0.410
		Residuals	0.948	0.804	0.781	0.638	0.605	0.218
	Double sigmoid	Average	1.029	0.321	0.735	0.161	0.448	0.158
		MS	0.820	0.221	0.663	0.082	0.305	0.085
		Residuals	0.826	0.218	0.650	0.077	0.304	0.082
	tanh-estimator	Average	1.024	0.792	0.861	0.739	0.806	0.197
		MS	0.876	0.804	0.854	0.476	0.525	0.196
		Residuals	0.928	0.803	0.856	0.577	0.629	0.195
	LPA (PB-CM)	Best single modality	2.660	0.495	1.664	0.495	1.664	0.495
LR		0.777	0.253	0.603	0.330	0.872	0.270	
Similarity		Average	1.071	0.321	0.732	0.683	1.296	0.427
		MS	0.994	0.276	0.686	0.516	1.223	0.334
		Residuals	1.030	0.294	0.703	0.577	1.260	0.366
Min-max		Average	1.282	0.304	0.742	0.734	1.473	0.379
		MS	1.081	0.302	0.686	0.404	1.316	0.362
		Residuals	1.144	0.300	0.701	0.422	1.370	0.359
Double sigmoid		Average	1.124	0.330	0.860	0.554	1.340	0.461
		MS	0.891	0.214	0.682	0.341	1.027	0.271
		Residuals	0.891	0.216	0.680	0.340	1.030	0.267
tanh-estimator		Average	1.155	0.396	0.768	0.476	1.544	0.495
		MS	1.070	0.281	0.734	0.386	1.407	0.370
		Residuals	1.096	0.327	0.742	0.432	1.442	0.421

References

- [1] J. D. Woodward and P. Higgins, *Biometrics : Identity assurance in the information age*. McGraw-Hill Osborne Media,, 2003. 1, 15, 16
- [2] N. B. S. Project, *Biometric Technology Application Manual Volume One: Biometric Basics*. National Biometric Security Project, 2008, vol. 1. 2
- [3] J. Ashbourn, *Practical Biometrics: From Aspiration to Implementation*, 2nd ed. Springer, 2016. 2
- [4] J. Daugman, *Iris Recognition at Airports and Border-Crossings*. Boston, MA: Springer US, 2009, pp. 819–825. 2
- [5] “Unique identification authority of india: Aadhaar dashboard,” https://uidai.gov.in/aadhaar_dashboard/india.php, accessed: 2019-04-01. 2
- [6] D. Maltoni, D. Maio, A. K. Jain, and S. Prabhakar, *Handbook of Fingerprint Recognition*. Springer, 2003. 3
- [7] J. Daugman, “High confidence visual recognition of persons by a test of statistical independence,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 15, no. 11, pp. 1148–1161, Nov. 1993. 3, 60
- [8] A. Kong and D. Zhang, “Competitive coding scheme for palmprint verification,” *Proc. Int’l Conf. Pattern Recognition*, vol. 1, pp. 520–523, Dec. 2004. 3, 24, 60, 84
- [9] Z. Sun, T. Tan, Y. Wang, and S. Li, “Ordinal palmprint representation for personal identification [representation read representation],” *Proc. IEEE Computer Society Conf. Computer Vision and Pattern Recognition*, vol. 1, pp. 279–284, Jun. 2005. 3, 25, 60, 84, 124
- [10] T. Ahonen, A. Hadid, and M. Pietikäinen, “Face description with local binary patterns: Application to face recognition,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 28, no. 12, pp. 2037–2041, Dec. 2006. 3, 25, 106
- [11] D. Lowe, “Distinctive image features from scale-invariant keypoints,” *Int’l. J. Computer Vision*, vol. 60, no. 2, pp. 91–110, Jan. 2004. 3, 26, 104, 107

