Fingerprint Template Security Strategies

A thesis submitted during 2016 to the University of Hyderabad in partial fulfillment of the award of a Ph.D. degree

by

MULAGALA SANDHYA

**School of Computer and Information Sciences
University of Hyderabad
P.O. Central University, Gachibowli
Hyderabad – 500046
Telangana, India**

CERTIFICATE

This is to certify that the thesis entitled **“Fingerprint Template Security Strategies”** submitted by **Mulagala Sandhya** bearing **Reg. No. 13MCPC19** in partial fulfillment of the requirements for the award of **Doctor of Philosophy** in **Computer Science** is a bonafide work carried out by her under my supervision and guidance.

The thesis has not been submitted previously in part or in full to this or any other University or Institution for the award of any degree or diploma.

Dr. M. V. N. K Prasad,

Associate Professor,
Institute for Development &
Research in Banking Technology,
Road No.1, Castle Hills,
Masab Tank, Hyderabad- 500057

Director,

Institute for Development &
Research in Banking Technology,
Hyderabad- 500057

Dean,

School of Computer &
Information Sciences,
University of Hyderabad,
Hyderabad - 500046

DECLARATION

I, **Mulagala Sandhya**, hereby declare that this thesis entitled **"Fingerprint Template Security Strategies"** submitted by me under the guidance and supervision of **Dr. M. V. N. K Prasad** is a bonafide research work which is also free from plagiarism. I also declare that it has not been submitted previously in part or in full to this University or any other University or Institution for the award of any degree or diploma. I hereby agree that my thesis can be deposited in Shodganga / INFLIBNET.

A report on plagiarism statistics from the University Librarian is enclosed.

Name: MULAGALA SANDHYA

Date:

Signature of the student
Reg. No: 13MCPC19

//Countersigned//

Signature of the Supervisor:

Dedicated To My Parents

Acknowledgements

"He who learns but does not think is lost! He who thinks but does not learn is in great danger" - Confucius
"Tell me and I forget, Teach me and I remember, Involve me and I learn" - Benjamin Franklin
"Learning never exhausts the mind" - Leonardo da Vinci

Every day during my Ph.D. studies has been a great opportunity for learning. Though only my name appears on the cover of this thesis, a great many people have contributed to its production. It is a pleasant aspect that I have now the opportunity to express my gratitude for all of them.

First and the foremost, I would like to thank **God**, for blessing me and giving me the strength to keep going.

I owe my gratitude to my advisor **Dr. M. V. N. K Prasad**. I have been amazingly fortunate to have such an advisor who gave me the freedom to explore on my own, and at the same time the guidance to recover when my steps faltered. In addition to his support and flexibility, I want to thank him for fine- tuning my research topic and keeping me focused.

I would like to acknowledge my doctoral review committee members **Prof. B. M. Mehtre** and **Prof. V. Ravi** for their constructive criticisms at different stages of my research. I also thank the **anonymous reviewers** and **handling editors** of my publications for their insightful comments which helped me to improve the manuscripts.

I am also thankful to **Institute for Development and Research in Banking Technology (IDRBT)**, Hyderabad that has provided a stimulating environment and facilities without which the completion of this work could not be imagined. I gratefully acknowledge IDRBT Director **Dr. A.S. Ramasastri** for the encouragement, funding, and support during my course of study. Moreover, thanks to the **faculty** members at **IDRBT** for their encouragement and support whenever needed.

I sincerely thank **Dean** and **Faculty** of **School of Computer and Information Sciences, University of Hyderabad** for the administrative assistance and support for discussions whenever needed. I want to extend my appreciation to **Prof C. Raghavendra Rao**, for his immense help in identifying ways to achieve my research goals.

My special regards to the **teachers** at different stages of my education. Because of their kindness I feel, was able to reach a stage where I could write this thesis. A special mention of thanks to my **friends** at different stages of my career for their timely help and friendship, which should be always remembered.

On the personal level, I want to thank my **colleagues** at IDRBT for keeping me motivated throughout the duration of the work and sharing the joys and frustrations of research.

I can never forget to be highly indebted to my **parents** for their emotional support, no matter what path I choose. Their patience and sacrifice will remain my inspiration throughout my life. Most importantly, I wish to thank my father **Late Gopala Krishna**, who, if there feel proud for my successful Doctoral study. His memory will be with me always. I want to thank my mother **Jhansi Rani**, who has been my inspiration for hard work. I would never be able to pay back the love and affection showered upon by them. A special mention of thanks to my younger brother **Dileep** for his love and affection.

As always it is impossible to mention everybody who had an impact on this work, however, there are those whose spiritual support is even more important. I feel a deep sense of gratitude for my grandparents, **Sitaramiah** and **Venkamma**, **Surya Narayana** and **Narayanamma** who formed part of my vision and taught me good things that really matter in life.

I must thank my relatives **Varalakshmi, Ram Mohan Rao, Nagamani, Ambica, Sudheer, Venkataramiah, Kusumamba, Sarada**, and many more, whose well wishes made me see this day.

Finally, I would like to acknowledge the people who mean the world to me, my husband and children. My dearest husband **Narayana Murthy** made everything possible in so many ways. His infallible love and encouragement have always been my strength. My children **Durga Sai Sri** and **Saranya Sai Priya** are the pride and joy of my life. They have grown up watching me study and juggle with family and work. I love you more than anything. I hope I have not lost too much during the tenure of my study.

... Thank you all

M SANDHYA

Abstract

With the emergence of biometric authentication systems, template protection for biometrics captured attention in the recent years. The privacy concern arises due to storage and misuse of biometric data in various applications. We systematically reviewed the published literature on Biometric Template Protection (BTP) during 2005-2016 and presented the current status of BTP schemes by a methodical analysis and taxonomy of BTP approaches, modalities, the fusion of modalities (multi-modal), and hybrid methods. We presented the research implications, and extraction outcomes of Systematic Literature Review (SLR) conducted on BTP schemes. This SLR helps researchers and practitioners to find relevant information on BTP methods thereby reducing time and complexity in searching the appropriate studies.

The BTP approaches are classified as Cancelable Biometrics, Bio-Cryptosystems, hybrid methods, and Homomorphic encryption based methods. Earlier works in designing cancelable fingerprint templates used a threshold on number of minutiae points considered for cancelable template design. Hence, we focused on designing cancelable templates for fingerprints by using each and every minutia point in the fingerprint image. For this, we developed two methods for generating cancelable fingerprint templates using k-Nearest Neighborhood Structure (kNNS) and Delaunay Triangle Feature Set (DTFS) construction. Experiments and analysis conducted on publicly available benchmark databases prove the tenability of the proposed methods.

Biometric fusion is the use of multiple inputs or methods of processing of biometric samples. It has many advantages such as improved accuracy, efficiency, applicability and robustness. We proposed two fusion methods for generating cancelable fingerprint templates. First, we compute two structures namely, Local Structure (LS) and Distant Structure (DS) and fuse them at the feature level. Second, we fuse the match scores obtained from two algorithms (kNNS and DTFS) at score level using weighted sum rule and T-operators. Experiments and analysis conducted on publicly available benchmark databases prove the credibility of the proposed fusion methods.

Combining traditional cryptography and biometrics leads to the design of biometric cryptosystems. We developed a fingerprint cryptosystem using Delaunay Neighbor Structures (DNS). We also developed hybrid methods combining cancelable biometrics and bio-cryptosystems using fuzzy commitment scheme. Cancelable fingerprint cryptosystems are designed using Delaunay Neighbor Structures, multiple spiral curves, by employing error correcting codes. Experiments and analysis done on publicly available benchmark databases prove the efficacy of the proposed methods.

Contents

Acknowledgments **v**

Abstract **viii**

List of Figures **xv**

List of Tables **xx**

List of Algorithms **xxiii**

Glossary **xxiv**

1 Introduction **1**

1.1 Biometric Recognition System 2

1.2 Attacks on Biometric Systems 3

1.2.1 Attacks on User Interfaces 3

1.2.2 Attacks on Modules 4

1.2.3 Attacks on Channels between modules 4

1.2.4 Attacks on Template Databases 4

1.3 What is Biometric Template Protection? 5

1.4 Architecture overview 5

1.5 Desirable properties of Biometric Template Protection techniques . . 6

1.6 Evolution of Biometric Template Protection Schemes 7

1.7 Categories of Biometric Template Protection schemes 8

1.8 Cancelable Biometrics 8

1.8.1 Salting 8

1.8.2 Non-invertible transforms 10

1.9 Biometric Cryptosystems 10

1.9.1 Key binding Biometric cryptosystems 11

1.9.2 Key generating Biometric cryptosystems 13

1.10 Hybrid methods 14

1.11 Homomorphic Encryption 14

1.12 Information Fusion in Biometrics 15

1.13 Motivation for present work, aim, and objectives 16

1.14 Contributions of the thesis 17

1.15 Thesis organization 18

2 A Systematic Literature Review on Biometric Template Protection 20

2.1 Sources of Information for Collecting Studies for this SLR 22

2.2 Search Criteria 23

2.3 Initial and Final Selection of studies 24

2.4 Selected Studies Under Cancelable Biometrics 25

2.4.1 Salting methods 25

2.4.2 Non-invertible Transforms 27

2.5 Selected Studies Under Biometric Cryptosystems 33

2.5.1 Key-binding Bio-cryptosystems 33

2.5.2 Key generating Biometric cryptosystems 38

2.6 Selected Studies Under Hybrid Methods 39

2.7 Selected Studies Under Homomorphic Encryption 41

2.8 Analysis and Discussion of SLR Results 42

2.9 Justification for present work 50

3 Cancelable Fingerprint Template Generation Techniques 51

3.1 Fingerprint Features 52

3.2 Fingerprint image preprocessing and feature extraction 53

3.3 kNNS algorithm 53

3.3.1 Construction of kNNS for each minutia point 54

3.3.2 Quantize each $kNNS_i$ and map to 2D array 56

3.3.3 Generation of bit string for each $kNNS_i$ 57

3.3.4 Generation of cancelable template 57
3.3.5 Matching 58
3.4 DTFS_INCIR and DTFS_AVGLO Algorithms 61
3.4.1 Constructing Delaunay triangulation net from fingerprint minutiae 61
3.4.2 Computation of feature set from Delaunay triangles 64
3.4.3 Quantize DTFS, and map to a 3D array 66
3.4.4 Generation of bit string 67
3.4.5 Generation of cancelable template 67
3.4.6 Matching 68
3.5 Performance measures of a biometric system 71
3.6 Benchmark Databases 73
3.7 Experimental Results and Discussion 73
3.7.1 Experiment Setup 73
3.7.2 Accuracy 74
3.7.3 Revocability 76
3.7.4 Security Analysis 77
3.8 Performance Comparison with existing methods 83
3.9 Summary 84

4 Fusion Techniques for Protecting Fingerprint Templates 86
4.1 Feature Level Fusion Method 87
4.1.1 Computation of transformed features for minutiae points . . . 87
4.1.2 Representing LS and DS as bit strings 90
4.1.3 Fusion of bit strings B_{ls} and B_{ds} 91
4.1.4 Generation of cancelable template 91
4.1.5 Matching 92
4.2 Score Level Fusion Method 96
4.2.1 Fusion of match scores of kNNS and DTFS_AVGLO algorithms using weighted sum rule 97
4.2.2 Fusion of match scores of kNNS and DTFS_AVGLO Algorithm using T-operators 97
4.3 Experimental Results and Discussion 99

4.3.1 Experimental Setup 99
4.3.2 Accuracy 100
4.3.3 Revocability 106
4.3.4 Security analysis of the FEATURELEVEL_LSDS method . . 108
4.3.5 Security Analysis of score level fusion methods 110
4.4 Performance Comparison with existing methods 113
4.5 Summary 114

5 Bio-Cryptosystems and Hybrid Techniques for Securing Fingerprint Templates 116
5.1 Fingerprint Cryptosystem using Delaunay Neighbor Structures 117
5.1.1 Computation of DNS 117
5.1.2 Bit string generation from DNS 120
5.1.3 Apply Dimensionality reduction matrix to each bit string . . . 120
5.1.4 Employing Fuzzy Commitment Scheme 121
5.1.5 Matching 122
5.1.6 Experimental Results and Discussion 124
5.2 Hybrid technique using Delaunay Neighbor Structures 127
5.2.1 Constructing DNS for fingerprint minutiae 128
5.2.2 Transform DNS and generate bit string 128
5.2.3 Apply Dimensionality reduction matrix to the bit string 130
5.2.4 Employing fuzzy commitment scheme 130
5.2.5 Matching 131
5.2.6 Experimental Results and Discussion 133
5.3 Hybrid technique using Multiple Spiral Curves 138
5.3.1 Construction of multiple spiral curves 138
5.3.2 Bit string representations of multiple spiral curves 140
5.3.3 Apply Dimensionality reduction matrix to the bit string 141
5.3.4 Employing fuzzy commitment scheme 141
5.3.5 Matching 141
5.3.6 Experimental Results and Discussion 141
5.4 Hybrid technique using Convolution Coding 148

5.4.1 Compute transformed features using Delaunay triangulation net and generate bit string 148
5.4.2 Employ fuzzy commitment scheme by using convolution coding 148
5.4.3 Matching 149
5.4.4 Experimental Results and Discussion 152
5.5 Performance Comparison with existing methods 155
5.6 Summary 158

6 Conclusions and Future Directions 159
6.1 Conclusions 159
6.2 Future research directions 161

Author's Publications 162

References 164

List of Figures

1.1 (a)Traditional authentication systems (based on “what he knows and what he possess?)” (b)Biometric based authentication systems based on “who he is intrinsically?” 2

1.2 Block Diagram of Biometric Recognition System 3

1.3 Points of Attack on Biometric Systems adapted from [1] 3

1.4 Architecture of Biometric Template Protection methods 6

1.5 Evolution of Biometric Template Protection Schemes 7

1.6 A hierarchical taxonomy of Biometric Template Protection schemes . 9

1.7 Framework of Cancelable Biometric System 10

1.8 Framework of a Key Binding Biometric Cryptosystem 11

1.9 Framework of Fuzzy Commitment Scheme 12

1.10 Framework of Fuzzy Vault Scheme 13

1.11 Framework of Key Generating Biometric Cryptosystem 13

1.12 Framework of Quantization Scheme 14

1.13 Multiple sources of evidence 15

1.14 Levels of fusion in biometrics 16

2.1 Review Methodology based on [2, 3] 21

2.2 Stages in selection of studies 25

2.3 Overall percentage of Biometric Template Protection categories in the literature 43

2.4 Overall percentage of Biometric Template Protection schemes developed on each biometric trait 46

2.5 Percentage of Salting methods developed on each biometric trait . . . 46

2.6 Percentage of Non-invertible transforms developed on each biometric trait . . . 47

2.7 Percentage of Fuzzy commitment schemes developed on each biometric trait . . . 47

2.8 Percentage of Fuzzy Vault schemes developed on each biometric trait 48

2.9 Percentage of Key generating systems developed on each biometric trait 48

2.10 Percentage of Hybrid methods developed on each biometric trait . . . 49

2.11 Percentage of Homomorphic Encryption based methods developed on each biometric trait . . . 49

3.1 Fingerprint image and its features . . . 52

3.2 Flow diagram of kNNS algorithm. . . . 54

3.3 kNNS of reference minutia *r* with neighborhood (k)= 5 . . . 55

3.4 Array A_1 with cell size C_x, C_y to which *kNNS* is mapped . . . 56

3.5 Local matching between enrolled and query fingerprint templates in kNNS algorithm . . . 59

3.6 Flow diagram for DTFS_INCIR and DTFS_AVGLO algorithms . . . 62

3.7 "Delaunay triangulation net" constructed from minutiae points of a fingerprint image. . . . 63

3.8 A triangle ΔT considered for computation of feature set in DTFS_INCIR 64

3.9 A triangle ΔT considered for computation of feature set in DTFS_AVGLO . . . 66

3.10 A_1 with cell sizes C_x, C_y, C_z where DTFS is mapped . . . 67

3.11 ROC curves for kNNS algorithm in same key scenario for FVC 2002 databases . . . 76

3.12 FAR/FRR obtained for kNNS algorithm in same key scenario for FVC 2002 (a)DB1 (b)DB2 (c)DB3 . . . 77

3.13 Genuine and imposter score distributions of kNNS algorithm in same key scenario for FVC 2002 (a)DB1, (b)DB2, (c)DB3 . . . 78

3.14 ROC curves in same key scenario for DTFS_INCIR algorithm . . . 79

3.15 ROC curves in same key scenario for DTFS_AVGLO algorithm . . . 79

3.16 FAR/FRR obtained in same key scenario for database FVC 2002 DB2 for algorithm (a)DTFS_INCIR (b)DTFS_AVGLO . . . 80

3.17 Genuine and imposter score distribution for FVC 2002 DB1 using algorithm (a)DTFS_INCIR (b)DTFS_AVGLO 81

3.18 Genuine, imposter and pseudo-imposter distribution of kNNS algorithm for database “FVC 2002 DB2” in “same key scenario” 82

3.19 d' value comparison of kNNS, DTFS_INCIR, DTFS_AVGLO algorithms with existing methods 84

4.1 Overall framework of proposed feature level fusion method 88

4.2 Representation of LS_r and DS_r for reference minutia point r with $p = 3$ and $q = 2$ 89

4.3 Array A_1 with cells of size c_x, c_y on which LS_r is projected. 90

4.4 Array A_2 with cells of size c_w, c_h on which DS_r is projected. 90

4.5 Generating fused bit string(B_{fus}) 92

4.6 Matching between enrolled and query templates 93

4.7 Schematic diagram of proposed score level fusion method. 97

4.8 ROC curves of FEATURELEVEL_LSDS fusion method on FVC 2002 databases 102

4.9 FAR/FRR of FEATURELEVEL_LSDS fusion method for FVC 2002 (a)DB1 (b)DB2 (c)DB3 103

4.10 Score distributions of FEATURELEVEL_LSDS fusion method for FVc 2002 (a)DB1 (b)DB2 (c)DB3 104

4.11 ROC curves for SCORELEVEL_WS method with weights $w_1 = 0.5$ and $w_2 = 0.5$ 106

4.12 FAR/FRR obtained for SCORELEVEL_WS method with weights $w_1 = 0.5$ and $w_2 = 0.5$ 108

4.13 ROC curves for SCORELEVEL_TOP method using Sugeno weber T-operators for FVC 2002 DB1 109

4.14 FAR/FRR obtained for SCORELEVEL_TOP method using(a)Sugeno-Weber T-norm with p=0.4 (b)Sugeno-Weber T-conorm with p= -0.4 for FVC 2002 DB1 110

4.15 Fused score distribution of SCORELEVEL_WS method with weights w_1=0.5 and w_2=0.5 for FVC 2002 (a)DB1 (b)DB2 (c)DB3 111

4.16 Fused score distribution of SCORELEVEL_TOP method for FVC 2002 DB1 using (a)Sugeno-Weber T-norm with p=0.4 (b)Sugeno-Weber T-conorm with p= -0.4 112

4.17 Pseudo imposter distribution of FEATURELEVEl_LSDS method for FVC 2002 DB2 112

4.18 d' value comparison of FEATURELEVEL_LSDS, SCORELEVEL_WS, SCORELEVEL_TOP methods with existing methods 114