- [12] Q. Zhao, W. Bu, and X. Wu, "Sift-based image alignment for contactless palmprint verification," *2013 International Conference on Biometrics (ICB)*, pp. 1–6, June 2013. 3, 124
- [13] S. Li and A. Jain, *Handbook of Face Recognition*. Springer, 2011. 3, 17, 93
- [14] H. Wechsler, *Reliable Face Recognition Methods: System Design, Implementation and Evaluation*, 1st ed. Springer, 2010. 3, 16, 31
- [15] B. Bhanu and A. Kumar, *Deep Learning for Biometrics*. Springer, 2017. 3, 111
- [16] M. Vatsa, R. Singh, and A. Majumdar, *Deep Learning in Biometrics*. CRC Press, 2018. 3, 111
- [17] S. Aoyama, K. Ito, and T. Aoki, "Similarity measure using local phase features and its application to biometric recognition," *Proc. IEEE Computer Society Conf. Computer Vision and Pattern Recognition Workshops*, pp. 180–187, Jun. 2013. 3, 4, 49, 50, 55, 57, 60, 75, 78, 81, 88, 94, 103, 107, 120
- [18] C. D. Kuglin and D. C. Hines, "The phase correlation image alignment method," *Proc. Int'l Conf. Cybernetics and Society*, pp. 163–165, 1975. 4, 38
- [19] A. V. Oppenheim and J. S. Lim, "The importance of phase in signals," *Proceedings of the IEEE*, vol. 69, no. 5, pp. 529–541, May 1981. 4
- [20] K. Takita, T. Aoki, Y. Sasaki, T. Higuchi, and K. Kobayashi, "High-accuracy subpixel image registration based on phase-only correlation," *IEICE Trans. Fundamentals*, vol. E86-A, no. 8, pp. 1925–1934, Aug. 2003. 4, 43, 47
- [21] K. Takita, M. A. Muquit, T. Aoki, and T. Higuchi, "A sub-pixel correspondence search technique for computer vision applications," *IEICE Trans. Fundamentals*, vol. E87-A, no. 8, pp. 1913–1923, Aug. 2004. 4, 55, 60
- [22] M. A. Muquit, T. Shibahara, and T. Aoki, "A high-accuracy passive 3D measurement system using phase-based image matching," *IEICE Trans. Fundamentals*, vol. E89-A, no. 3, pp. 686–697, Mar. 2006. 4
- [23] L. H. Chien and T. Aoki, "Robust motion estimation for video sequences based on phase-only correlation," *Proc. Int'l Conf. Signal and Image Processing (IASTED)*, 2004. 4
- [24] K. Ito, H. Nakajima, K. Kobayashi, T. Aoki, and T. Higuchi, "A fingerprint matching algorithm using phase-only correlation," *IEICE Trans. Fundamentals*, vol. E87-A, no. 3, pp. 682–691, Mar. 2004. 4, 21, 43, 97

-
- [25] K. Ito, A. Morita, T. Aoki, T. Higuchi, H. Nakajima, and K. Kobayashi, "A fingerprint recognition algorithm using phase-based image matching for low-quality fingerprints," in *IEEE International Conference on Image Processing 2005*, vol. 2, Sept 2005, pp. II-33. 4
- [26] K. Ito, A. Morita, T. Aoki, H. Nakajima, K. Kobayashi, and T. Higuchi, "A fingerprint recognition algorithm combining phase-based image matching and feature-based matching," in *Advances in Biometrics*, D. Zhang and A. K. Jain, Eds. Berlin, Heidelberg: Springer, 2006, pp. 316-325. 4
- [27] K. Miyazawa, K. Ito, T. Aoki, K. Kobayashi, and H. Nakajima, "An efficient iris recognition algorithm using phase-based image matching," *Proc. Int'l Conf. Image Processing*, pp. II-49-II-52, Sep. 2005. 4
- [28] K. Ito, S. Iitsuka, and T. Aoki, "A palmprint recognition algorithm using phase-based correspondence matching," *Proc. Int'l Conf. Image Processing*, pp. 1977-1980, Nov. 2009. 4, 55, 60, 75, 78, 81, 84, 97
- [29] K. Ito, T. Aoki, T. Hosoi, and K. Kobayashi, "Face recognition using phase-based correspondence matching," *Proc. IEEE Conf. Automatic Face and Gesture Recognition*, pp. 173-178, March 2011. 4
- [30] A. Jain, P. Flynn, and A. Ross, *Handbook of Biometrics*. Springer, 2008. 5, 11, 12, 18
- [31] A. K. Jain, A. A. Ross, and K. Nandakumar, *Introduction to Biometrics*. Springer, 2011. 5, 11, 12, 13, 16, 17, 18, 19, 26, 27, 32
- [32] A. K. J. Arun A. Ross, Karthik Nandakumar, *Handbook of Multibiometrics (International Series on Biometrics)*, 1st ed. Springer, 2006. 6, 119, 121
- [33] B. V. K. V. Kumar, A. Mahalanobis, and R. D. Juday, *Correlation Pattern Recognition*. Cambridge University Press, 2005. 7, 68
- [34] M. Khosrow-Pour, Ed., *Biometrics: Concepts, Methodologies, Tools, and Applications*. IGI Global, 2017, vol. 3. 11, 60
- [35] J. L. Wayman, "10 - the scientific development of biometrics over the last 40 years," in *The History of Information Security*, K. D. Leeuw and J. Bergstra, Eds. Amsterdam: Elsevier Science B.V., 2007, pp. 263 - 274. 11
- [36] J. R. Vacca, *Biometric Technologies and Verification Systems*. Butterworth-Heinemann, 2007. 11, 15

- [37] A. Bertillon, *Signaletic Instructions Including the Theory and Practice of Anthropometrical Identification*. The Werner Company, 1896. [Online]. Available: https://archive.org/details/signaleticinstru00bert/page/n5_12
- [38] A. K. Jain, A. Ross, and S. Prabhakar, "An introduction to biometric recognition," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 14, no. 1, pp. 4–20, Jan 2004. 13
- [39] D. Zhang, *Palmprint Authentication*. Kluwer Academic Publication, 2004. 15
- [40] D. Zhang, W. Zuo, and F. Yue, "A comparative study of palmprint recognition algorithms," *ACM Computing Surveys*, vol. 44, no. 1, pp. 2:1–2:37, Jan. 2012. 15, 60
- [41] D. R. Ashbaugh, *Quantitative-Qualitative Friction Ridge Analysis: An Introduction to Basic and Advances Ridgeology*. CRC Press LCC, 1999. 15
- [42] D. Barrett, *Palmistry*, 1st ed., ser. The predictions library. Dorling Kindersley, 1995. 15
- [43] M. van Mensvoort, "25 years hand analysis research," <http://www.handresearch.com>, accessed: 2019-04-15. 15
- [44] N. V. Boulgouris, E. Micheli-Tzanakou, and K. N. Plataniotis, *Biometrics: Theory, Methods, and Applications*. Wiley-IEEE Press, 2009. 15, 19
- [45] C. B. Christian Rathgeb, *Iris and Periocular Biometric Recognition*, ser. Iet Security. The Institution of Engineering and Technology, 2017. 15
- [46] J. Daugman, "How iris recognition works," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 14, no. 1, pp. 21–30, Jan 2004. 15
- [47] J. Daugman, "Biometric personal identification system based on iris analysis," <https://patents.google.com/patent/US5291560A/en>, 1984, uS Patent 5291560, WIPO patent 9409446. 16
- [48] J. Smereka, V. Boddeti, and B. Kumar, "Probabilistic deformation models for challenging periocular image verification," *IEEE Trans. Information Forensics and Security*, vol. 10, no. 9, pp. 1875–1890, Sep. 2015. 17, 95, 96, 104, 111, 115
- [49] F. Smeraldi and J. Bigun, "Retinal vision applied to facial features detection and face authentication," *Pattern Recogn. Lett.*, vol. 23, no. 4, pp. 463–475, Feb. 2002. 17, 21