5.1 Flow diagram for the BIOCRYP_DNS method 118

5.2 Delaunay triangulation formed from fingerprint minutiae and DNS for minutia point r 119

5.3 (a)DNS of minutia point r (b) Mapping DNS to a 3D array 119

5.4 Computing local score between encrypted templates of trained and testing images 122

5.5 ROC curves for BIOCRYP_DNS method for FVC 2002 databases . . 125

5.6 FAR/FRR of BIOCRYP_DNS method for FVC 2002 (a)DB1 (b)DB2 (c)DB3 databases 126

5.7 Genuine and imposter score distribution of BIOCRYP_DNS method for FVC 2002 (a)DB1 (b)DB2 (c)DB3 databases 127

5.8 Overall framework of the HYBRID_DNS method 129

5.9 2D array on which TFV_i is mapped 130

5.10 ROC curves of HYBRID_DNS method for FVC 2002 databases . . . 134

5.11 FAR/FRR of HYBRID_DNS method for FVC 2002 (a)DB1 (b)DB2 (c)DB3 135

5.12 Genuine and Imposter score distribution of HYBRID_DNS method for FVC 2002 (a)DB1 (b)DB2 (c)DB3 136

5.13 d' values for the HYBRID_DNS method on FVC 2002 and 2004 databases 137

5.14 KS test values for HYBRID_DNS method on FVC 2002 and 2004 databases 137

5.15 Flow diagram of HYBRID_MSC method 139

5.16 (a)Spiral curve constructed for a reference minutia point(b)The metrics considered for a triangle to build the feature set L_r 140
5.17 ROC curves of HYBRID_MSC method on FVC 2002 databases . . . 144
5.18 FAR/FRR of HYBRID_MSC method for FVC 2002 databases (a) DB1 (b) DB2 (c) DB3 145
5.19 Score distribution for HYBRID_MSC method on FVC 2002 databases (a) DB1 (b) DB2 (c) DB3 146
5.20 d' values of HYBRID_MSC method for databases of FVC 2002 . . . 146
5.21 KS test values of HYBRID_MSC for databases of FVC 2002 147
5.22 Genuine, Imposter, and Pseudo-imposter score distribution of HYBRID_MSC method on FVC 2002 DB2 147
5.23 Overall flow diagram for HYBRID_CC method 149
5.24 A (3,1,3)convolution encoder 150
5.25 ROC curves of HYBRID_CC method for FVC 2002 databases 153
5.26 FAR/FRR of HYBRID_CC method for FVC 2002 (a)DB1 (b)DB2 (c)DB3 databases 154
5.27 Genuine and Imposter score distribution of HYBRID_CC method for FVC 2002 (a)DB1 (b)DB2 (c)DB3 databases 155
5.28 Genuine and Imposter and Psedo-imposter score distribution of HYBRID_CC method for FVC 2002 DB2 156
5.29 Comparison of d' values for BIOCRYP_DNS, HYBRID_DNS, HYBRID_MSC, HYBRID_CC methods with existing methods 157

List of Tables

2.1 Research Questions and Motivation 22

2.2 Inclusion and Exclusion Criteria 23

2.3 Search Strings used for SLR 24

2.4 Summary of selected studies under Salting methods in chronological order (EER, FAR, FRR values are shown in %) 27

2.5 Summary of selected studies under Non-inveritble transforms in chronological order (EER, FAR, FRR values are shown in %) 30

2.6 Summary of selected studies under Non-inveritble transforms in chronological order (EER, FAR, FRR values are shown in %) 31

2.7 Summary of selected studies under Key binding cryptosystems using fuzzy commitment scheme in chronological order (EER, FAR, FRR values are shown in %) 34

2.8 Summary of selected studies under key binding cryptosystems using fuzzy vault scheme in chronological order (EER, FAR, FRR values are shown in %) 37

2.9 Summary of selected studies under Key-generating cryptosystems in chronological order (EER, FAR, FRR values are shown in %) 39

2.10 Summary of selected studies under Hybrid methods in chronological order (EER, FAR, FRR values are shown in %) 40

2.11 Summary of selected studies under Homomorphic Encryption in chronological order 42

2.12 Active Research Communities in designing Biometric Template Protection methods 44

2.13 Publicly available databases on which BTP schemes are evaluated in the literature and their source 45

3.1 EER obtained for kNNS algorithm for FVC 2002, 2004 databases in the same key scenario 74

3.2 EER obtained for DTFS_INCIR algorithm for FVC 2002, 2004 databases in the same key scenario 75

3.3 EER obtained for DTFS_AVGLO algorithm for FVC 2002, 2004 databases in same key scenario. 75

3.4 d' values obtained for kNNS, DTFS_INCIR, DTFS_AVGLO algorithms in same key and different key scenarios 82

3.5 KS test values obtained for kNNS, DTFS_INCIR, DTFS_AVGLO algorithms in same key and different key scenarios 82

3.6 EER comparison of kNNS, DTFS_INCIR, DTFS_AVGLO algorithms with existing methods in “same key scenario” 83

4.1 T-norms implemented for fusion of match scores. 99

4.2 T-conorms implemented for fusion of match scores. 99

4.3 EER obtained for FVC 2002 and 2004 databases by FEATURELEVEL_LSDS fusion method in same key scenario 101

4.4 d' and KS test values for FVC 2002 databases using FEATURELEVEL_LSDS fusion method 102

4.5 d' and KS test values obtained for FVC 2004 databases using FEATURELEVEL_LSDS fusion method 102

4.6 EER obtained for SCORELEVEL_WS and SCORELEVEL_TOP methods for FVC 2002 databases 105

4.7 d' and KS test values of SCORELEVEL_WS method for FVC 2002 databases 106

4.8 d' and KS test values SCORELEVEL_TOP method using T-norms for databases in FVC 2002. 107

4.9 d' and KS test values SCORELEVEL_TOP method using T-conorms for databases in FVC 2002. 107

4.10 EER comparison of FEATURELEVEL_LSDS, SCORELEVEL_WS, SCORELEVEL_TOP methods in same key scenario with existing methods 113

5.1 EER resulted for BIOCRYP_DNS method for FVC 2002 databases . . 125

5.2 EER resulted for HYBRID_DNS method on FVC 2002 databases . . 133

5.3 EER resulted for HYBRID_DNS method for FVC 2004 databases . . 134

5.4 EER for HYBRID_MSC method on FVC 2002 databases 143

5.5 EER and d' values obtained for HYBRID_CC method on FVC 2002 databases 152

5.6 Comparison of EER for BIOCRYP_DNS, HYBRID_DNS, HYBRID_MSC, HYBRID_CC methods with existing methods 157

List of Algorithms

3.1 kNNS($x_i, y_i, \theta_i, k, A_1, C_X, C_Y, R$): To compute k-Nearest Neighborhood Structure . . . 59

3.2 DTFS_INCIR($x_i, y_i, \theta_i, A_1, C_X, C_Y, C_Z, R$): To compute Delaunay Triangle Feature Set using incircle center of each triangle . . . 69

3.3 DTFS_AVGLO($x_i, y_i, \theta_i, A_1, C_X, C_Y, C_Z, R$): To compute Delaunay Triangle Feature Set using average of edge lengths and orientation difference at each vertex . . . 70

4.1 FEATURELEVEL_LSDS ($x_i, y_i, \theta_i, p, q, A_1, A_2, c_x, c_y, c_w, c_h, L, R$): To compute Local Structure and Distant Structure and fuse them . . . 94

4.2 SCORELEVEL_WS(ms_1, ms_2): To compute fused scores from match scores returned from kNNS and DTFS_AVGLO algorithms using weighted-sum rule . . . 98

4.3 SCORELEVEL_TOP(ms_1, ms_2, p): To compute fused scores from match scores returned from kNNS and DTFS_AVGLO algorithms using T-operators . . . 100

5.1 BIOCRYP_DNS ($x_i, y_i, \theta_i, A_1, c_x, c_y, c_z, n, k, \lambda$): To compute Delaunay Neighbor Structures and generate secure template . . . 123

5.2 HYBRID_DNS ($x_i, y_i, \theta_i, A_1, c_l, c_h, n, k, \lambda$): To compute Transformed DNS and generate secure template . . . 131

5.3 HYBRID_MSC ($x_i, y_i, \theta_i, S_1, c_x, c_y, c_z, c_w, n, k, \lambda$): To compute Multiple Spiral Curves and generate secure template . . . 142

5.4 HYBRID_CC ($x_i, y_i, \theta_i, A_1, c_x, c_y, c_z, n, k, \lambda$): To devise a hybrid method using convolution coding and generate secure template . . . 150

Glossary

ABAC	Anonymous Biometric Access Control
AES	Advanced Encryption Standard
BCH	Bose, Chaudhuri, and Hocquenghem
BiPS	Binarized Phase Spectrum
CIRF	Correlation Invariant Random Filtering
DFT	Discrete Fourier Transform
DITOM	Densely Infinite To One Mapping
DNS	Delaunay Neighbor Structure
DS	Distant Structure
DTFS	Delaunay Triangle Feature Set
GF	Galois Field
GMS	Global Match Score
HE	Homomorphic Encryption
HM	Hamming Measure
HMM	Hidden Markov Model
ICA	Independent Component Analysis

kAQ	k-Anonymous Quantization
kNNS	k-Nearest Neighborhood Structure
KS	Kolmogorov-Smirnov test

LBP	Local Binary Pattern
LDP	Local Direction Pattern
LMS	Local Match Score
LS	Local Structure

MCC	Minutiae Cylinder Code
MLC	Multi-Line Code
MVD	Minutiae Vicinity Decomposition

NIR	Near Infrared
NXOR	Not eXclusive OR

PCA	Principal Component Analysis
PI	Pseudonymous Identifier
PIC	Pseudonymous Identity Comparator
PIE	Pseudonymous Identifier Encoder
PIN	Personal Identification Number
PIR	Pseudonymous Identifier Recorder

QoP	Quality of Performance

RBR	Renewable Biometric Record
RBT	Randomized Biometric Templates
RDQT	Randomized Dynamic Quantization Transformation

RGHE	Randomized Graph-based Hamming Embedding
ROC	Receiver Operating Characteristic
SDK	Software Development Kit
SHA	Secure Hash Algorithm
SOM	Self Organizing Map
T-norm	Triangular norm
TFV	Transformed Feature Vector
VNS	Voronoi Neighbor Structure
XOR	eXclusive OR

Chapter 1

Introduction

Identity management plays a critical role in a number of applications. Examples of such applications include regulating international border crossings, controlling access to shared resources and information, performing remote financial transactions [4]. A person can be recognized in three ways as shown in Figure 1.1: (i)what he knows (Personal Identification Number (PIN), password, or cryptographic key) (ii)what he possesses extrinsically (Identification card, passport, or driving license, etc) and (iii)who he is intrinsically (physical traits) or what he does (behavioral traits). The third way establishes the person's identity based on his inherent physical or behavioral traits and is known as biometric recognition [5]. Biometrics offers an enhanced level of security, where the person present "proofs" that directly connect with their own intrinsic physical or behavioral characteristics [6]. The most commonly used physical biometric traits are a fingerprint, face, iris, palmprint, hand geometry, hand vein, finger vein, voice, etc. Behavioral biometric characteristics include keystroke dynamics, gait etc [7, 8]. The advantage of biometric recognition over conventional authentication systems like knowledge based systems or token-based systems is that the user need not remember a password (or PIN) or carry a token [9]. Thus, the identity of the user is difficult to duplicate, forge, lost or forgotten to result in uniqueness, permanence, and non-repudiation of biometrics [10].

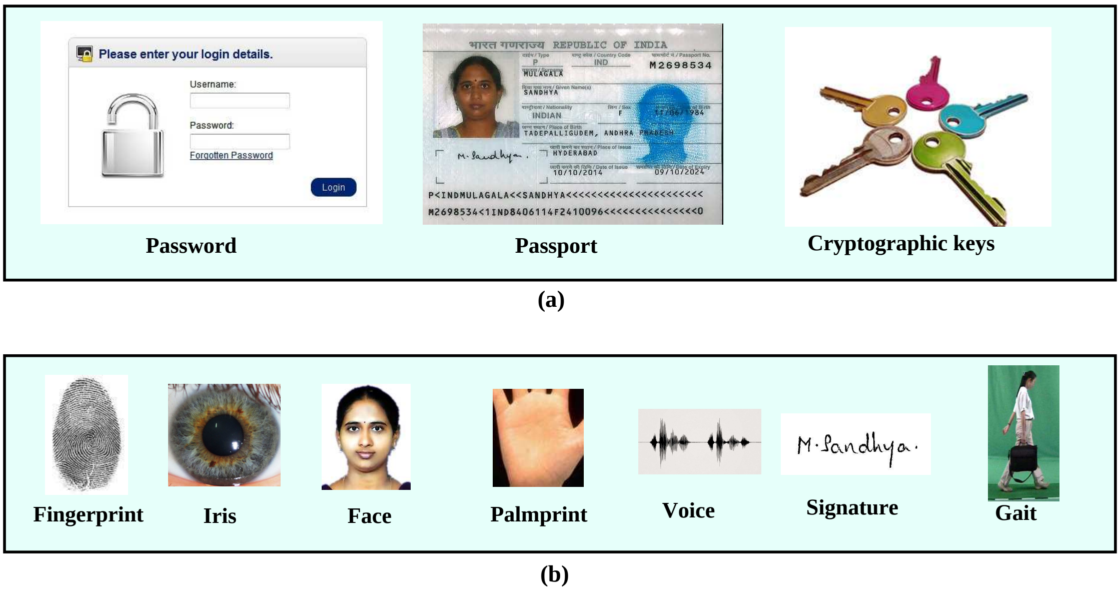


Figure 1.1: (a)Traditional authentication systems (based on "what he knows and what he possess?)" (b)Biometric based authentication systems based on "who he is intrinsically?"

1.1 Biometric Recognition System

The block diagram of a biometric recognition system is shown in Figure 1.2. The sensors are used to obtain the biometric characteristics [6]. While obtaining the data of a biometric, there may be the possibility of inclusion of noise or unwanted background information or foreign objects, etc. Hence preprocessing is needed to remove them. Filters are used to remove the noise. Segmentation is used to remove the background information and foreign objects [5].

The feature extractor module is an important step in the biometric recognition system. The type of features to be used may vary depending on the application [11]. The first time an individual uses a biometric system is called enrollment [8]. During enrollment stage, the feature extractor and template generator run simultaneously to extract features from biometric data and store them as templates in the database [10]. During Identification/Verification stage, the query biometric is processed in the same way. Now, the query template is compared with the enrolled template that yields in a match (accept) or non-match (reject) [6].

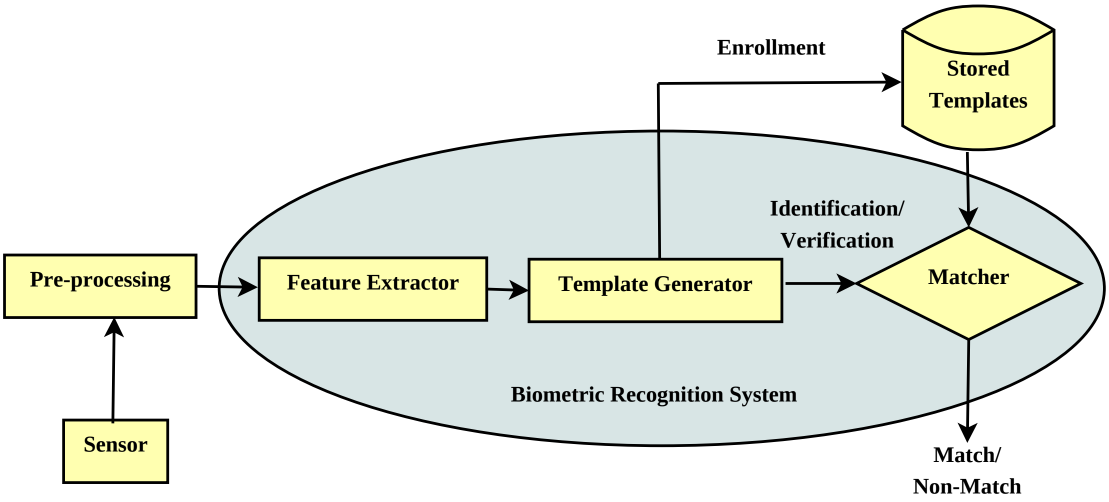


Figure 1.2: Block Diagram of Biometric Recognition System

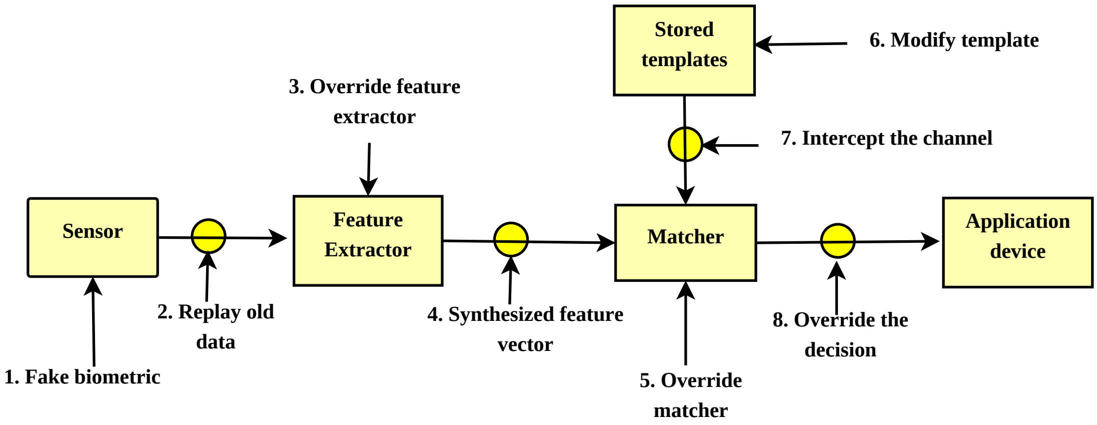


Figure 1.3: Points of Attack on Biometric Systems adapted from [1]

1.2 Attacks on Biometric Systems

Ratha et al. [1] found eight attack points of biometric systems as shown in Figure 1.3. The attacks may be on user interfaces, on modules, on channels between modules, and on template databases.

1.2.1 Attacks on User Interfaces

The attack at the user interface is due to the presentation of a fake biometric trait [12]. In this mode of attack, a possible reproduction of the biometric being used will be presented to the sensor [1]. Examples include a face mask, fake finger, or a copy of a signature. If the sensor does not differentiate the spoof and genuine biometric traits, the adversary enters into the system under a false identity [13].

1.2.2 Attacks on Modules

The attacks on the **sensor module** can be the coercive attack, spoofing attack or device substitution or denial of service attacks [14]. Coercive attack refers to presenting a true biometric to the sensor in an unauthorized manner, for *e.g.* the adversary forces the genuine user to grant him access to the particular system [15]. Spoofing attack refers to copying the biometric of the legitimate user and transfer it to adversary thereby fooling the system. A sensor device may be simulated and replaced with the genuine capture device [16]. The attacks on **feature extractor module** are forcing it to produce pre-selected features of the adversaries. **Matchers** are attacked to override the match scores and produce fake scores [17].

1.2.3 Attacks on Channels between modules

The adversary may intercept the channels between the modules like, the channel between sensor and feature extractor, feature extractor and matcher, template database and matcher, matcher and application device [4]. Possible attacks here may be replay attack, the synthesized feature vector, overriding the decision [18]. Replay attack allows an old recorded signal to be replayed into the system bypassing the sensor [1]. A synthesized feature vector may be produced after the features have been extracted from the input signal. If the final result can be overridden with the choice of result from the hacker, the final outcome is very dangerous [14].

1.2.4 Attacks on Template Databases

The templates in the database are attacked by the adversaries, reading them, modify and replace the templates, change the IDs and biometric associated with it. These type of attacks are considered to be the most threatening attacks [14]. The attacker tries to modify one or more templates in the database which could result in authorization for a fraudulent individual, or at least denial of service for the person associated with the corrupted template [1].

1.3 What is Biometric Template Protection?

The unauthorized access to biometric templates is a dangerous threat for user's security and privacy [9]. It is believed that a biometric can not be reconstructed from the extracted template. But studies in the literature such as [19] proved that a biometric can be reconstructed from its template and match score. In Ross et al. [20], an algorithm is devised where a fingerprint image is reconstructed from its matching minutiae points. If biometric data are captured by the attacker, they may be permanently lost. The privacy concern arises due to storage and misuse of biometric data in various applications. Exposure of biometric traits in diverse applications makes a serious compromise on user's privacy [12]. The distinctiveness and permanence of biometrics made it advantageous over conventional authentication systems. But, the same artifact made the permanent loss of identity in biometric systems [21]. To address this issue, biometric templates should be protected [22].