-
- [50] M. Savvides, R. Abiantun, J. Heo, S. Park, C. Xie, and B. V. K. Vijayakumar, "Partial holistic face recognition on frgc-ii data using support vector machine," in *2006 Conference on Computer Vision and Pattern Recognition Workshop (CVPRW'06)*, June 2006, pp. 48–48. 17
- [51] B. Heisele, T. Serre, and T. Poggio, "A component-based framework for face detection and identification," *International Journal of Computer Vision*, vol. 74, no. 2, pp. 167–181, Aug 2007. 17
- [52] U. Park, A. Ross, and A. K. Jain, "Periocular biometrics in the visible spectrum: A feasibility study," in *2009 IEEE 3rd International Conference on Biometrics: Theory, Applications, and Systems*, Sept 2009, pp. 1–6. 17
- [53] G. Santos and H. Proença, "Periocular biometrics: An emerging technology for unconstrained scenarios," *Proc. IEEE Symp. Computational Intelligence in Biometrics and Identity Management*, pp. 14–21, Apr. 2013. 17, 93
- [54] I. Nigam, M. Vatsa, and R. Singh, "Ocular biometrics: A survey of modalities and fusion approaches," *Information Fusion*, vol. 26, pp. 1–35, Nov. 2015. 17, 93
- [55] F. Alonso-Fernandez and J. Bigun, "A survey on periocular biometrics research," *Pattern Recogn. Lett.*, vol. 82, no. 2, pp. 92–105, Oct. 2016. 17, 93
- [56] A. K. Jain, K. Nandakumar, and A. Ross, "50 years of biometric research: Accomplishments, challenges, and opportunities," *Pattern Recognition Letters*, vol. 79, pp. 80 – 105, 2016. 17
- [57] S. Aoyama, K. Ito, T. Aoki, and H. Ota, "A contactless palmprint recognition algorithm for mobile phones," in *Proceedings of International Workshop on Advanced Image Technology 2013*, January 2013, pp. 409–413. 19
- [58] D. Kusanagi, S. Aoyama, K. Ito, and T. Aoki, "A practical person authentication system using second minor finger knuckles for door security," *IPSI Trans. Computer Vision and Applications*, vol. 9, no. 8, pp. 1–13, Mar. 2017. 19, 97
- [59] O. Çeliktutan, S. Ulukaya, and B. Sankur, "A comparative study of face landmarking techniques," *EURASIP Journal on Image and Video Processing*, vol. 2013, no. 1, p. 13, Mar 2013. 19
- [60] L. Marval Pérez, S. Aoyama, K. Ito, and T. Aoki, "Score level fusion of multibiometrics using local phase array," *Lecture Notes in Computer Science (Proc. Pacific-Rim Conf. Multimedia)*, vol. 9315, pp. 215–224, Sep. 2015. 19, 88, 127

- [61] D. Zhang, W. Kong, J. You, and M. Wong, "Online palmprint identification," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 25, no. 9, pp. 1041–1050, Sep. 2003. 20, 60, 85
- [62] W. Li, B. Zhang, L. Zhang, and J. Yan, "Principal line-based alignment refinement for palmprint recognition," *IEEE Trans. on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 42, no. 6, pp. 1491–1499, Nov 2012.
- [63] R. P. Wildes, "Iris recognition: an emerging biometric technology," *Proceedings of the IEEE*, vol. 85, no. 9, pp. 1348–1363, Sep. 1997. 20
- [64] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, vol. 1, 2001, pp. I–511–I–518 vol.1. 21
- [65] P. Viola and M. Jones, "Robust real-time face detection," *Int'l J. Computer Vision*, vol. 57, no. 2, pp. 137–154, May 2004. 21
- [66] D. Bolme, B. Draper, and J. Beveridge, "Average of synthetic exact filters," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 2105–2112, Jun. 2009. 21
- [67] V. Boddeti, J. Smereka, and B. Kumar, "A comparative evaluation of iris and ocular recognition methods on challenging ocular images," *Proc. Int'l Joint Conf. Biometrics*, pp. 1–8, Oct. 2011. 21, 94, 111
- [68] F. Alonso-Fernandez and J. Bigun, "Near-infrared and visible-light periocular recognition with Gabor features using frequency-adaptive automatic eye detection," *IET Biometrics*, vol. 4, no. 2, pp. 74–89, Jun. 2015. 21
- [69] C. N. Padole and H. Proenca, "Periocular recognition: Analysis of performance degradation factors," in *2012 5th IAPR International Conference on Biometrics (ICB)*, March 2012, pp. 439–445. 21, 93, 95, 104, 106
- [70] M. J. Burge and K. W. Bowyer, *Handbook of Iris Recognition*. Springer, 2013. 23, 93
- [71] T. S. Lee, "Image representation using 2d gabor wavelets," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 18, no. 10, pp. 959–971, Oct 1996. 23
- [72] L. Fei, Y. Xu, W. Tang, and D. Zhang, "Double-orientation code and nonlinear matching scheme for palmprint recognition," *Pattern Recognition*, vol. 49, pp. 89–101, Jan. 2016. 25, 60, 84, 85, 86, 87

-
- [73] T. Ojala, M. Pietikäinen, and D. Harwood, "A comparative study of texture measures with classification based on featured distributions," *Pattern Recognition*, vol. 29, no. 1, pp. 51–59, Jan. 1996. 25, 104
- [74] A. Ross, R. Jillela, J. Smereka, V. Boddeti, B. Kumar, R. Barnard, X. Hu, P. Pauca, and R. Plemmons, "Matching highly non-ideal ocular images: An information fusion approach," *Proc. Int'l Conf. Biometrics*, pp. 446–453, Apr. 2012. 27, 95, 106, 107, 111
- [75] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, 2nd ed. New York: Wiley, 2001. 29, 31
- [76] R. Bolle, J. Connell, S. Pankanti, N. Ratha, and A. Senior, *Guide to Biometrics*. New York: Springer, 2003. 31
- [77] D. Maio, D. Maltoni, R. Cappelli, J. L. Wayman, and A. K. Jain, "FVC2000: fingerprint verification competition," *IEEE Trans. PAMI*, vol. 24, no. 3, pp. 402–412, 2002. 32
- [78] J. L. Horner and P. D. Gianino, "Phase-only matched filtering," *Appl. Opt.*, vol. 23, no. 6, pp. 812–816, Mar 1984. 37
- [79] Q. Chen, M. Defrise, and F. Deconinck, "Symmetric phase-only matched filtering of Fourier-Mellin transforms for image registration and recognition," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 16, no. 12, pp. 1156–1168, Dec. 1994. 38
- [80] A. V. Oppenheim, R. W. Schaffer, and J. R. Buck, *Discrete-time Signal Processing (2Nd Ed.)*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1999. 40
- [81] K. Ito and T. Aoki, "Phase-based image matching and its application to biometric recognition," *Proc. Asia-Pacific Signal and Information Processing Association Annual Summit and Conf.*, pp. 1–7, Oct. 2013. 44, 96
- [82] S. Nagashima, T. Aoki, T. Higuchi, and K. Kobayashi, "A subpixel image matching technique using phase-only correlation," *Proc. Int'l Symp. Intelligent Signal Processing and Communication Systems*, pp. 375–378, Dec. 2006. 44, 47
- [83] K. Miyazawa, K. Ito, T. Aoki, K. Kobayashi, and H. Nakajima, "An effective approach for iris recognition using phase-based image matching," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 30, no. 10, pp. 1741–1756, Oct. 2008. 49, 97
- [84] S. Iitsuka, K. Ito, and T. Aoki, "A practical palmprint recognition algorithm using phase information," *Proc. Int'l Conf. Pattern Recognition*, pp. 1–4, Dec. 2008. 49, 60, 88, 97, 103