1.4 Architecture overview

The architecture of "Biometric Template Protection" as given in [23] is provided in Figure 1.4. In enrollment phase, a "Pseudonymous Identifier (PI)" and "Auxiliary Data (AD)" are obtained from a biometric sample by encoding it with "Pseudonymous Identifier Encoder (PIE)". The pair {PI, AD} is called "Renewable Biometric Record (RBR)". During verification phase, the extracted biometric feature is given to "Pseudonymous Identifier Recorder (PIR)". Hence, a "Pseudonymous Identifier (PI*)" is produced following the "Pseudonymous Identity Comparator (PIC)" that compares PI and PI*, which results in a similarity score.

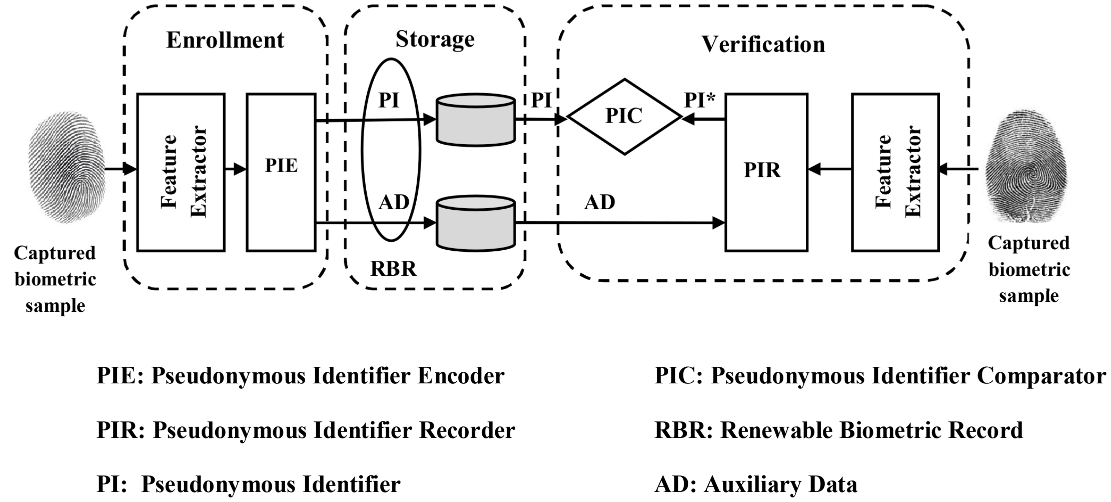

Enrollment
Storage
Verification
Feature Extractor
PIE
PI
AD
RBR
PI
PIC
PI*
AD
PIR
Feature Extractor
Captured biometric sample
Captured biometric sample
PIE: Pseudonymous Identifier Encoder
PIC: Pseudonymous Identifier Comparator
PIR: Pseudonymous Identifier Recorder
RBR: Renewable Biometric Record
PI: Pseudonymous Identifier
AD: Auxiliary Data

Figure 1.4: Architecture of Biometric Template Protection methods

1.5 Desirable properties of Biometric Template Protection techniques

According to Simoens et al. [24], the performance of "Biometric Template Protection" systems can be evaluated by three categories: "technical, protection, and operational". The technical performance can be evaluated by accuracy, throughput, storage requirements. The operational performance can be evaluated by modality independence, interoperability, Quality of Performance (QoP). The protection performance includes irreversibility and diversity of biometric information. As per ISO/IEC standard 24745 about "Biometric Information Protection", a "Biometric Template Protection" scheme should satisfy four requirements:

(1) **Diversity**: The secured templates must not be cross-matched. This ensures user's privacy.

(2) **Revocability**: If the transformed template is compromised, the protection method should be able to generate a new template from the same biometric data.

(3) **Irreversibility**: The original biometric can not be obtained by secured (or transformed) template. The computational cost should be infeasible for obtaining the original biometric data, while it should be easy to generate the secured template.

(4) **Performance**: The biometric template protection technique developed should not degrade the accuracy of the recognition system.

1.6 Evolution of Biometric Template Protection Schemes

The evolution of "Biometric Template Protection" schemes is shown in Figure 1.5. In 1996, Soutar et al. [25] built a biometric cryptosystem named it as Mytec1. Later authors enhanced its version to Mytec2 in 1998 [26]. Juels and Wattenberg [27] initiated a novel method named "**Fuzzy Commitment Scheme"** that combines error correction codes with cryptography for protecting biometric template. Later in the early 2000s, Ratha et al. [28] provided the importance of "**non-invertible transforms"**. The native biometric is transformed and can not be recovered thereby increasing security, but with the loss of accuracy. In 2002, "**Fuzzy Vault Scheme"** was presented by Juels and Sudan [29]. A key is locked using the biometric features resulting in a vault. In the same year 2002, Feng and Wah [30] proposed the key generating cryptosystem using "**Quantization Scheme"**. "**Fuzzy Extractors"** were developed in 2004 by Dodis et al. [31]. This was termed as "**Secure Sketch"** by Sutcu et al. [32]. The secure sketch approach is similar to robust hashing and similarity-preserving transformations. In 2006, Boult et al. [33, 34] developed revocable biotokens that combine both cryptosystems and features transformation, hence called "**hybrid method"**. In 2009, **"Homomorphic**

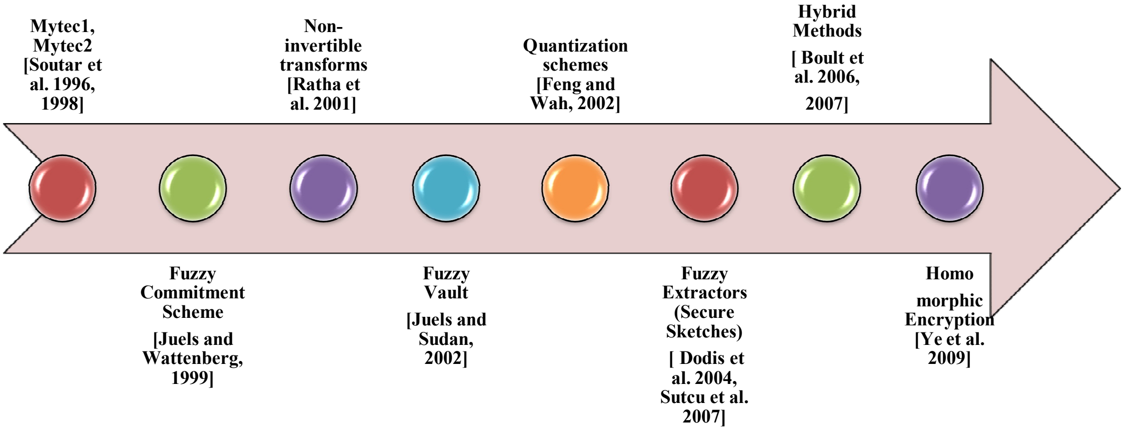

Mytec1, Mytec2 [Soutar et al. 1996, 1998]
Non-invertible transforms [Ratha et al. 2001]
Quantization schemes [Feng and Wah, 2002]
Hybrid Methods [Boult et al. 2006, 2007]
Fuzzy Commitment Scheme [Juels and Wattenberg, 1999]
Fuzzy Vault [Juels and Sudan, 2002]
Fuzzy Extractors (Secure Sketches) [Dodis et al. 2004, Sutcu et al. 2007]
Homo morphic Encryption [Ye et al. 2009]

Figure 1.5: Evolution of Biometric Template Protection Schemes

Encryption" is introduced into biometrics by Ye et al. [35]. A detailed discussion of the aforementioned techniques is given in the succeeding sections.

1.7 Categories of Biometric Template Protection schemes

The major schemes of "Biometric Template Protection" can be categorized into cancelable biometrics, biometric cryptosystems, hybrid methods, and homomorphic encryption-based methods as shown in Figure 1.6. Further, the cancelable biometric systems are divided into salting methods and non-invertible transforms, biometric cryptosystems are divided into the key binding and key generating systems. Each of these schemes is discussed in detail in the following sections.

1.8 Cancelable Biometrics

Cancelable biometric systems uses a transformation function that is dependent on a factor, called as key. Figure 1.7 represents the generic framework of a cancelable biometric system.

1.8.1 Salting

Salting is a "Biometric Template Protection" approach, where biometric features are transformed using an invertible function. In these systems, the key should be stored securely or recalled by the user for authentication as the transformation used is invertible [36]. Since keys are introduced, a low FAR results for salting methods. It is easy to revoke the compromised templates by changing user-specific keys. In case of such keys are used, they need to be presented at authentication [37]. However, salting suffers the major drawback of permanent loss of biometric data because: once the key is compromised, the adversary recovers original template since the transformation is invertible [12].

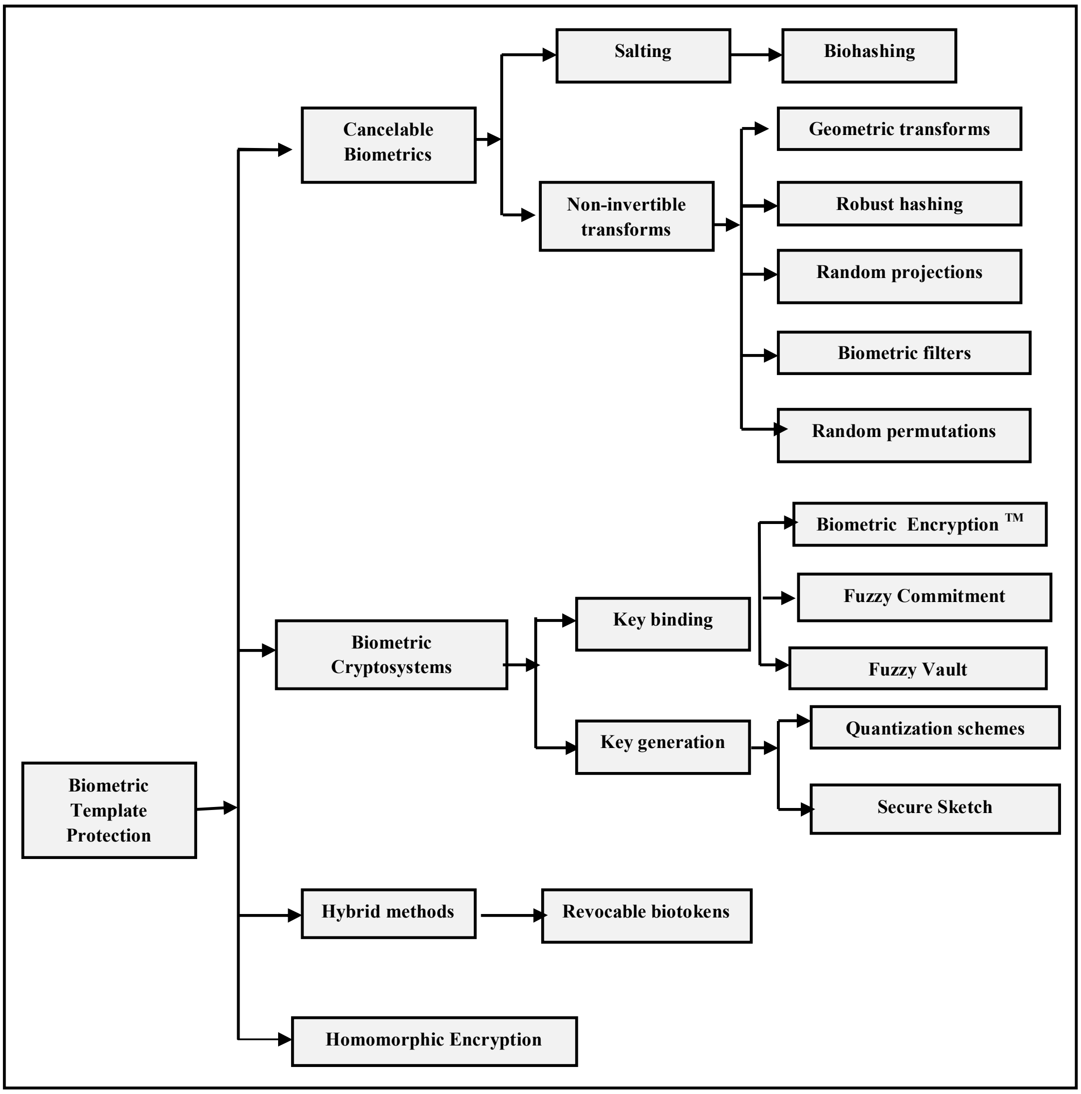


Figure 1.6: A hierarchical taxonomy of Biometric Template Protection schemes

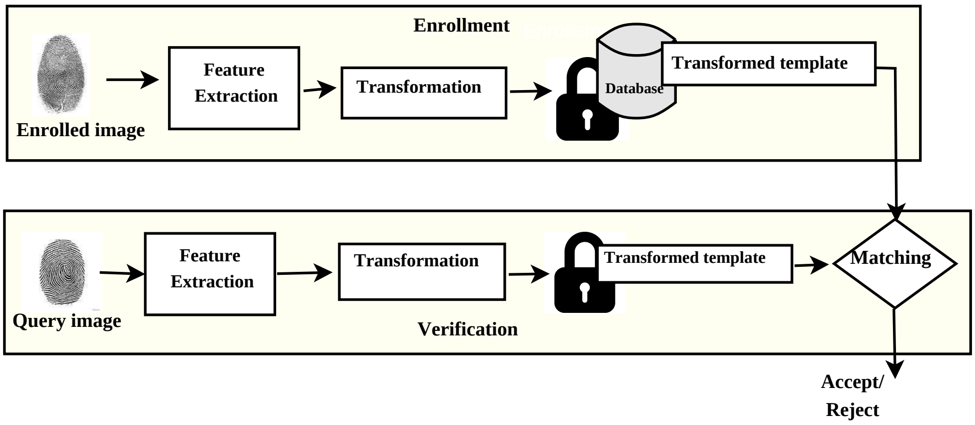


Figure 1.7: Framework of Cancelable Biometric System

1.8.2 Non-invertible transforms

Unlike salting methods, non-invertible transforms apply the irreversible transformation. The parameters of the transformation function, termed as key, should be produced during authentication as shown in Figure 1.7. This covers the major drawback of salting methods that the original biometric can not be recovered, hence increases security, but with a loss of accuracy [28]. This loss is due to the artifact that the biometric templates are difficult to align. To reduce this difficulty, later alignment-free methods were proposed. Further, non-invertible transforms achieve better revocability and diversity compared to salting approach [12]. An effort should be made in the design of non-invertible transforms to maintain the trade-off between discriminability and non-invertibility.

1.9 Biometric Cryptosystems

As defined in [38], a biometric cryptosystem refers to binding a key to a biometric feature or generating a key from the biometric feature [12]. In a biometric cryptosystem, “helper data” is used to bind/generate keys [16]. The biometric cryptosystems are classified into the “key binding” and “key generation” systems based on how the “helper data” is derived [39].

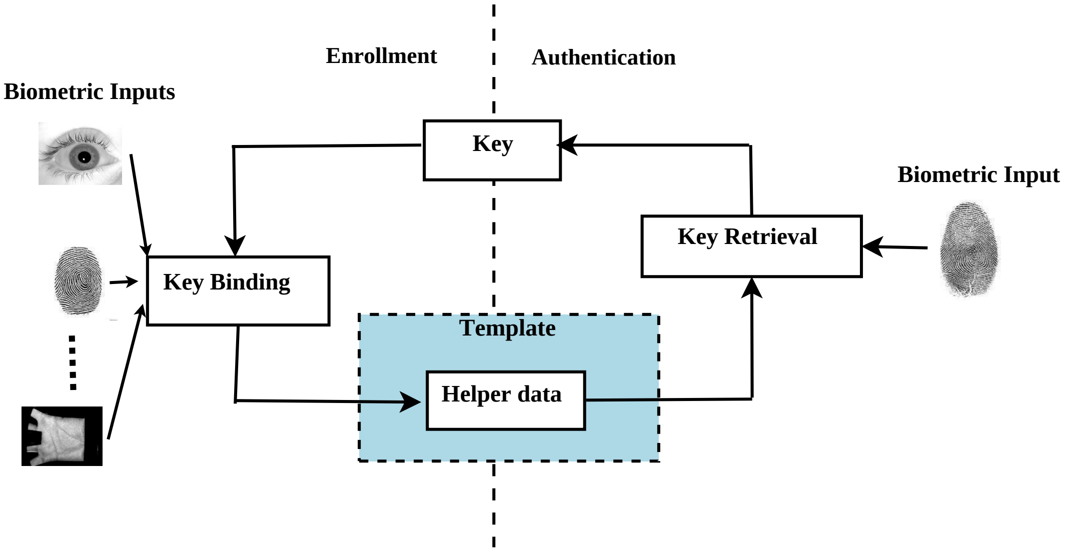


Figure 1.8: Framework of a Key Binding Biometric Cryptosystem

1.9.1 Key binding Biometric cryptosystems

In a “key binding” cryptosystem, a user-specific chosen key is bound to one or more biometric templates to obtain helper data. The combination of key and the biometric templates bound is stored as a secure template, often termed as “helper data” [40]. By a suitable decoding attempt, the keys are acquired from the helper data [16]. An overview of “key binding” cryptosystem is shown in Figure 1.8.

The first “key binding” biometric cryptosystem was developed for fingerprints by Soutar et al. [26], named $BiometricEncryption^{TM}$, also called as Mytec1, Mytec2 [25]. This biometric encryption was impractical because of mismatch between accuracy and security.

1.9.1.1 Fuzzy Commitment Schemes

Later Juels and Wattenberg [27] introduced “**fuzzy commitment scheme”** that combines cryptography and “error correcting codes”. In the enrollment stage, a random key is chosen [41]. This key is encoded using “error correcting codes”. Now, a random codeword c is generated. An XOR operation is done between the biometric feature vector and codeword [42]. This produces an encrypted template and is stored as helper

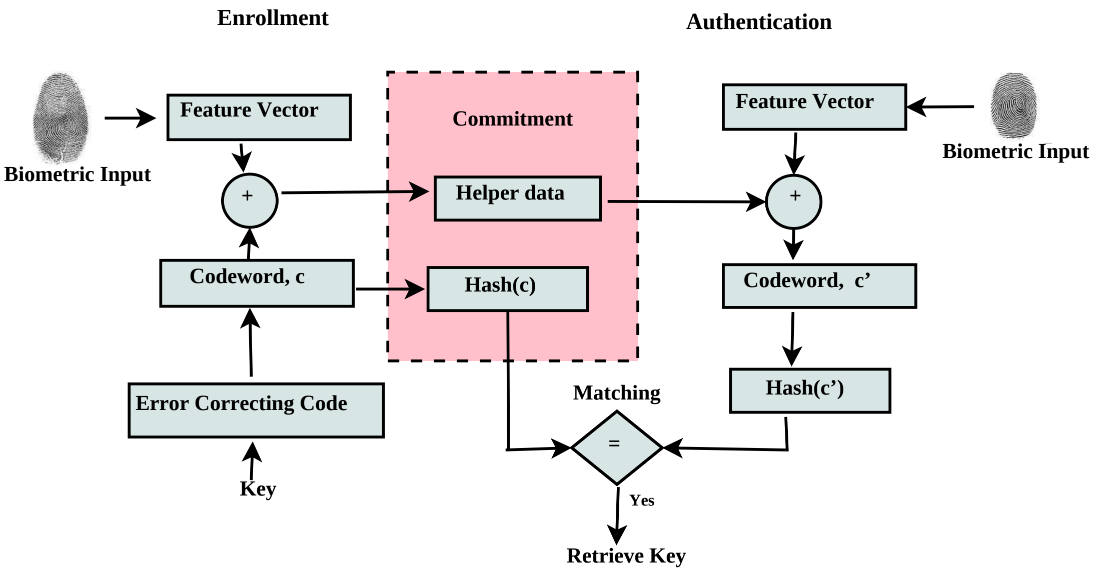


Figure 1.9: Framework of Fuzzy Commitment Scheme

data. The hash value of c ("*say hash(c)*") is computed and stored along with "helper data". During authentication, the feature vector of the query image is presented to the decoder and is XOR'd with the stored helper data to get the codeword c' [43]. If $hash(c) = hash(c')$, then $c = c'$ and hence the user is considered to be genuine and accepted else treated as an imposter and rejected [44]. The generic framework of "Fuzzy Commitment Scheme" is shown in Figure 1.9.

1.9.1.2 Fuzzy Vault Schemes

Another popular "key binding" cryptosystem is "**Fuzzy Vault Scheme"** shown in Figure 1.10. Juels and Sudan [29] is the pioneer in building a "fuzzy vault scheme". The key idea is to lock a key k by an unordered set A yielding a vault V_A. During enrollment, a polynomial p encodes key k. A is projected onto p. Chaff points are added to secure the genuine points of p. During authentication, if another set B overlaps A, key k is reconstructed. Following Jules and Sudan [29] framework, many trials to implement fuzzy vaults for fingerprints are proposed [16, 45, 46].

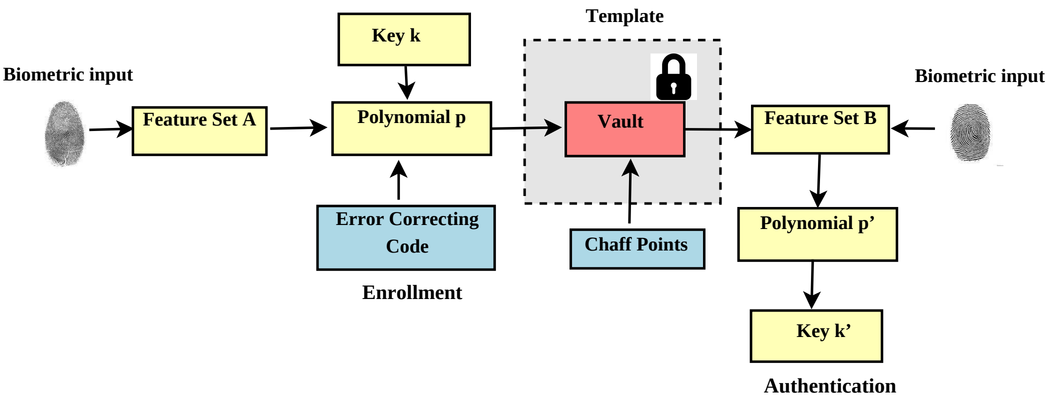


Figure 1.10: Framework of Fuzzy Vault Scheme

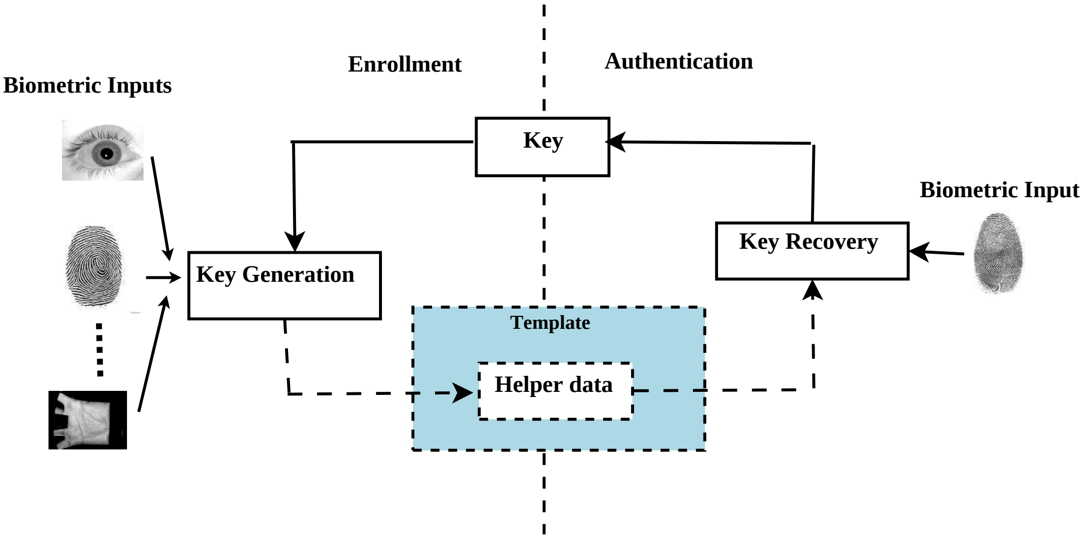


Figure 1.11: Framework of Key Generating Biometric Cryptosystem

1.9.2 Key generating Biometric cryptosystems

The "key generating" cryptosystems generate keys directly through biometric templates [47]. The framework of "key generating" cryptosystems is shown in Figure 1.11. One of the key generating cryptosystems is **Quantization scheme** [30]. In this method, helper data are quantized to obtain stable keys. These schemes take feature vectors of several biometric samples and obtain intervals of the feature elements [48]. The intervals are encoded and then stored in the form of helper data. During authentication, biometric features are calculated and mapped to the determined intervals to

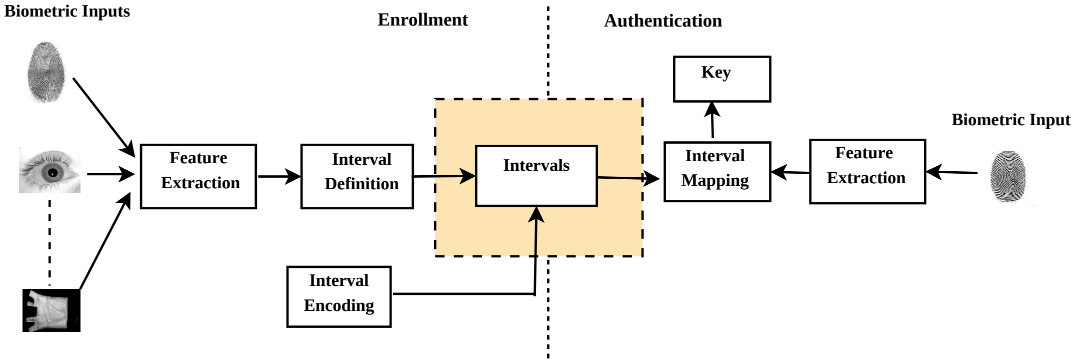


Figure 1.12: Framework of Quantization Scheme

produce a key as shown in Figure 1.12.

1.10 Hybrid methods

Hybrid methods of "Biometric Template Protection" can be designed by combining Cancelable Biometrics and Bio-Cryptosystems. A cancelable template is further encrypted to form a secure template [49]. Various schemes discussed so far can be combined by using a single trait or different traits [50]. For *e.g.*, a scheme that secures a "salted" template using a biometric cryptosystem [26, 51] may have the advantages of both salting (which provides high diversity and revocability) and biometric cryptosystem (which provides high security) approaches.

1.11 Homomorphic Encryption

As an alternative to aforementioned techniques, another approach called "Homomorphic Encryption (HE)" from the traditional cryptography is introduced into biometrics by Ye et al. [35]. HE schemes allow a limited subset of computation on the encrypted biometric data without having to decrypt it [52]. Combining HE with biometric recognition systems would meet the BTP requirements without degrading the accuracy [53].

1.12 Information Fusion in Biometrics

Multibiometric systems rely on the evidence presented by multiple sources of biometric information [50]. The factors that affect the design of a multibiometric system are the cost for acquisition of biometric traits, determining the sources of multiple evidence, the processing sequence of multiple biometric traits, the type of information and the fusion methodology used. Biometric fusion is the use of multiple inputs or methods of processing of biometric samples [54]. It has many advantages such as improved accuracy, efficiency, applicability and robustness. Biometric fusion can be done by using a single trait or multiple traits as shown in Figure 1.13.

Multi-sensor, Multi-sample, Multi-instance, Multi-algorithm systems use single trait for fusing biometric data [55]. Multi-modal systems use multiple biometric traits of a person for fusion. In multi-sensor systems, a single biometric trait is sensed through multiple sensors. In Multi-sample systems, multiple samples are collected by using the same sensor at different times. Multi-algorithmic systems use multiple feature sets extracted from same biometric data or multiple matching schemes operating on a single feature set. Multi-instance systems use multiple instances of same biometric data. Multi-modal systems combine different biometric traits for establishing identity [11]. Fusion can be done at the sensor level, feature level,

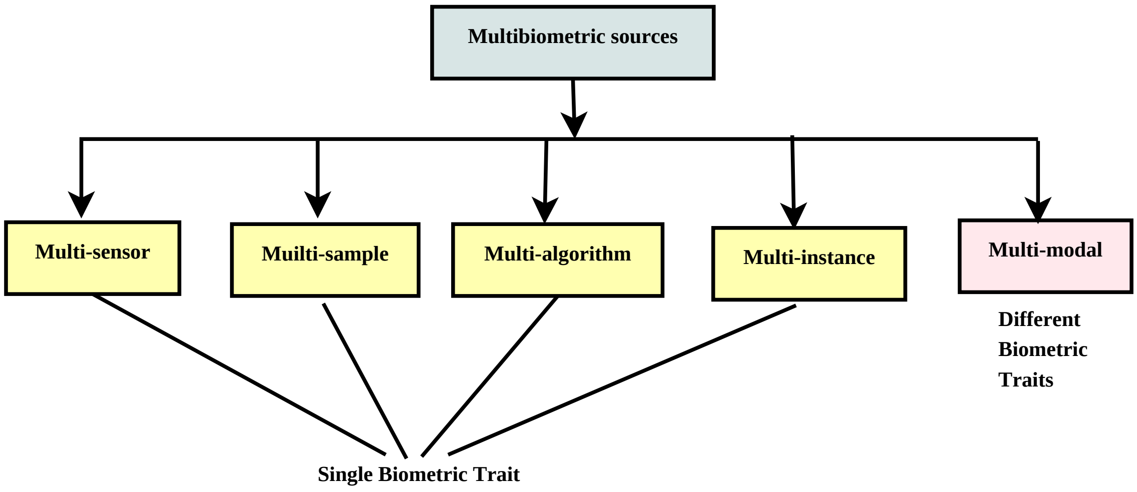


Figure 1.13: Multiple sources of evidence

decision level, and score level [54] as shown in Figure 1.14. Sensor level and feature level fusion are considered as fusion prior to matching. Decision level and score level fusion are considered as fusion after matching [11].

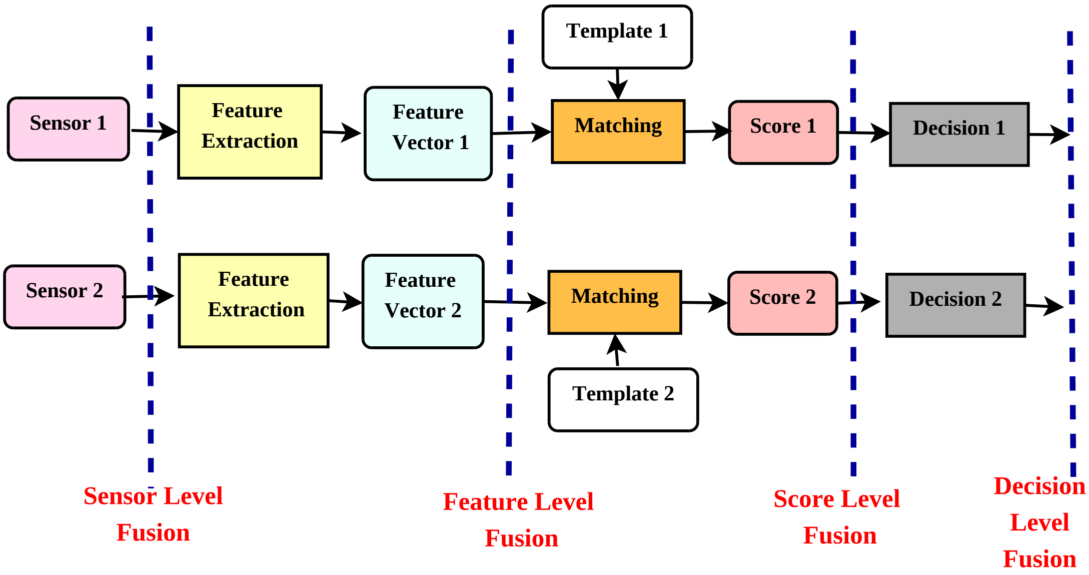


Figure 1.14: Levels of fusion in biometrics

1.13 Motivation for present work, aim, and objectives

The exposure of biometric traits in a variety of applications for verification makes a serious compromise on user's privacy. As per the standard ISO/IEC 19794-2:2005, the data format for representation of fingerprint is the fundamental notion of minutiae. But studies in the existing literature proved that fingerprint image can be reconstructed from the minutiae points [20, 56, 57, 58, 59, 60]. To address this issue, fingerprint templates are protected using various schemes such as cancelable biometrics, biometric cryptosystems, hybrid methods, and homomorphic encryption-based methods [42]. The aim of the present work is to provide methods and solutions for fingerprint template protection by maintaining the trade-off between security and accuracy of the proposed system. The objectives of the present work are:

- To get insight on the state-of-art of fingerprint template protection methods.

- To understand the existing template protection methods for fingerprints and provide solutions for better security and performance.
- To provide solutions that get resist to the potential attacks against fingerprint template protection schemes.
- To understand the research gap that needs to be addressed and to find the future directions in this area of research.

1.14 Contributions of the thesis

This thesis provides the following contributions made for fingerprint template protection.

(1) A Systematic Literature Review is conducted on "Biometric Template Protection" and identified the approaches, modalities, and fusion methods for protecting biometric templates.

(2) An algorithm for generating cancelable fingerprint templates was developed by building "k-Nearest Neighbourhood Structure (kNNS)" for all minutiae points.

(3) Proposed an algorithm for generating cancelable fingerprint templates using "Delaunay triangulation net" of fingerprint minutiae points. We propose two methods to derive a feature set from "Delaunay triangles".

(4) Proposed a feature level fusion technique on transformed features of minutiae points namely "Local Structure (LS)" and "Distant Structure (DS)".

(5) Proposed a fusion technique for fingerprint template security that combines the match scores obtained from two algorithms, using weighted sum rule and T-operators.

(6) A bio-cryptosystem for fingerprints is proposed using "Delaunay Neighbor Structures (DNS)" computed from the minutiae points. A "fuzzy commitment scheme" is employed to secure the template.

(7) Proposed a hybrid technique for protecting fingerprint templates by combining DNS and “fuzzy commitment scheme”.

(8) Proposed a hybrid technique for protecting fingerprint templates by combining multiple spiral curves and “fuzzy commitment scheme”.

(9) A hybrid technique was built by combining the bit string generated from Delaunay triangles with “convolution coding in fuzzy commitment scheme”.

1.15 Thesis organization

The rest of the chapters in this thesis are organized as follows: Chapter 2 conducts a Systematic Literature Review (SLR) on “Biometric Template Protection” techniques. This chapter reports the extraction outcomes resulted from the SLR. Chapter 3 presents two techniques for generating cancelable fingerprint templates. First, an alignment-free method was developed using “k-Nearest Neighborhood Structure (kNNS)” derived for all minutiae points. Second, a novel scheme that is developed using “Delaunay triangulation net” of fingerprint minutiae. Two methods namely “DTFS_INCIR and DTFS_AVGLO” are proposed to derive a feature set from “Delaunay triangles”.

Chapter 4 presents two fusion techniques for securing the fingerprint templates. First, two transformed features of minutiae points namely “Local Structure (LS)” and “Distant Structure (DS)” are represented as bit strings. A fusion on bit strings is done at the feature level to produce a cancelable template. Second, the two algorithms kNNS and DTFS_AVGLO are fused at score level to generate cancelable templates by using weighted sum rule and T-operators.

Chapter 5 deals with four techniques for the design of bio-cryptosystem and hybrid methods for fingerprints by employing “fuzzy commitment scheme”. First, a bio-cryptosystem was developed using “Delaunay Neighbor Structures (DNS)” formed from the fingerprint minutiae points. Second, a hybrid method was built by transforming the “Delaunay Neighbor Structures”. Third, a cancelable fingerprint cryptosystem was designed using “multiple spiral curves” and “fuzzy commitment scheme”. Fourth, the transformed features from DTFS_INCIR algorithm are employed into the “fuzzy commitment scheme” using convolution coding.

All of the proposed techniques described in Chapter 3, 4, 5 are tested on “FVC 2002 and FVC 2004 databases”. The experimental analysis is presented in each of the individual chapters. Chapter 6 concludes the thesis and outlines the future directions.

Chapter 2

A Systematic Literature Review on Biometric Template Protection

In contrast with the conventional literature review, a "Systematic Literature Review (or simply a systematic review, structured literature review or SLR) is a literature review focused on a research question that tries to identify, classify, select and synthesize the research evidence relevant to that question [2]". An SLR is a structured process that identifies, evaluates and interprets existing evidence of research that is pertinent to a specific research query. Hence, it follows a predefined search category. An SLR provides a framework for locating new research tasks. Figure 2.1 shows the steps of systematic reviews. SLR discovers gaps in ongoing research and recommends areas for further examination. This SLR differs from unstructured reviews in the following ways:

(1) This survey included emerging approach for novices called homomorphic encryption.

(2) There is no SLR on BTP till date making research difficult in identifying gaps and latest research trends. For this, we conduct an SLR on BTP and identify the approaches, modalities, fusion of modalities (multi-modal), and hybrid methods for protecting biometric templates.

(3) We answered the research questions shown in Table 2.1 according to the guidelines of SLR [2].

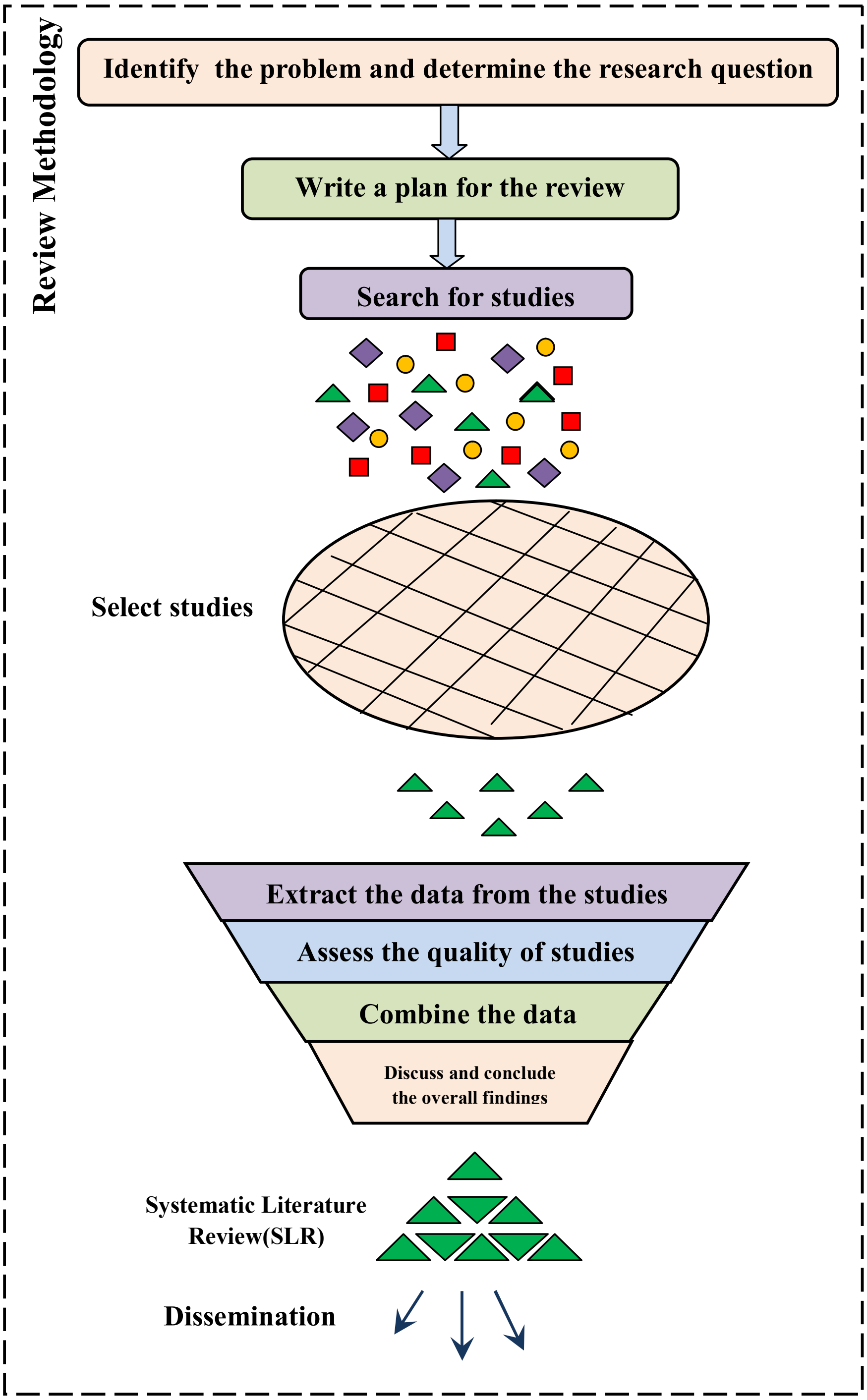


Figure 2.1: Review Methodology based on [2, 3]

S.No	Research Questions	Motivation
1	What is the current state-of-art of BTP methods?	To get Insight on the state-of-art of biometric template protection methods. This study presented and compared the different types and subtypes of BTP. The main aim of this review is to make future research through relative investigation of existing research
2	What are the main research motivations for BTP?	
3	How to identify and classify approaches to BTP?	
4	Which Modality's templates are protected using BTP schemes?	
5	What are the potential attacks against BTP schemes?	
6	How to understand the existing protection methods for better security and performance?	
7	What are the benchmark and online available datasets for evaluating BTP methods?	
8	What are the open issues and challenges for protecting biometric templates	To understand the research gap that needs to be addressed and to find the future directions in this area of research

Table 2.1: Research Questions and Motivation

(4) We included recently presented methods for protecting biometric templates and percentage of BTP categories, and the percentage of BTP schemes developed for each biometric trait.