- [85] D. Zhang, *BIOMETRIC SOLUTIONS For Authentication In An E-World*. Kluwer Academic Publication, 2002. 60
- [86] J. Ashbourn, *Biometrics in the New World: The Cloud, Mobile Technology and Pervasive Identity*. Springer, 2014. 60
- [87] X. Wu, D. Zhang, K. Wang, and B. Huang, “Palmprint classification using principal lines,” *Pattern Recognition*, vol. 37, no. 10, pp. 1987 – 1998, 2004. 60
- [88] K. Ito, T. Aoki, H. Nakajima, K. Kobayashi, and T. Higuchi, “A palmprint recognition algorithm using phase-based image matching,” *Proc. Int’l Conf. Image Processing*, pp. 2669–2672, Oct. 2006. 60
- [89] J. Mairal, F. Bach, and J. Ponce, *Sparse Modeling for Image and Vision Processing*. Now Publishers, 2014. 61, 65, 97
- [90] “PolyU palmprint database,” <http://www4.comp.polyu.edu.hk/~biometrics/>. 61, 84
- [91] “CASIA palmprint database,” <http://www.cbsr.ia.ac.cn/english/PalmprintDatabases.asp>. 61, 84, 124
- [92] I. Rish and G. Grabarnik, *Sparse Modeling: Theory, Algorithms, and Applications*, 1st ed. Boca Raton, FL, USA: CRC Press, Inc., 2014. 65
- [93] M. Savvides, B. Kumar, and P. Khosla, ““Corefaces” - robust shift invariant PCA based correlation filter for illumination tolerant face recognition,” *Proc. IEEE Computer Society Conf. Computer Vision and Pattern Recognition*, pp. 834–841, Jul. 2004. 68
- [94] A. Rodriguez, V. Boddeti, B. Kumar, and A. Mahalanobis, “Maximum margin correlation filter: A new approach for localization and classification,” *IEEE Trans. Image Processing*, vol. 22, no. 2, pp. 631–643, Feb. 2013. 68
- [95] H. Bristow, A. Eriksson, and S. Lucey, “Fast convolutional sparse coding,” *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 391–398, Jun. 2013. 69, 72, 74
- [96] B. Kong and C. Fowlkes, “Fast convolutional sparse coding (FCSC),” *Tech. Rep, UC Irvine*, 2014. 69, 72, 74
- [97] B. Wohlberg, “Efficient convolutional sparse coding,” *Proc. IEEE Conf. Acoustics, Speech and Signal Processing*, pp. 7173–7177, May 2014. 69, 72, 74

-
- [98] H. Bristow and S. Lucey, “Optimization methods for convolutional sparse coding,” *CoRR*, vol. abs/1406.2407, Jun. 2014. 69
- [99] F. Heide, W. Heidrich, and G. Wetzstein, “Fast and flexible convolutional sparse coding,” *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 5135–5143, Jun. 2015. 69, 72, 74
- [100] S. Mallat and Z. Zhang, “Matching pursuits with time-frequency dictionaries,” *IEEE Trans. Signal Processing*, vol. 41, no. 12, pp. 3397–3415, Dec. 1993. 69
- [101] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, “Distributed optimization and statistical learning via the alternating direction method of multipliers,” *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, 2010. 69
- [102] A. Beck and M. Teboulle, “A fast iterative shrinkage-thresholding algorithm for linear inverse problems,” *SIAM J. Imaging Sci.*, vol. 2, no. 1, pp. 183–202, 2009. 69
- [103] J. Tropp, “Greed is good: algorithmic results for sparse approximation,” *IEEE Trans. Inf. Theor.*, vol. 50, no. 10, pp. 2231–2242, Oct. 2004. 70
- [104] M. Elad, *Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing*. Springer, 2010. 70
- [105] D.-S. Huang, W. Jia, and D. Zhang, “Palmprint verification based on principal lines,” *Pattern Recognition*, vol. 41, no. 4, pp. 1316 – 1328, 2008. 85
- [106] M. Tistarelli and S. Z. Li, “Chapter 1 - real world applications: A literature survey,” in *Face recognition in adverse conditions*, M. De Marsico, M. Nappi, and M. Tistarelli, Eds. IGI Global, 2014, pp. 1–15. 93
- [107] K. Ricanek, M. Savvides, D. Woodard, and G. Dozier, “Unconstrained biometric identification: Emerging technologies,” *Computer*, vol. 2, no. 43, pp. 56–62, Feb. 2010. 93
- [108] S. Bharadwaj, H. Bhatt, M. Vatsa, and R. Singh, “Periocular biometrics: When iris recognition fails,” *Proc. IEEE Int’l Conf. Biometrics: Theory, Applications and Systems*, pp. 1–6, Sep. 2010. 94
- [109] P. Miller, J. Lyle, S. Pundlik, and D. Woodard, “Performance evaluation of local appearance based periocular recognition,” *Proc. IEEE Int’l Conf. Biometrics: Theory, Applications and Systems*, pp. 1–6, Sep. 2010. 94