(5) We listed out active research communities in developing BTP schemes.

(6) All online available datasets on which BTP schemes are evaluated are listed.

2.1 Sources of Information for Collecting Studies for this SLR

The following electronic databases are used for searching the studies as guided by [2]:

(1) IEEE Xplore Digital Library (ieeexplore.ieee.org)

(2) ACM Digital Library (dl.acm.org)

	Criteria	Prenominal
Inclusion	Peer reviewed research articles that propose methods, solution, experiments and evaluates different Biometric Template Protection methods	These papers contain significant content and quality through peer review. Different Biometric Template Protection solutions are studied
Exclusion	Books, Book Chapters, Thesis, Editorials, Short papers, Editorials, Abstracts, Non-English scripts, white papers	The relevant papers of authors of the thesis, books, thesis are searched in journals, conferences and incorporated. The white papers are situational and the studies that do not provide any reasonable solutions are excluded

Table 2.2: Inclusion and Exclusion Criteria

(3) ScienceDirect (www.sciencedirect.com)

(4) SpringerLink (www.spingerlink.com)

(5) Taylor & Francis Online (www.tandfonline.com)

(6) IET Digital Library (digital-library.theiet.org)

(7) Google Scholar (scholar.google.co.in)

(8) IEICE Knowledge Discovery (https://search.ieice.org/bin/search.php?lang=E)

(9) WorldScientific (www.worldscientific.com)

(10) Hindawi Publishing Corporation (www.hindawi.com)

2.2 Search Criteria

We improve the review scope of the SLR by inclusion and exclusion criteria given in Table 2.2. The various search strings used in this SLR are given in Table 2.3. This SLR included quantitative and qualitative research articles written in English from 2005-2016. We presented the research articles in journals, conferences, workshops, symposium, magazines, technical reports from industry.

S.No	Keywords	Synonyms	Year	Content Type
1	Biometric Template Protection	Biometric Template Security, Secure Biometrics, Template Protection, Biometric information protection, Biometric Security	2005-2016	Journal, Conference, Workshop, Symposium, Transactions, Magazine, Symposium, Technical reports from industry
2	Cancelable Biometrics	Cancelable templates, Non-invertible templates, Changeable Biometrics, Biohashing, Cancellable Biometrics, Secure hashing Salting Feature transformation		
3	Biometric Cryptosystem	Multi-biometric Cryptosystem Biometric Encryption Heper Data Scheme Bio Cryptosystem		
4	Biometric Key binding	Fuzzy Commitment, Fuzzy Vault, Key binding cryptosystem		
5	Revocable Biometrics	Biotokens		
6	Secure Sketch	Fuzzy Extractors		
7	Biometric Key Generation	Private Key Generation, Key Generating Cryptosystem		
8	Homomorphic Encryption	Privacy-preserving encryption		

Table 2.3: Search Strings used for SLR

2.3 Initial and Final Selection of studies

The primary research on “Biometric Template Protection” was started in 1998. But extensive work on “Biometric Template Protection” was carried out after 2005. Hence, our SLR reports more research work after 2005. We searched the studies based on inclusion and exclusion criteria described in Table 2.2. Further, we also filtered the papers at different stages as shown in Figure 2.2. Our methodical search fetched 358 articles at stage-1, filtering articles by title returned 280 articles at stage-2, by abstract and conclusion returned 210 articles at stage-3, by full-text returned 148 articles at stage-4, finally filtering by common challenges and investigation returned 119 articles.

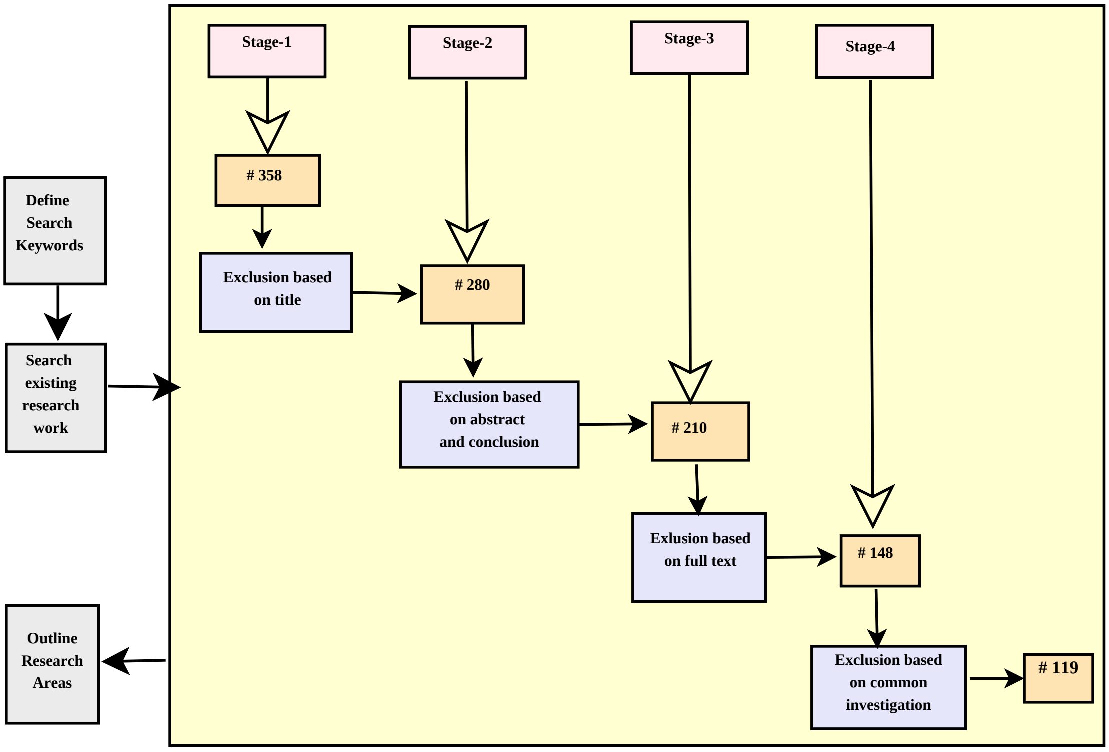

Stage-1
Stage-2
Stage-3
Stage-4
Define Search Keywords
Search existing research work
Outline Research Areas
# 358
Exclusion based on title
# 280
Exclusion based on abstract and conclusion
# 210
Exlusion based on full text
# 148
Exclusion based on common investigation
# 119

Figure 2.2: Stages in selection of studies

2.4 Selected Studies Under Cancelable Biometrics

2.4.1 Salting methods

Connie et al. [36] proposed a salting method called palmhash. The palmprint templates are hashed with pseudo-random keys. This palmhash code is stored in portable devices. Multiple sets of palmhash codes can be maintained in multiple applications. Thus the privacy and security of the applications is enhanced. In Jeong et al. [61], authors proposed an appearance-based method for generating cancelable face templates. A normalization is applied between Principal Component Analysis (PCA) and Independent Component Analysis (ICA) coefficient vectors of face templates. PCA and ICA coefficient vectors extracted from an input face image are normalized using their norm. The two normalized vectors are scrambled randomly and a new transformed face coefficient vector (transformed template) is generated by addition of the two normalized vectors. Teoh et al. [62] introduced biophasor technique for fingerprints. This method is iterated mixing of pseudo-random number with fingerprint feature. This method enables straightforward revocation of biometric template via token replacement. This work considered stolen key scenario and was extended in Teoh et al. [63] for face templates and Teoh et al. [64] for speech recognition. A biometric hash is generated from the online signature in Yip et al. [65]. This method works without template storage by incorporating user token and Wavelet-Fourier compression. The design of cancelable Palmcode was presented by Leng et al. [66]. The texture features of cancelable PalmCode were generated from Gabor filters whose parameters are randomized by the user-specific tokenised pseudo-random number. The comparison and analysis of palmhash code and palmphasor code were extensively discussed in Leng et al. [67, 68]. Teoh et al. [69] presented a salting method for face templates. Fisher discriminant analysis is used to extract discriminative projections of a face template and projected randomly in orthogonal directions. This random projection is applied with a binarization scheme to reduce intra-user variations. This maximizes the entropy of the template. A detailed description on analysis of Biohashing methods and its variants was given in Kong et al. [70].

Chin et al. [71] proposed a salting method called S-Iris encoding. The encoding method uses iterated "inner product" of iris feature and the pseudo-random number. An extended random projection method was designed in Kim et al. [72] for face databases by using better feature extraction transformation rather than template transformation. A discretized random orthonormal projection was proposed by Wang et al. [73] for face databases. In Teoh et al. [37], the quantized random projection using Johnson–Lindenstrauss Lemma was adopted for BioHash. This paper also discusses stolen-token problem. In Zuo et al. [74], authors proposed two methods: GRAY-SALT and BIN-SALT for generating cancelable iris templates. The Gabor features of Iris are added with synthetic iris patterns. The problem of outlier amplification does not effect this method. Ouda et al. [75, 76] developed cancelable iris codes based on tokenless BioEncoding methods. In contrast with Biohashing, Bioencoding is a one-factor authentication scheme where tokens are not used but solely dependent on Iris codes. The security of Bioencoding based salting methods is discussed in detail in Osama et al. [77]. The summary of salting methods is listed in Table 2.4.

Article's reference	Biometric trait	Technique	Dataset	Result
Connie et al. [36] (2005)	Palmprint	Palmhash	palm images of 50 users	FAR=0, FRR=0, EER=0
Teoh et al. [62] (2006)	Fingerprint	Biophasor	FVC 2002, DB1 and DB2	EER=0.65
Teoh et al. [69] (2006)	Face	Random Multi-space Quantization	FERET	FAR=0.002, FRR=0.001, EER=0.002, (FRR at FAR=0)=0.43
Chin et al. [71] (2006)	Iris	Random secret integration	CASIA	FAR=1.43, FRR=3.75, EER=2.59, $d^1 = 6.85$
Yip et al. [65] (2006)	Signature	Replaceable online signature hash	SVC	EER=0, Mean=0.422, std=0.009
Jeong et al. [61] (2006)	Face	Appearance based approach PCA and ICA	AR Face	-
Teoh et al. [63] (2007)	Face	Multispace Random Projections (MRP)	FERET	EER=16
Kuan et al. [78] (2007)	Signature	Mixing of BioPhasor and 2^N discretization	SVC	EER=9.38, mean=0.051, std=0.026
Him et al. [72] (2007)	Face	Extended Random Projection	AR face and BERC visual face	EER=10.915
Wang et al. [73] (2007)	Face	Discretized random orthonormal tranformation	ORL and GT	FAR=0, FRR=0, EER=0

Teoh et al. [37] (2008)	Face	BioHash	Eigen face	EER=2.11 for genuine token
				EER=0 for stolen token

Zuo et al. [74] (2008)	Iris	GRAY-SALT, BIN-SALT	MMU	-
Hirata et al. [79] (2009)	Finger vein	Number Theoritic Transform	Own dataset	-
Teoh et al. [64] (2010)	Speech	Probabilistic Random Projection (PRP)	YOHO Speech Corpus	FAR= 3.92, FRR=3.97, EER=3.95
Ouda et al. [75] (2010)	Iris	BioEncoding	CASIA v1.0	EER=1.3, d^1=6.35
Ouda et al. [76] (2010)	Iris	One-factor authentication	CASIA v3.0	EER=2.3, d^1=5.52
Pillai et al. [80] (2010)	Iris	Sectored Random Projections (SRP)	MMU	mean=0.4997, std=0.0136

Table 2.4: Summary of selected studies under Salting methods in chronological order (EER, FAR, FRR values are shown in %)

2.4.2 Non-invertible Transforms

Ratha et al. [28] introduced cancelable biometrics using non-invertible transforms. The idea of cancelable biometrics is to alter the raw biometric templates by using either feature or signal domain transformations. Templates that are in a uniform space (either feature or signal) even after transformation are called cancelable templates

[56]. Three such transformations for fingerprints, namely “Cartesian, polar and functional transformation” were proposed in Ratha et al. [81]. Before applying these transformations, the image has to be pre-aligned by estimating the singular point location and orientation. In “Cartesian transformation”, the minutiae points are evaluated after alignment of x-axis with minutia’s orientation. The Cartesian system is divided into fixed size cells. The transformation consists of changing the cell positions of the coordinate system [56]. In “polar transformation”, the minutiae points are computed in polar coordinates on the singular point. The “Cartesian and polar transformation” had a limitation that “a small change in minutia position prior to transformation may result in a large change after transformation”. This was reduced using “functional transformation” that employs locally smooth functions [81, 82]. This work was extended in Chikkerur et al. [83] by generating registration-free cancelable templates for fingerprints that result in better verification rates compared to [81]. In Hammerle et al. [84] iris texture is re-mapped by using a distorted grid mesh.

Zuo et al. [74] suggested GRAY-COMBO, BIN-COMBO transforms to generate cancelable iris template. In their work, a small performance fall was reported. Farberbock et al. [85] applied block and mesh deformation in two variants to iris images, *i.e.*, rectangular and polar iris images by extracting the iris texture from images, but before feature extraction. Row-permutations were applied to iris codes in Rathegeb et al. [86] to secure iris templates. Maiorona et al. [87, 88, 89] developed a secure recognition of biometrics whose templates are represented as a set of sequences. The transformed templates are generated by linearly convolving the set of sequences. In this work, the authors used “Hidden Markov Models (HMM)” for matching the templates. For experimental evaluation, on-line signature database is used.

A group signature scheme called as knowledge signatures were implemented for generating cancelable voice templates in Xu et al. [90]. This technique makes the users of a group to sign so that it does not reveal their identity as described in Camenisch et al. [91]. Yang et al. [92] developed a “non-invertible” transform for fingerprints by considering local and global features of minutia points. The distances between minutiae pair are perpendicularly projected to a circle. Another method for non-invertible transformations is using random projections as described in Pillai et al.

[80]. Sectored Random Projections for Iris codes were proposed which performed better compared to random projections and robust to the occlusions of eyelids and eyelashes. The authors extended their work in Pillai et al. [93] by using sparse representations.

A new representation of minutiae points of a fingerprint image is done using bit strings in Lee et al. [94]. The minutia points of a fingerprint image are mapped to a 3D array that is divided into small cells. A bit string is generated by finding the cells that include the minutiae points. In Nanni et al. [95] the local, global and regional matchers are merged to protect online-signature templates. In Ahmad et al. [97], a pair-polar relationship of minutiae is obtained. A correlation based filter method using chip matching was proposed in Takahashi et el. [98]. The use of "Correlation Invariant Random Filtering (CIRF)" results in the same accuracy as that of Fingerprint Verification Competition. Cancelable fingerprint templates were designed based on "Densely Infinite To One Mapping (DITOM)" approach in Wang et al. [99]. Das et al. [100] designed an alignment free hashing algorithm based on "Minimum Distance Graphs" for fingerprints. The feature set is constructed from inter-minutiae distance vectors originating from the core point. Matching between hashes is implemented using corresponding search algorithm.

Jin et al. [118] presented a method to generate a revocable fingerprint template in terms of bit-string from a set of minutiae points via a polar grid based 3-tuple quantization technique. Moujahdi et al. [101] developed a non-invertible transform by building a spiral cube for face biometric using chaotic logistic map for random projection of feature vectors. A non-invertible transform is built by the intersection of hyperplanes. Ferrara et al. [102] built a non-inveritble Minutiae Cylinder Code (pMCC) for fingerprints as an enhancement to Cappelli et al. [119]. Wong et al. [103] generated cancelable fingerprint templates using a multi-line code of minutiae points. The authors extended their work in Wong et al. [104] by building an enhanced multi-line code for fingerprints. An adaptive bloom filter application to the iris codes was proposed in Rathgeb et al. [106, 107]. The application of this Bloom filter-based approach enabled template protection, compression and acceleration of biometric identification. The authors extended their work in Rathgeb et al. [109]

Article's reference	Biometric trait	Technique	Dataset	Result
Ratha et al. [81] (2007)	Fingerprint	Cartesian, polar, functional transformation	IBM-99	GAR=0.90
Chikkerur et al. [83] (2008)	Fingerprint	Localized texture based representatiom	IBM-99	GAR=1.0
Zuo et al. [74] (2008)	Iris	GRAY-COMBO, BIN-COMBO	MMU	ROC curve, Histogram of scores
Mairorana et al. [87, 88] (2008)	Online signature	HMM, Stemming, function mixing, Shifting function	MCYT online signature corpus	EER=13.8
Xu et al. [90] (2008)	Voice	Knowledge Signatures	Collection of 900 samples	GAR=99.5
Yang et al. [92] (2009)	Fingerprint	Perpendicular projection of pair minutiae to a circle	FVC 2002 DB2	EER=13
Rathgeb et al. [86] (2010)	Iris	Local intensity variations in iris texture	CASIA v3	(FRR at FAR=0) = 1.978, EER=1.016
Farberbock et al. [85] (2010)	Iris	Rectangular and polar Block-remapping, Mesh Warping	CASIA v3	Block re-mapping, EER=3.1 Mesh Warping EER=2.0
Mairorana et al. [89] (2010)	Online signature	Baseline approach, mixing approach, Shifting approach, HMM	MCYT online signature corpus	EER=9.12
Pillai et al. [80] (2010)	Iris	Sectored Random Projections (SRP)	MMU	mean=0.4997, std=0.0136
Lee et al. [94] (2010)	Fingerprint	Minutia based bit strings	FVC 2004 DB1, DB2, DB3	DB1 EER= 10.3, mean=0.117 DB2 EER=9.5, mean=0.133 DB3 EER=6.8, mean=0.168
Nanni et al. [95] (2010)	Online Signature	combining local, global, regional matchers	MCYT	EER=3
Jin et al. [96] (2011)	Fingerprint	Minutiae Vicinity Decomposition (MVD)	FVC 2002 DB2	EER=3.90
Pillai et al. [93] (2011)	Iris	Random Projections Sparse Representation	ICE2005, ND-IRIS-0405 MBGC videos	-
Ahmad et al. [97] (2011)	Fingerprint	Pair-polar coordinates	FVC 2002 DB1, DB2, DB3	DB1 EER= 9 DB2 EER=6 DB3 EER=27
Takahashi et al. [98] (2011)	Fingerprint	Correlation Invariant Random Filtering (CIRF)	Veridicom 5th Sense	-
Wang et al. [99] (2012)	Fingerprint	Densely Infinite To One Mapping (DITOM)	FVC 2002 DB1, DB2, DB3	DB1 EER=3.5 DB2 EER=5 DB3=7.5

Table 2.5: Summary of selected studies under Non-inveritble transforms in chronological order (EER, FAR, FRR values are shown in %)

Article's reference	Biometric trait	Technique	Dataset	Result

Das et al. [100]
(2012)
	Fingerprint	Minimum Distance
		Graphs
			FVC 2002
			DB1, DB2
				FVC 2002 DB1,
				EER=2.27
				FVC 2002 DB2,
				EER=3.79

Moujahdi et al. [101] (2012)	Face	Random projection Logistic map	YALE, UMIST	EER=3.34

Ferrara et al. [102]
(2012)
	Fingerprint	Non-invertible
		Minutiae Cylinder
		Code
			FVC 2002 DB1, DB2,
			DB3, DB4
			FVC 2006 DB2
				For FVC 2006 DB2,
				FNMR=1.42 at FMR=0
				EER=0.12
				$FNMR_{1000}$=0.14

Wong et al. [103, 104] (2013)	Fingerprint	Multi-line Code	FVC 2002 DB1,DB2 FVC 2004 DB1, DB2	FVC 2002 DB1, EER= 1.97 DB2, EER=2.54 FVC 2004 DB1, EER= 6.53 DB2, EER=9.2
Yang et al. [105] (2013)	Fingerprint	Delaunay triangle local structure	FVC 2002 DB1,DB2	FVC 2002 DB1, EER= 5.93 For DB2, EER=4
Rathgeb et al. [106, 107] (2014)	Iris	Adaptive bloom filters	CASIA-v3	1-FNMR=97.95 at FMR=0.01 EER=1.14

Ferrara et al. [108]
(2014)
	Fingerprint	Non-invertible
		Minutiae Cylinder
		Code
			FVC 2006 DB2	FNMR=1.42 at FMR=0
				EER=0.13

Rathgeb et al. [109]
(2014)
	Iris	Mixing iris codes
		Adaptive bloom filters
			IITD v1.0	EER=0.5
				Compression=10

Jin et al. [18] (2014)	Fingerprint	Randomized Graph-based Hamming Embedding (RGHE)	FVC 2002 DB1, DB2, FVC 2004 DB1, DB2	FVC 2002 DB1 EER=4.36 DB2 EER= 1.77 FVC 2004 DB1, EER=24.71 For DB2, EER=21.82
Prasad et al. [110] (2014)	Fingerprint	Multi-line neighboring relation	FVC 2002 DB1, DB2, DB3	DB1 EER=0.62 DB2 EER=1.33 DB3 EER=2.64
Jin et al. [111] (2014)	Fingerprint	2D random projected MVD	FVC 2002 DB1, DB2	DB1 EER= 3.07 DB2 EER= 1.02
Prasad et al. [112] (2014)	Fingerprint	Minimum Spanning Tree	FVC 2002 DB1, DB2, DB3 FVC 2004 DB1, DB2	FVC 2002 DB1 EER= 0.8, DB2 EER=0.8, DB3 EER=2.4, FVC 2004 DB1 EER=4, DB2 EER=4
Moujahdi et al. [113] (2014)	Fingerprint	Fingerprint Shell	FVC 2002 DB1, DB2	EER for DB1= 2.03, EER for DB2=1.01, DB1 d^1 2.4703 DB1 K-S test= 0.7812
Leng et al. [114] (2014)	Palmprint	Multi-directional 2D palmphasor	PolyU v2	EER=0.1938
Escobar et al. [115] (2015)	Fingerprint	Chaotic Encryption using Logistic map	-	-
Wong et al. [21] (2016)	Fingerprint	Kernel PCA enabled Multi Line Code	FVC 2002 DB1, DB2 FVC 2004 DB1, DB2	FVC 2002 DB1, EER= 1.61 DB2, EER= 1.69 FVC 2004 DB1, EER= 3.73 DB2, EER= 3.74

Wang et al. [116]
(2016)
	Fingerprint	Blind system
		approach
			FVC 2002 DB1, DB2
			DB3
				EER for DB1=4
				DB2= 3
				DB3= 8.5

Wang et al. [117]
(2017)
	Fingerprint	Partial Hadamard
		transform
			FVC 2002 DB1, DB2
			DB3
				EER for DB1=1
				DB2=2
				DB3=5.2

Table 2.6: Summary of selected studies under Non-inveritble transforms in chronological order (EER, FAR, FRR values are shown in %)

by mixing binary iris templates of both eyes of a single subject at a feature level. This method reported error rates below 0.5%. The unlinkability and irreversibility analysis for adaptive bloom filter based method was given in Rathgeb et al. [120]. In Prasad et al. [112], authors used “minimum spanning tree” based methods for generating cancelable templates for fingerprints. Two methods namely Cartesian space method and boundary based method were used to generate chain codes from fingerprint minutiae. Bit-strings are produced using plane based quantization method. A cancelable fingerprint template was obtained using random projected “Minutiae Vicinity Decomposition (MVD)” features [96, 111].

A “non-invertible Randomized Graph-based Hamming Embedding (RGHE)” for generating cancelable fingerprint template is developed in Jin et al. [18]. A multi-neighboring relation was proposed in Prasad et al. [110] where plane based quantization technique followed by “Discrete Fourier Transform (DFT)” and user-specific random matrix multiplication was used to generate renewable templates for fingerprints. Leng et al. [114] presented a cancelable palmprint method based on “multi-directional two-dimensional PalmPhasor fusion”. Escobar et al. [115] proposed a non-invertible transform for fingerprints using chaotic encryption based on the logistic map and Murillo-Escobar’s algorithm. Wong et al. [21] built a binary cancelable template using minutia descriptor called multi-line code (MLC). The MLC template is transformed to a fixed-length bit-string using “Kernel Principal Components Analysis (KPCA)”.

Wang and Hu [116] presented a blind system identification approach for cancelable fingerprint template generation. A binary string derived from quantized pair-minutiae vectors is protected using its frequency samples. Wang et al. [117] proposed a partial Hadamard transform for generating cancelable fingerprint templates. This method preserves the stochastic distance between binary vectors after the transformation. The summary of non-invertible transforms is listed in Tables 2.5 and 2.6.

2.5 Selected Studies Under Biometric Cryptosystems

2.5.1 Key-binding Bio-cryptosystems

2.5.1.1 Fuzzy Commitment schemes

Hao et al. [41] developed a technique to combine cryptography to the iris biometric. A binary string is generated from iris codes. The authors coined a "two-layer error correction" method that combines "Hadamard" and "Reed-Solomon" codes. In Zheng et al. [43], other than using error correcting codes in "fuzzy commitment scheme", error tolerant lattice functions are used. Simulation is done on Iris plants. Bringer et al. [121] introduced a min-sum decoding of "fuzzy commitment scheme" for iris codes. Reed-Muller codes are used to build a matrix from the iris codes. The authors extended their work in Bringer et al. [124] by detecting the suitable error-correcting code regarding a given database of biometric data. Teoh et al. [122] built a "Randomized Dynamic Quantization Transformation Technique (RDQT)" for fingerprint data using the "fuzzy commitment scheme".

Mairorana et al. [123] built a user-adaptive error correction method for online signatures. In Meng et al. [125], authors built a "Near-Infrared (NIR)" face biometric by enhancing BioHash algorithm. An NXOR mask is inputted to the "error correcting code". Rathgeb et al. [126] provided a methodical construction of using "fuzzy commitment schemes" for iris by making an analysis of error distributions. In Lu et al. [127], the face features based on "Principal Component Analysis (PCA)" is integrated with "fuzzy commitment scheme". A binarization algorithm for on-line signatures based on "fuzzy commitment scheme" was applied in Mairona et al. [128]. This method is based on function-based signature verification and achieved better accuracy than feature based representation methods.

Nandakumar et al. [129] proposed "Binarized Phase Spectrum (BiPS)" for fingerprints. A binary string is obtained from minutiae points by quantizing Fourier phase spectrum. The BiPS representation is secured by "fuzzy commitment scheme" employing turbo codes. Ignatenko et al. [135] investigated the privacy leakage in

Article's reference	Biometric trait	Technique	Dataset	Result
Hao et al. [41] (2006)	Iris	Two-layer error correction using Hadamard and Reed-Solomon codes	700 images 70 eyes, with 10 samples of each eye	FRR=0.47 at (FAR=0)

Zheng et al. [43]
(2006)
	Iris	Error-tolerent
		lattice functions
			-	FAR= 3.3
				FRR=0.6

Bringer et al. [121] (2007)	Iris	Min-sum decoding Reed-Muller codes	ICE	FRR=5.62 at (FAR $<10^{-5}$)
Teoh et al. [122] (2007)	Fingerprint	RDQT Reed-Solomon Code	FVC 2002 DB1	FRR=0.9 at FAR=0
Mairona et al. [123] (2008)	On-line Signature	Adpative selection of error correction	MCYT	FRR=13.07 at FAR=4

Bringer et al. [124]
(2008)
	Iris
	Fingerprint
		Min-sum decoding
		Reed-Muller codes
			ICE
			CASIA
			FVC 2000
				For ICE,
				FRR=5.62 at (FAR $<10^{-5}$)
				For CASIA,
				FRR=6.65 at FAR=0
				For FVC 2000
				FRR=2.73 at FAR=5.53

Meng et al. [125] (2009)	Face	Enhance BioHash NXOR mask	NIR face	FRR=7.99 at FAR=0.11
Rathgeb et al. [126] (2009)	Iris	Block level ECC	CASIAv3	FRR=4.64 at FAR=0
Lu et al. [127] (2009)	Face	PCA Gray Codes	CMU PIE	FRR=0.47 at FAR=0
Mairo et al. [128] (2010)	Online-Signature	Signature binarization algorithm	MCYT	EER= 1.43
Nanda kumar et al. [129] (2010)	Fingerprint	BiPS Turbo codes	FVC 2002 DB1, DB2	DB1 FNMR=16.2 at FMR=0, EER=2.1 DB2 FNMR=12.6 at FMR=0, EER=2.1
Rathgeb et al. [130] (2010)	Iris	Intra-class error analysis Hadamarad codes	CASIA v3	FRR=4.92 at FAR=0
Rathgeb et al. [40] (2011)	Iris	Reliability balanced feature level fusion Hadamard codes	CASIA v3	FRR=4.58 at FAR<0.01
Li et al. [131] (2012)	Fingerprint	Minutia triplets Combining ECC	FVC 2002 DB2	FRR=4.85 at FAR=0
Imam verdiyev et al. [132] (2013)	Fingerprint	Minutia texture descriptors	FVC 2002 DB2	FRR=95.3 at FAR=0
Yuan et al. [133] (2014)	Face and Ear	Feature level fusion using PCA BCH Encoding	ORL face USTB ear3	EER=6.1
Billeb et al. [134] (2015)	Voice	Guassian mixture Universal Background Hadamard code	text independent digit corpus	FNMR= 0.9764 at FMR=0
Wang et al. [55] (2014)	Face and Iris	Score level fusion Aczel-Alsina T-norm	CASIA iris NVIE face	1.163×10^{-3}

Table 2.7: Summary of selected studies under Key binding cryptosystems using fuzzy commitment scheme in chronological order (EER, FAR, FRR values are shown in %)

“fuzzy commitment schemes” in four scenarios, namely “memoryless totally symmetric, memoryless input-symmetric, memoryless, and stationary ergodic” of biometric data statistics. Rathgeb et al. [130] used intra-class error analysis to rearrange iris codes. Rathgeb et al. [40] presented a feature level fusion method for “fuzzy commitment schemes” on iris data. The binary data is rearranged and combined in a way that error correction abilities are utilized properly. The authors presented a statistical attack for Iris “fuzzy commitment schemes” in Rathgeb et al. [136, 137]. The authors examined the effect of iris blur and noise to “fuzzy commitment schemes” in Rathgeb et al. [138].

Zhou et al. [139, 140] shows that secrets from iris codes can be retrieved by Markov model in an iris fuzzy commitment. The authors measure the privacy and secrecy of iris “fuzzy commitment schemes” using “information theoretical metrics” in Zhou et al. [141]. Kelkboom et al. [142] provided an explanation for large deviation in key sizes for “fuzzy commitment schemes”. Li et al. [131] developed a minutiae triplet binary feature method for fingerprints. Using the binary string as input to “fuzzy commitment scheme”, various “error correction codes” are combined to design a fingerprint cryptosystem. Imamverdiyev et al. [132] designed a fingerprint cryptosystem using texture descriptors such as “Gabor filter, Local Binary Pattern (LBP), and a Local Direction Pattern (LDP), and their combinations”. The texture descriptors are binarized and given as input to the “fuzzy commitment scheme”.

Yuan et al. [133] built a multimodal cryptosystem by fusing human face and human ear at feature level using principal component analysis. In Billeb et al. [134], the authors used binarization technique to extract voice reference data from speaker models like Gaussian mixture models and universal background models. Wang et al. [55] presented a multi-biometric template protection technique using face and iris data. Aczel-Alsina triangular norm is used to perform a score level fusion of the proposed method. The summary of “key binding” cryptosystems using “fuzzy commitment scheme” is listed in Table 2.7.

2.5.1.2 Fuzzy Vault Schemes

Uludag et al. [45] presented a fuzzy vault method by securing 128-bit AES keys in minutiae points of a fingerprint. This method derived experimental results and analysis for FVC 2002 DB2 databases in Uludag et al. [46]. Nandakumar et al. [143] built a fully automatic fingerprint cryptosystem using “fuzzy vault scheme”. High curvature points obtained from orientation field of minutiae points are used as helper data to align the template and query minutiae. The authors extended their work in Nandakumar et al. [51] by using a password for hardening the fuzzy vault. Nagar et al. [144] used minutiae descriptors that record orientation and ridge frequency data in a minutia’s neighborhood for locking the polynomial evaluations in the fuzzy vault.

Nyang et al. [145] built a fuzzy vault for face biometric using weighted features. Lee et al. [146] proposed a “fuzzy vault” system using local iris features. Numerous iris features from local regions are extracted and a clustering method is applied to get the exact features. A shift matching method is used for aligning the iris templates with iris data. Kelkboom et al. [147] presented multi-algorithm fusion at “feature, score, and decision-level” for 3D face images. Zhou et al. [148] generates artificial minutiae from a PIN or password for fingerprints. Now, the feature sets of genuine minutiae and artificial minutiae are fused to generate a protected template.

Nagar et al. [149] developed a multi-modal cryptosystem that fuses iris, fingerprint, and face at the feature level. This work used both “fuzzy vault” and “fuzzy commitment” methods for designing the cryptosystem. “Berlekamp Massive algorithm” is used to decode the vault. Nguyen et al. [150] proposed a chaff point generation method for fingerprint “fuzzy vault” by dividing fingerprint image into cells. Each cell generates a random chaff point. Eskander et al. [151, 152, 153] designed a machine learning approach that adapts user key size for off-line signature verification. Bhateja et al. [154] described an on-line signature based cryptosystem using artificial neural network classifier. Firstly, the signature is broken into time slices, then a “weighted back propagation algorithm” is employed to train each slice. “AdaBoost algorithm” is employed to combine the decisions of different networks.

Article's reference	Biometric trait	Technique	Dataset	Result
Uludag et al.[45] (2005)	Fingerprint	CRC Encoding and Decoding	IBM-GTDB	FRR=27 at FAR=0
Uludag et al.[46] (2006)	Fingerprint	CRC Encoding and Decoding	FVC 2002 DB2	GAR=72.6 at FAR=0

Nanda
kumar et al. [143]
(2007)
	Fingerprint	High curvature points
		Orientation field
			FVC 2002 DB2
			MSU-DBI
				For FVC 2002 DB2
				GAR=97 at FAR=0.24
				For MSU-DBI
				GAR=96.9 at FAR=0.16

Nanda
kumar et al. [51]
(2007)
	Fingerprint	Password Hardening
		Orientation field
			FVC 2002 DB2
			MSU-DBI
				for FVC 2002 DB2
				GAR=90 at FAR=0
				For MSU-DBI
				GAR=80.6 at FAR=0

NYang et al.[145] (2007)	Face	Weighted features	-	-

Nagar et al.[144]
(2008)
	Fingerprint	Minutiae
		Descriptors
			FVC 2002 DB2
			FVC 2002 DB2
				for FVC 2002 DB2
				GAR=95 at FAR=0.01

Lee et al.[146]
(2008)
	Iris	Multiple features
		from multiple
		local regions
		shift-matching
			BERC v1
			CASIA v1
			CASIA v3
				For BERC v1
				GAR=99.925 at FAR=0.035
				for CASIA v1
				GAR=83.408 at FAR=0.0235
				For CASIA v3
				GAR=91.171 at FAR=0.0275

Kelkboom et al.[147] (2009)	3D Face	multi-algorithm fusion	FRGC v2	EER=2.58
Zhou et al.[148] (2011)	Fingerprint	Artificial minutiae	NIST SD14	FNMR=0.051625 at FMR=0
Nagar et al.[149] (2012)	Iris, Fingerprint face	Feature level fusion Secure sketch	CASIA v1 FVC 2002 DB2 XM2VTS	GAR=75
Liu et al.[155] (2012)	Fingerprint	Lorez Chaotic System	FVC 2004 DB2	-
Bhateja et al.[154] (2014)	Online Signature	Artificial Neural Network based classifier AdaBoost algorithm	SVC 2004	GAR=17.78 at FAR=2.24
Liu et al.[156] (2014)	Palmprint	multidimensional fuzzy vault	HA-BJTU	GAR=80
Rathgeb et al.[157] (2015)	Iris	Ridge features	IITD Iris	GAR=97.21 at FAR=0

Nguyen et al.[158]
(2015)
	Fingerprint	Ridge features	FVC 2002 DB1,DB2
			FVC 2004 DB1, DB3
				for FVC 2002 DB1
				GAR=86 at FAR=0
				for FVC 2002 DB2
				GAR=89 at FAR=0

Tams et al.[159] (2015)	Fingerprint	minutia orientation, frequency, local minutia structures	FVC 2002 DB1	GAR=82 at FAR=0
Bansal et al.[160] (2015)	Fingerprint	Hadamard transformation	FVC 2002 DB1	GAR=77 at FAR=0

Caili et al.[161]
(2016)
	Fingerprint	Pair-polar (P-P)
		Structures
			FVC 2000 DB1
			FVC 2002
			FVC 2004 DB2
			FVC 2006 DB2, DB3
				For FVC 2002 DB2
				FRR=11.50 at FAR=0
				EER=3.37
				For FVC 2006 DB2
				FRR=5.78 at FAR=0
				EER=1.59

Table 2.8: Summary of selected studies under key binding cryptosystems using fuzzy vault scheme in chronological order (EER, FAR, FRR values are shown in %)

Liu et al. [156] proposed a multidimensional “fuzzy vault” that can handle intra-class variances. A subspace error-tolerant mechanism is embedded into “fuzzy vault scheme”. Yang et al. [162] developed a fingerprint bio-cryptosystem by modifying Voronoi Neighbor Structures (VNS). A secure sketch called PinSketch [163] is used for securing the fingerprint template. Rathgeb et al. [157] presented a “fuzzy vault” system by transforming bloom filter-based representation of iris templates into a unordered set of integer values. Nguyen et al. [158] developed a fingerprint “fuzzy vault” based on ridge features information and noise generation technique to improve the security of the system. Tams et al. [164] redesigned a minutia-based fingerprint “fuzzy vault” execution, which is resistant to attacks via record multiplicity. The authors extended their work in Tams et al. [159] by improving “fuzzy vault scheme” that employs minutiae orientation, frequency descriptors, and local minutia structures.

Bansal et al. [160] applied Hadamard transformation for fingerprint “fuzzy vault”. Laftih et al. [165] presented an attack by altering the original data in fuzzy vault biometric cryptosystems. Caili et al. [161] proposed fingerprint cryptosystem using pair-polar (P-P) minutiae structures. The summary of “key binding” cryptosystems using “fuzzy vault scheme” is listed in Table 2.8.