- [110] P. Miller, A. Rawls, S. Pundlik, and D. Woodard, "Personal identification using periocular skin texture," *Proc. ACM Symp. Applied Computing*, pp. 1496–1500, Mar. 2010. 95
- [111] P. J. Phillips, P. J. Flynn, T. Scruggs, K. W. Bowyer, J. Chang, K. Hoffman, J. Marques, J. Min, and W. Worek, "Overview of the face recognition grand challenge," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 1, June 2005, pp. 947–954 vol. 1. 95
- [112] P. J. Phillips, H. Moon, S. A. Rizvi, and P. J. Rauss, "The feret evaluation methodology for face-recognition algorithms," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 22, no. 10, pp. 1090–1104, Oct 2000. 95, 123
- [113] D. Woodard, S. Pundlik, J. Lyle, and P. Miller, "Periocular region appearance cues for biometric identification," *Proc. IEEE Conf. Computer Vision and Pattern Recognition Workshops*, pp. 162–169, Jun. 2010. 95
- [114] P. J. Phillips, P. J. Flynn, J. R. Beveridge, W. T. Scruggs, A. J. O'Toole, D. Bolme, K. W. Bowyer, B. A. Draper, G. H. Givens, Y. M. Lui, H. Sahibzada, J. A. Scallan, and S. Weimer, "Overview of the multiple biometrics grand challenge," in *Advances in Biometrics*, M. Tistarelli and M. S. Nixon, Eds. Berlin, Heidelberg: Springer, 2009, pp. 705–714. 95
- [115] U. Park, R. Jillela, A. Ross, and A. Jain, "Periocular biometrics in the visible spectrum," *IEEE Trans. Information Forensics and Security*, vol. 6, no. 1, pp. 96–106, Mar. 2011. 95
- [116] R. R. Jillela, A. A. Ross, V. N. Boddeti, B. V. K. V. Kumar, X. Hu, R. J. Plemmons, and P. Pauca, *Handbook of Iris Recognition*. Springer, 2013, ch. Iris Segmentation for Challenging Periocular Images. 95, 104, 106
- [117] M. Uzair, A. Mahmood, A. Mian, and C. McDonald, "Periocular region-based person identification in the visible, infrared and hyperspectral imagery," *Neurocomputing*, vol. 149, pp. 854–867, Feb. 2015. 95
- [118] S. Behera, M. Gour, V. Kanhangad, and N. Puhan, "Periocular recognition in cross-spectral scenario," *Proc. Int'l Joint Conf. Biometrics*, pp. 681–687, Oct. 2017. 95
- [119] A. Sequeira, L. Chen, P. Wild, J. Ferryman, F. Alonso-Fernandez, K. B. Raja, R. Raghavendra, C. Busch, and J. Bigun, "Cross-eyed - cross-spectral iris periocular recognition database and competition," in *2016 International Conference of the Biometrics Special Interest Group (BIOSIG)*, Sept 2016, pp. 1–5. 95

-
- [120] Z. Zhao and A. Kumar, “Accurate periocular recognition under less constrained environment using semantics-assisted convolutional neural network,” *IEEE Trans. Information Forensics and Security*, vol. 12, no. 5, pp. 1017–1030, May 2017. 95, 96, 104, 111, 115
- [121] H. Proença and J. Neves, “Deep-PRWIS: Periocular recognition without the iris and sclera using deep learning frameworks,” *IEEE Trans. Information Forensics and Security*, vol. 13, no. 4, pp. 888–896, Apr. 2018. 95
- [122] A. Alahmadi, M. Hussain, H. Aboalsamh, and M. Zuair, “ConvSRC: Smartphone based periocular recognition using deep convolutional neural network and sparsity augmented collaborative representation,” *CoRR*, vol. abs/1801.05449, pp. 1–26, Jan. 2018. 95
- [123] J. Smereka and V. Kumar, “What is a “good” periocular region for recognition?” *Proc. IEEE Conf. Computer Vision and Pattern Recognition Workshops*, pp. 117–124, Jun. 2013. 95, 111
- [124] J. Smereka, B. Kumar, and A. Rodriguez, “Selecting discriminative regions for periocular verification,” *Proc. IEEE Int’l Conf. Identity, Security and Behavior Analysis*, pp. 1–8, Feb. 2016. 95, 111
- [125] S. Aoyama, K. Ito, and T. Aoki, “A finger-knuckle-print recognition algorithm using phase-based local block matching,” *Information Sciences*, vol. 268, pp. 53–64, Jun. 2014. 97
- [126] S. Lyu, “Divisive normalization as an efficient coding transform: Justification and effectiveness,” *Neural Computation*, vol. 23, no. 11, pp. 2942–2973, Nov. 2011. 98
- [127] J. Malo, I. Epifanio, R. Navarro, and E. Simoncelli, “Nonlinear image representation for efficient perceptual coding,” *IEEE Trans. Image Processing*, vol. 15, no. 1, pp. 68–80, Jan. 2006. 98
- [128] S. Lyu and E. P. Simoncelli, “Nonlinear image representation using divisive normalization,” *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8, 2008. 98
- [129] S. Crihalmeanu, A. Ross, and R. Derakhshani, “Enhancement and registration schemes for matching conjunctival vasculature,” *Lecture Notes in Computer Science (Proc. Int’l Conf. Biometrics)*, vol. 5558, pp. 1240–1249, Jun. 2009. 100
- [130] “CASIA Iris Image Database V4.0,” <http://www.idealtest.org/dbDetailForUser.do?id=4>. 104, 106

- [131] M. Pietikinen, A. Hadid, G. Zhao, and T. Ahonen, *Computer Vision Using Local Binary Patterns*, 1st ed. Springer, 2011. 106
- [132] “Semantic-assisted convolutional neural network,” <http://www4.comp.polyu.edu.hk/~csajaykr/scnn.rar>. 114
- [133] F. Juefei-Xu and M. Savvides, “Subspace-based discrete transform encoded local binary patterns representations for robust periocular matching on NIST’s face recognition grand challenge,” *IEEE Trans. Image Processing*, vol. 23, no. 8, pp. 3490–3505, Aug. 2014. 111
- [134] Z. Zivkovic and F. van der Heijden, “Recursive unsupervised learning of finite mixture models,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 5, pp. 651–656, May 2004. 122
- [135] A. Jain, K. Nandakumar, and A. Ross, “Score normalization in multimodal biometric systems,” *Pattern Recogn.*, vol. 38, no. 12, pp. 2270–2285, Dec. 2005. 122, 125
- [136] P. J. Phillips, K. W. Bowyer, P. J. Flynn, X. Liu, and W. T. Scruggs, “The iris challenge evaluation 2005,” in *2008 IEEE Second International Conference on Biometrics: Theory, Applications and Systems*, Sept 2008, pp. 1–8. 123
- [137] “PolyU FKP database,” <http://www4.comp.polyu.edu.hk/~biometrics/FKP.htm>. 124
- [138] T. Ahonen, E. Rahtu, V. Ojansivu, and J. Heikkila, “Recognition of blurred faces using local phase quantization,” *2008 19th International Conference on Pattern Recognition*, pp. 1–4, Dec 2008. 124
- [139] L. Zhang, L. Zhang, D. Zhang, and H. Zhu, “Ensemble of local and global information for finger–knuckle-print recognition,” *Pattern Recognition*, vol. 44, no. 9, pp. 1990 – 1998, 2011, computer Analysis of Images and Patterns. 124

位相に基づく画像マッチングを用いた高性能バイオメトリクス認証

High-Performance Biometric Recognition
Using Phase-Based Image Matching

by

Luis Rafael Marval Pérez

Graduate School of Information Sciences, Tohoku University

July 2019

Copyright © 2019 Luis Rafael Marval Pérez,
All Rights Reserved.