2.5.2 Key generating Biometric cryptosystems

Monrose et al. [166, 167] proposed a method for generating cryptographic keys repeatedly from spoken user input. A user speaking chooses a password and generates a cryptographic key to encrypt files. Feng et al. [30] applied quantization scheme on on-line signatures and generated the 40-bit hash. Dynamic time wrapping is used to match enrolled and query templates. Hoque et al. [47] generated keys from signatures by dividing feature space into subspaces and in turn into cells. Rathgeb et al. [48] applied interval mapping technique to generate cryptographic keys from iris characteristics.

Cappelli et al. [168] applied “Randomized Biometric Templates (RBT)” algorithm to voice biometrics. The voice signal is split into time quanta. Now, features are extracted from each quantum for evaluation. “Self-Organizing Map (SOM)” is used to model the speech signal. Lifang et al. [169] generated keys using face features. “Principal Component Analysis (PCA)” is applied to face image which generates a binary vector. Distinguishable bits are selected to form a key. The optimal bit order number is saved in a look-up table. Liu et al. [170] developed a “key generating” system from fingerprints using fuzzy extractors and combining minutiae based and image based features. The summary of “key generating” cryptosystems is listed in Table 2.9.

Article’s reference	Biometric trait	Technique	Dataset	Result
Monrose et al. [166, 167] (2001)	Voice	Speaker Password	90 samples	False Neagatives <9
Feng et al. [30] (2002)	Signature	BioPKI	-	EER= 8
Hoque et al. [47] (2008)	Signature	Feature Subspaces	4500 samples	FAR=7.6
Rathgeb et al. [48] (2009)	Iris	Interval Mapping	-	FRR=36.5 at FAR=0.07
Carrara et al. [168] (2010)	Voice	RBTs	TI46	FRR <10
Lifang et al. [169] (2010)	Face	PCA Reed-Solomon Encoding	ORL	FRR =97 at FAR=0
Liu et al. [170] (2010)	Fingerprint	Fuzzy extractor minutiae based features image based features	FVC 2002 DB1, DB2	For DB1, EER=3.81 For DB2, EER=1.07

Table 2.9: Summary of selected studies under Key-generating cryptosystems in chronological order (EER, FAR, FRR values are shown in %)

2.6 Selected Studies Under Hybrid Methods

Boult et al. [33] introduced revocable biotokens for face biometrics. This technique divides the data into two parts “fractional part” and “integer part”. The “fractional part” does the transformation and “integer part” is encrypted. The authors extended their work for fingerprints in [34]. Feng et al. [171] developed a “three-step hybrid algorithm based on random projection, discriminability-preserving (DP) transform, and fuzzy commitment scheme”. Nagar et al. [50] built a hybrid cryptosystem with minutiae descriptors for fingerprints. This work used both “fuzzy vault” and “fuzzy commitment” schemes for building the cryptosystem. The helper data extraction involves “fuzzy vault” encoding and the ordinate values are secured using “fuzzy

Article's reference	Biometric trait	Technique	Dataset	Result
Boult et al. [33] (2007)	Face	Revocable Biotoken	FERET	GAR=100 at FAR=0
Boult et al.[34] (2007)	Fingerprint	Revocable Biotoken	FVC 2000, FVC 2002, FVC 2004 DB1, DB2	FVC 2000 DB1, EER=0.029 FVC 2000 DB2, EER=0.025 FVC 2002 DB1, EER=0.021 FVC 2002 DB2, EER=0.012 FVC 2004 DB1, EER=0.086 FVC 2004 DB2, EER=0.075
Feng et al. [171] (2010)	Fingerprint	random projection DP transform fuzzy commitment	FERET CMU-PIE FRGC	-
Nagar et al. [50] (2010)	Fingerprint	Minutiae descriptors	FVC 2002 DB2	GAR=97 at FAR=0
Liu et al. [172] (2014)	Palmprint	Random multi-space projection Heterogeneous fuzzy vault	HA-BJTU	EER=0.08
Chin et al. [173] (2014)	Fingerprint Palmprint	Random Tilling 2^N discretization	FVC 2002 DB1 FVC 2004 DB1 PolyU	EER=0 at d^1=10.51
Kumar et al.[174] (2015)	Iris, Palmprint	Cell Array	PolyU IITD palm IITD iris	FRR=0.91 at FAR=0
Jin et al. [175] (2016)	Fingerprint	ECC-free key binding modified RGHE	FVC 2002 DB1 DB2, DB3, DB4 FVC 2004 DB2	-

Table 2.10: Summary of selected studies under Hybrid methods in chronological order (EER, FAR, FRR values are shown in %)

commitment".

Liu et al. [172] combined multispace random projection for cancelable and heterogeneous fuzzy vault technique for enhancing security of palmprints. Chin et al. [173] built a hybrid system using fingerprint and palmprint features. This method combines random tilling and 2^N discretization scheme. Leng et al. [176] built a hybrid cancelable palmprint cryptosystem by combining palmprint texture code and fuzzy vault. The 2DPalmHash code and fuzzy vault are blendled to protect palmprint templates.

Kumar et al. [174] presents a framework using two functions called parity code and hash code. One cell-array is used to scatter the hash code. Another cell-array is filled with chaff points. Now, parity-code is tangled by a regenerative XorCoding. Jin et al. [175] proposed an “Error Correcting Code (ECC) free key binding scheme along with cancelable transforms for minutiae-based fingerprint biometrics. The minutiae information is secured by a strong non-invertible cancelable transform called modified Randomized Graph based Hamming Embedding (RGHE)”. The summary of selected studies under Hybrid methods is listed in Table 2.10.

2.7 Selected Studies Under Homomorphic Encryption

Ye et al. [35] presented “Anonymous Biometric Access Control (ABAC)” which uses “k-Anonymous Quantization (kAQ) framework”. kAQ uses a lookup table to recognize *k* candidates. HE-based matching protocol is applied on these *k* candidates. Erkin et al. [52] proposed a privacy preserving face recognition system for eigen-faces by using the “Paillier homomorphic encryption scheme”. Later, Sadeghi et al. [53] improve the efficiency of this system. Rane et al. [177] presented Hamming distance calculation for fingerprint applications. Barni et al. [178, 179] demonstrated a distributed biometric system by exploiting “cryptosystems, homomorphic encryption on Fingercode templates in a semi-honest model”. Osadchy et al. [180] proposed a “secure hamming distance based HE for face biometrics. The system is called SCiFI”.

Kulkarni et al. [181] proposed a HE method by calculating values stored on server by performing XOR operation between biometric template vector and corresponding user’s key. Karabat et al. [182] introduced “THRIVE: threshold homomorphic encryption based secure and privacy preserving biometric verification system” that is applicable to any biometric. Barrero et al. [183] presented a “Paillier’s homomorphic probabilistic encryption” on online signature systems. The summary of selected studies under “Homomorphic Encryption” schemes developed in biometrics is listed in Table 2.11.

Article's reference	Biometric trait	Technique	Dataset
Ye et al. (2009) [35]	Iris	k-ABAC kAQ	CASIA
Erkin et al. (2009) [52]	Face	Paillier Homomorphic Encryption	ORL face
Sadeghi et al. (2009) [53]	Face	Oblivious Transfer(OT)	-
Rane et al. (2009) [177]	Fingerprint	Secure Hamming Distance	Proprietary database of 1035 fingers
Barni et al. (2010) [178, 179]	Fingerprint	Fingercode	408 images acquired by a CrossMatch Verifier 300 sensor
Osadchy et al. (2010) [180]	Iris	SciFi	CMU-PIE
Kulkarni et al. (2013) [181]	Iris	Secure Hamming Distance	ICE 2005
Karabat et al. (2015) [182]	Any Biometric	THRIVE	-
Barrero et al. (2016) [183]	Online Signature	Dynamic Time Warping Paillier Cryptosystem	Signature subset from BioSecureID Multi-modal database sensor

Table 2.11: Summary of selected studies under Homomorphic Encryption in chronological order

2.8 Analysis and Discussion of SLR Results

The analysis of the state-of-art literature on BTP schemes is made addressing the research questions discussed in Table 2.1. The percentage distribution of broad BTP categories developed in literature is shown in Figure 2.3. The overall percentage of BTP schemes developed on each biometric trait is shown in Figure 2.4. From Figure 2.3, it is observed that 49% of the published literature is on cancelable biometrics, 35% is on biometric cryptosystems, 8% on hybrid methods, and 8% on HE schemes. Hence, the hybrid methods and methods using HE, which have been paid little attention so far need to be developed to make use of advantages of multiple schemes.

From Figure 2.4, it is observed that BTP schemes developed so far are 44% on fingerprints, 21% on the iris, 12% on the face, 10% on the signature, 5% on multi-modal, 4% on palmprint, 3% on voice, and 1% on Finger vein traits. Hence, we can say multi-modal biometric template protection schemes need to be developed to make use of advantages of multi-biometric systems.

Figures 2.5, 2.6 represents the percentage of BTP schemes developed on each biometric trait under subcategories of cancelable biometrics, *i.e.* salting and non-invertible transforms respectively. Figures 2.7, 2.8, 2.9 represents the percentage of BTP schemes developed on each biometric trait under subcategories of biometric cryptosystems, *i.e.* fuzzy commitment schemes, fuzzy vault schemes, and key generating cryptosystems respectively.

Figures 2.10, 2.11 represents the percentage of BTP schemes developed on each biometric trait under hybrid methods and homomorphic encryption-based methods respectively. After the selection and analysis of studies, we had a look at the affiliation of authors and listed out the active research communities in developing BTP schemes in Table 2.12. The publicly available databases on which BTP schemes are evaluated in the literature are listed in Table 2.13.

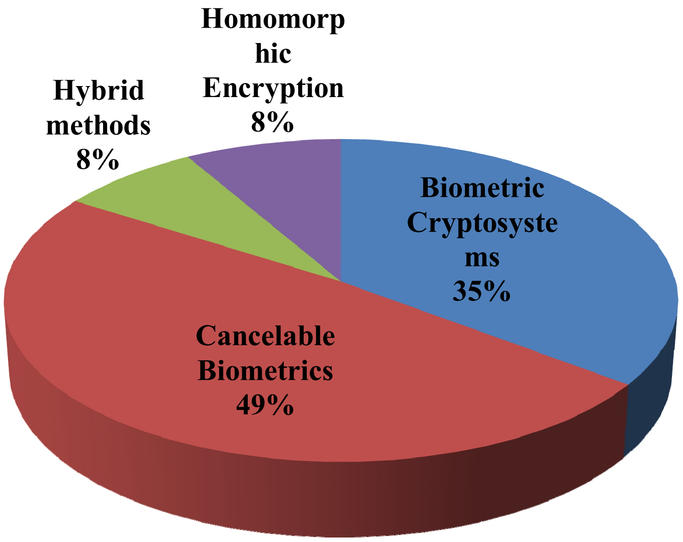


Figure 2.3: Overall percentage of Biometric Template Protection categories in the literature

S.No	Community	Key Studies
1	Biometrics Research Group, Michigan State University, USA `http://biometrics.cse.msu.edu/`	Jain et al. [12] Nagar et al. [50, 144, 149, 184, 185] Nandakumar et al. [22, 143]
2	da/sec Biometrics and Internet Security Research Group, Center for Advanced Security Research Darmstadt (CASED), Germany `https://www.dasec.h-da.de/`	Rathgeb et al. [106, 107, 109, 120, 157]
3	The Multimedia Signal Processing and Security Lab (WaveLab), University of Salzburg, Austria `wavelab.at/member-uhl.shtml`	Rahtgeb et al. [44, 126, 130, 136, 137]
4	IBM Research, Thomas J. Watson Research Center, USA `https://www.research.ibm.com/labs/watson/`	Ratha et al. [28, 81, 82]

5	Multimedia Security Lab, Yonsei University, South Korea
	`https://sites.google.com/site/`
	`multimediasecuritylab/`
		Teoh et al. [37, 63, 64, 69, 122]
		Wong et al. [104]
		Jin et al. [18, 111, 175]

6	Yokohama Research Laboratory, Hitachi Ltd, Japan `http://www.hitachi.com/rd/about/`	Hirata et al. [79] Takahashi et al. [98, 186] Murakami et al. [187]
7	Advanced Cryptosystems Research Group, National Institute of Advanced Industrial Science and Technology (AIST), Japan `https://staff.aist.go.jp/`	Takahashi et al. [186] Murakami et al. [187]
8	La Trobe University,Australia `http://www.latrobe.edu.au/`	Ahmad et al. [97] Wang et al. [99, 116] Yang et al. [105, 162]
9	University of New South Wales at the Australian Defence Force Academy (UNSW@ADFA), Australia `https://www.unsw.adfa.edu.au/`	Ahmad et al. [97] Rang et al. [99, 116] Yang et al. [105, 162] Li et al. [161, 188]
10	Centre for Automation Research, University of Maryland, USA `http://www.cfar.umd.edu/`	Patel et al. [49] Pillai et al. [80, 93]
11	Biometric Systems and Multimedia Forensics LAB, University of “Roma TRE”, Italy `http://biomedia4n6.uniroma3.it/index.html`	Mairorana et al. [87, 88, 89, 123, 128] Nanni et al. [95]
12	Biometrics Systems Laboratory, University of Bologna, Italy `http://biolab.csr.unibo.it/home.asp`	Adler et al. [13] Cappelli et al. [119] Nanni et al. [95] Ferrara et al. [102, 108]

13	CyLab, Biometrics Center,
	Carnegie Mellon University, USA
	`https://www.cylab.cmu.edu/`
	`research/center-biometrics.html`
		Savvides et al. [189]

14	Universiti tunku abdul rahman Kuala Lumpur, Malaysia `www.utar.edu.my`	Jin et al. [18, 111, 175]

Table 2.12: Active Research Communities in designing Biometric Template Protection methods

S.No	Database	Biometric trait	Abbreviation	Source
1	FVC 2000	Fingerprint	Fingerprint Verification Competetion	http://bias.csr.unibo.it/fvc2000/
2	FVC 2002			http://bias.csr.unibo.it/fvc2002/
3	FVC 2004			http://bias.csr.unibo.it/fvc2004/
4	FVC 2006			http://bias.csr.unibo.it/fvc2006/
5	FVC Ongoing			http://bias.csr.unibo.it/fvcongoing/
6	NIST-SD14		National Institute of Standards and Technology- Special Database	http://www.nist.gov/srd/ nistsd14.cfm
6	ICE	Iris	Iris Challege Evaluation	http://www.nist.gov/itl/ iad/ig/ice.cfm
7	IITD iris v1		IIT Dellhi Iris version 1	http://www4.comp.polyu.edu.hk/ csajaykr/IITD/Database_Iris.htm
8	CASIA Iris v1		National Laboratory of Pattern Recognition Institute of Automation, Chinese Academy of Sciences	http://biometrics.idealtest.org/ dbDetailForUser.do?id=1
9	CASIA Iris v3			http://biometrics.idealtest.org/ dbDetailForUser.do?id=3
10	CASIA Iris v4			http://biometrics.idealtest.org/ dbDetailForUser.do?id=4
11	FERET	Face	Facial Recognition Technology	http://www.nist.gov/itl/iad/ ig/colorferet.cfm
12	CALTECH		California Institute of Technology.	http://www.vision.caltech.edu/ html-files/archive.html
13	CMU-PIE		Carnegie Mellon University- Pose, Illumination and Expression	http://vasc.ri.cmu.edu/idb/ html/face/
14	NIR face		Hong Kong Polytechnic University, Near-Infrared	http://www4.comp.polyu.edu.hk/ biometrics/polyudb_face.htm
15	ORL face		AT& T Laboratories, Cambridge	http://www.cl.cam.ac.uk/ research/dtg/attarchive/ facedatabase.html
16	NVIE		Natural Visible and Infrared Facial Expression	http://nvie.ustc.edu.cn/
17	FRGC		Face Recognition Grand Challenge	http://www.nist.gov/ itl/iad/ig/frgc.cfm
17	XM2VTS		Multi Modal Verification for Teleservices and Security applications	http://www.ee.surrey.ac.uk/ CVSSP/xm2vtsdb/
18	AR face		-	http://www2.ece.ohio-state.edu/ aleix/ARdatabase.html
19	Poly U	Palmprint	Hong Kong Polytechnic University Palmprint	http://www4.comp.polyu.edu.hk/ biometrics/ MultispectralPalmprint/MSP.htm
20	SVC 2004	Signature	Signature Verification Competetion	http://www.cse.ust.hk/svc2004/
21	MCYT		(Ministerio de Cienciay Tecnologıa, Spanish Ministry of Science and Technology	Ortega-Garcia et al. [190]
22	TI 46	Voice	Texas Instruments 46- Word Speaker-Dependent Isolated Word Corpus	http://catalog.ldc.upenn.edu/ LDC93S9
23	YOHO speech		YOHO Speaker Verification	https://catalog.ldc.upenn.edu/ LDC93S9

Table 2.13: Publicly available databases on which BTP schemes are evaluated in the literature and their source

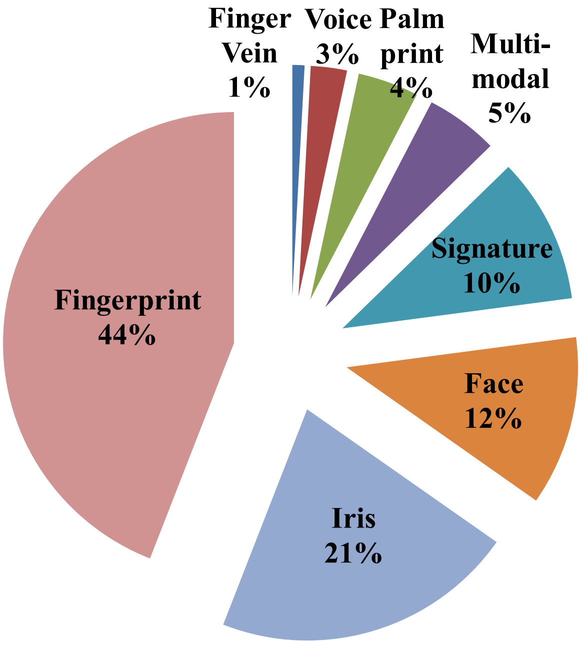


Figure 2.4: Overall percentage of Biometric Template Protection schemes developed on each biometric trait

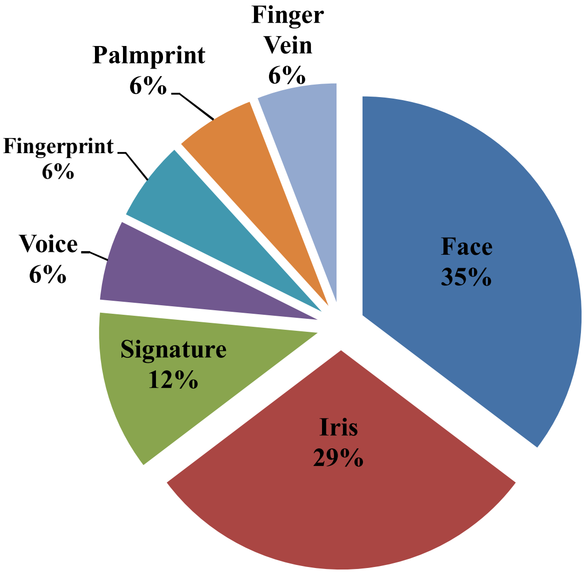


Figure 2.5: Percentage of Salting methods developed on each biometric trait

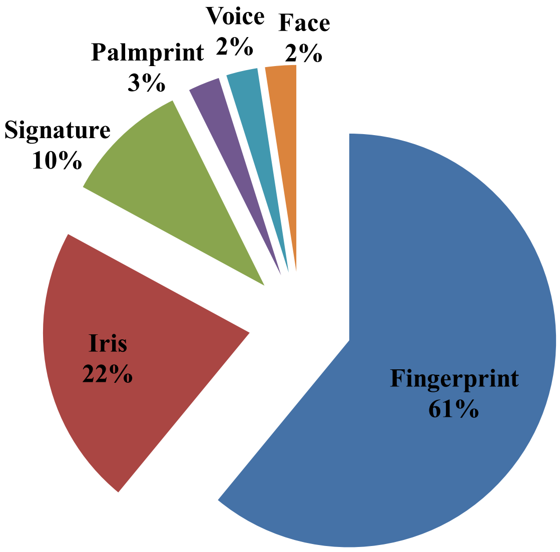


Figure 2.6: Percentage of Non-invertible transforms developed on each biometric trait

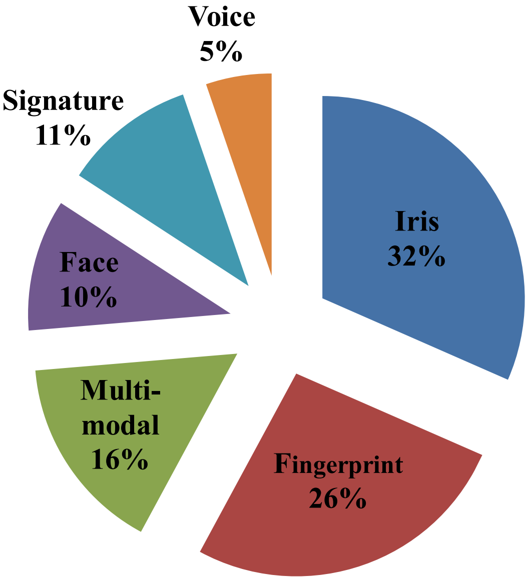


Figure 2.7: Percentage of Fuzzy commitment schemes developed on each biometric trait

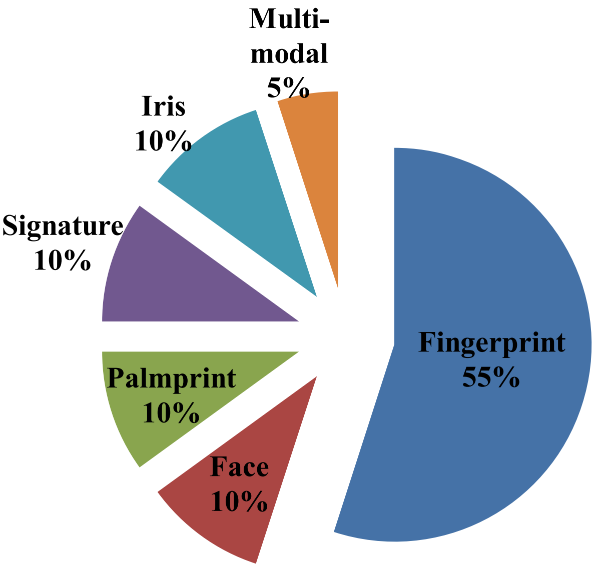


Figure 2.8: Percentage of Fuzzy Vault schemes developed on each biometric trait

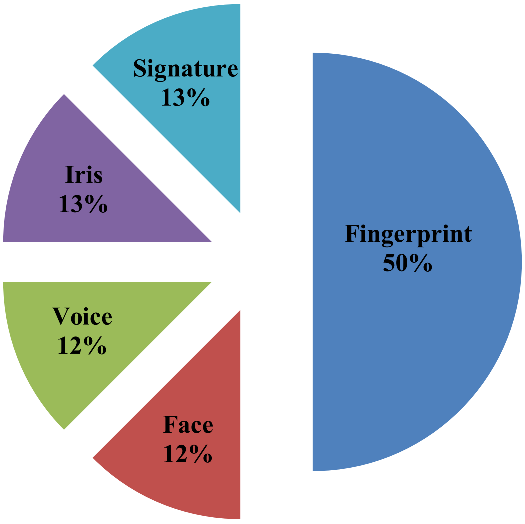


Figure 2.9: Percentage of Key generating systems developed on each biometric trait

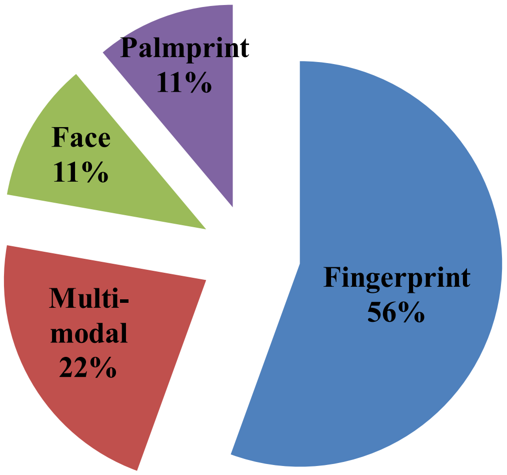


Figure 2.10: Percentage of Hybrid methods developed on each biometric trait

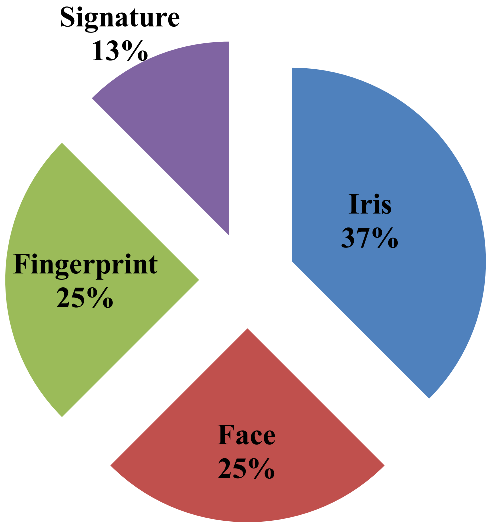


Figure 2.11: Percentage of Homomorphic Encryption based methods developed on each biometric trait

2.9 Justification for present work

Based on the literature survey, there are four categories for protecting biometric templates: Cancelable Biometrics, Bio-Cryptosystems, hybrid methods, and Homomorphic Encryption-based methods. Our contribution in this thesis is devising cancelable biometric techniques, bio-cryptosystems techniques, and hybrid techniques for fingerprint template security without degrading accuracy.

Literature survey reveals that various levels of fusion, viz sensor, feature, score, and decision level fusion can be applied to protect biometric templates. Hence, we proposed feature level and score level fusion techniques for fingerprint template security

It is noticed from SLR that the hybrid methods, which have been paid little attention so far need to be developed to make use of advantages of multiple schemes, viz cancelable biometrics and bio-cryptosystems. Hence, we proposed hybrid methods for protecting fingerprint templates

Chapter 3

Cancelable Fingerprint Template Generation Techniques

In section 1.8, we discussed how a cancelable biometric system is built. Minutiae-based systems for generating cancelable fingerprint templates are classified as "Registration based methods" [81, 99, 113] and "Registration-free (alignment-free) methods" [18, 97]. In the former, the fingerprint image is pre-aligned in accordance with the position of the "singular point" (*i.e.*, core or delta). Hence, these methods require exact detection of the singular points. But in the later, the local features, *i.e.*, minutiae points are considered for transformation. Literature survey reveals that there is a need for devising "alignment-free methods" because "registration-based methods" may fail due to lack of singular points in some fingerprint images. Further, in the existing methods, cancelable fingerprint templates are generated by taking a threshold of the number of minutiae considered in the transformation process, thus resulting in poor performance of the system. This fact motivated us to design alignment-free methods, which include every minutia point of a fingerprint image in the transformation process. The contributions of this chapter are described below:

(1) Proposed an "alignment-free" algorithm for generating cancelable fingerprint templates using "k-Nearest Neighborhood Structure (kNNS)" constructed for all minutiae points.

(2) Proposed an "alignment-free" algorithm for generating cancelable fingerprint templates using "Delaunay triangulation net" of the fingerprint minutiae

points. In this algorithm, we propose two methods namely DTFS_INCIR and DTFS_AVGLO to construct feature set from the Delaunay triangles.

This chapter is organized as follows: Section 3.2 gives details about preprocessing and feature extraction of a fingerprint image. The kNNS algorithm is presented in section 3.3. The DTFS_INCIR and DTFS_AVGLO algorithms are presented in section 3.4. The performance measures of a biometric system are briefly discussed in section 3.5. The benchmark databases on which proposed algorithms are evaluated are mentioned in section 3.6. The experiments conducted and analysis is discussed in section 3.7. The performance comparison of proposed algorithms with existing methods is discussed in section 3.8. The chapter is concluded by presenting a summary in section 3.9.

3.1 Fingerprint Features

We begin with a brief explanation on features of a fingerprint image. Figure 3.1 represents the features of a fingerprint image. The human fingerprint is comprised of various types of ridge patterns, classified according to the Henry system: left loop, right loop, arch, whorl, and tented arch. Minutiae, the discontinuities that interrupt the

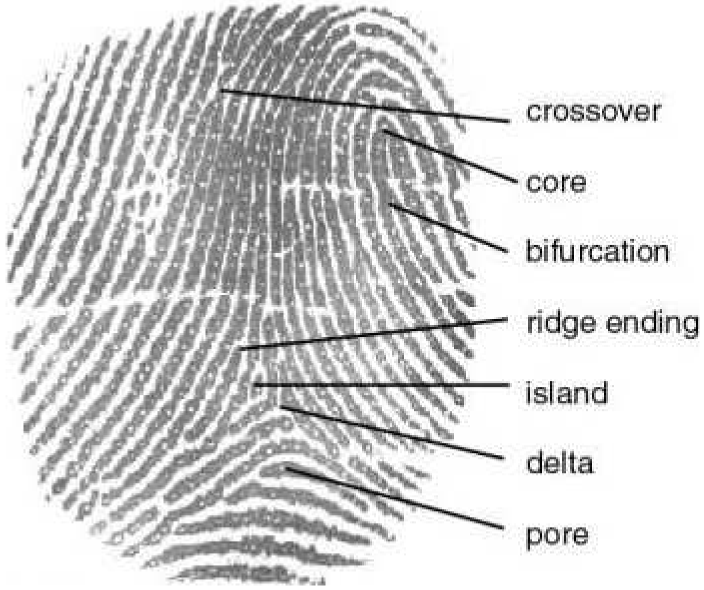


Figure 3.1: Fingerprint image and its features

smooth flow of ridges, are the basis for most fingerprint authentication systems. Minutiae are the ridge endings, the points at which a ridge stops, and bifurcations, the point at which one ridge divides into two. The other features of the fingerprint image are the crossover, core, island, delta, and pore [5].

3.2 Fingerprint image preprocessing and feature extraction

The preprocessing of fingerprint image before feature extraction is important because it can influence the speed and accuracy of the entire system. It includes separation of foreground and background regions, image enhancement, binarization and thinning. After preprocessing, we extract minutiae points from the image, represented as $M_i = (x_i, y_i, \theta_i)_1^t$, where t represent the number of minutiae in the fingerprint image, (x_i, y_i) represent x and y coordinates of minutia point, θ_i represent the orientation of minutia point.

3.3 kNNS algorithm

The flow diagram for the kNNS algorithm is shown in Figure 3.2. The algorithm contains the following steps:

(1) Construction of kNNS for each minutia point in the fingerprint image

(2) Quantize each kNNS and map to 2D array

(3) Generation of bit string

(4) Generation of cancelable template

(5) Matching

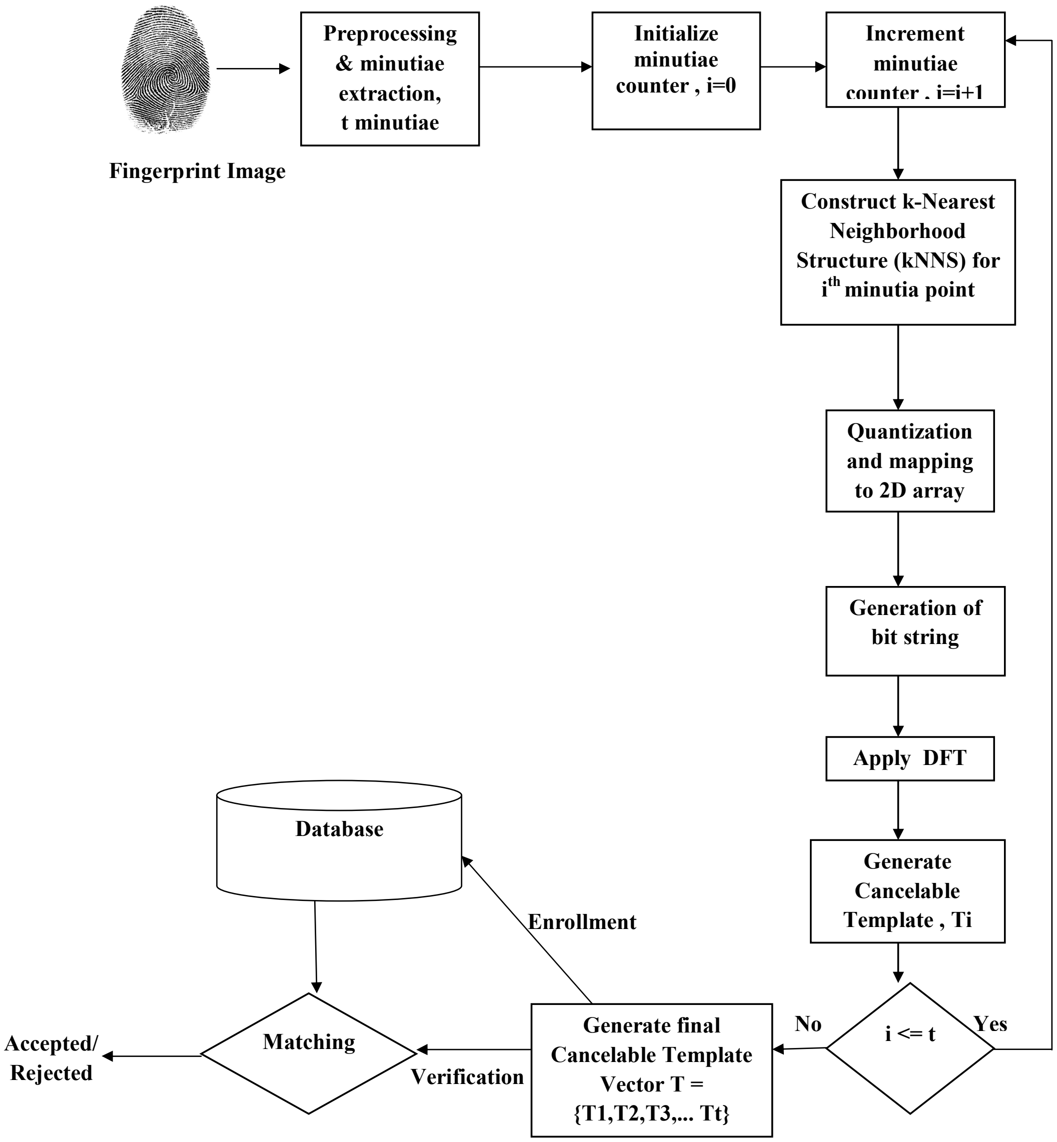

Fingerprint Image
Preprocessing & minutiae extraction, t minutiae
Initialize minutiae counter , i=0
Increment minutiae counter , i=i+1
Construct k-Nearest Neighborhood Structure (kNNS) for i^th minutia point
Quantization and mapping to 2D array
Generation of bit string
Apply DFT
Generate Cancelable Template , Ti
i <= t
Yes
No
Generate final Cancelable Template Vector T = {T1,T2,T3,... Tt}
Enrollment
Database
Verification
Matching
Accepted/ Rejected

Figure 3.2: Flow diagram of kNNS algorithm.

3.3.1 Construction of kNNS for each minutia point

Each minutia point is considered as reference minutia *r* as shown in Figure 3.3 and the $(kNNS_r)$ is computed as follows :

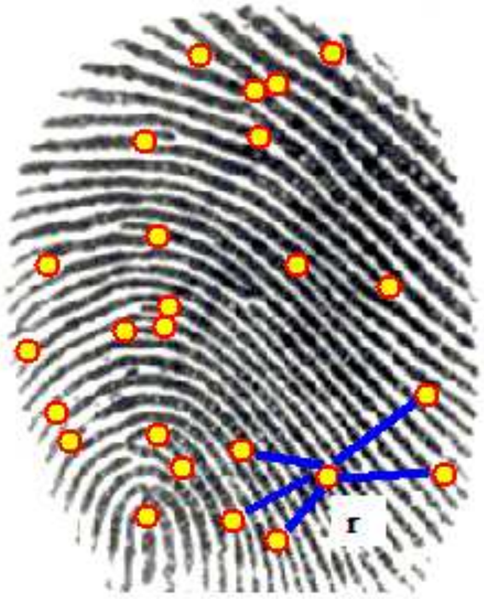


Figure 3.3: kNNS of reference minutia r with neighborhood (k)= 5

(1) Compute the distance from the reference minutia point $r(x_r, y_r, \theta_r)$ to k nearest minutiae points as follows:

$$\begin{aligned} \chi &= (x_j - x_r)cos\theta_r + (y_j - y_r)sin\theta_r \\ \gamma &= (x_j - x_r)sin\theta_r - (y_j - y_r)cos\theta_r \end{aligned} \tag{3.1}$$

$$d_{jr} = \sqrt{\chi^2 + \gamma^2}, j = 1..k \tag{3.2}$$

(2) Compute the difference of orientation between the reference minutia point r and k nearest minutiae points as follows:

$$\theta_{jr} = \theta_j - \theta_r, j = 1...k \tag{3.3}$$

if $\theta_{jr} < 0$, $\theta_{jr} = \theta_{jr} + 360$.

(3) The d_{jr} and θ_{jr} calculated in steps 1 and 2 forms a $kNNS$ for a reference minutia r given by

$$kNNS_r = (d_{1r}, \theta_{1r}), (d_{2r}, \theta_{2r}) \ldots (d_{kr}, \theta_{kr}) \tag{3.4}$$

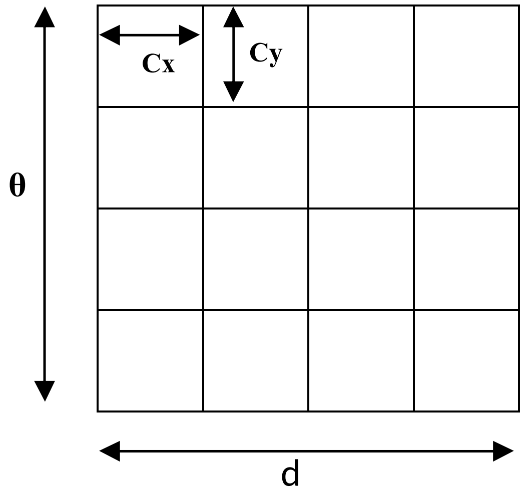


Figure 3.4: Array A_1 with cell size C_x, C_y to which $kNNS$ is mapped

where d_{jr} represents the distance from reference minutia point r to j^{th} nearest neighbor minutia point. θ_{jr} represents the difference of orientation between reference minutia point r and j^{th} minutia point.

(4) Repeat the above steps for all minutiae points in the fingerprint image. Hence, kNNS for t minutiae points in the fingerprint image is represented as:

$$kNNS = (kNNS_1, kNNS_2, ...kNNS_t) \tag{3.5}$$

3.3.2 Quantize each $kNNS_i$ and map to 2D array

A 2D array A_1 divided into cells of size C_x, C_y is defined [94]. Each $kNNS_i, i = 1..t$ is quantized and mapped to A_1 by taking d_{jr} on x-axis ranging in 0 to maximum of all distances computed, *i.e.*, $max(d_{jr})$, θ_{jr} on y-axis ranging in 0 to 2π. The size of the array is $W_x \times W_y$, where $W_x = \lfloor max(d_{jr})/C_x \rfloor$, $W_y = \lfloor 2\pi/C_y \rfloor$. Quantization is important as it produces a fixed length bit string with a variable size $kNNS_i$. Here, W_x and W_y denote the number of cells in A_1 and $\lfloor . \rfloor$ denote the floor function. The length and width of cells in A_1 is shown in Figure 3.4.

3.3.3 Generation of bit string for each $kNNS_i$

Now, we find the cells in A_1 that includes (d_{jr}, θ_{jr}) of $kNNS_i$ as follows:

$$\begin{Bmatrix} x_i \\ y_i \end{Bmatrix} = \begin{Bmatrix} \lfloor d_{jr}/C_x \rfloor \\ \lfloor \theta_{jr}/C_y \rfloor \end{Bmatrix} \tag{3.6}$$

where x_i, y_i represents indices of the array. If one point or more than one point falls in a cell, the value of corresponding cell is set to 1 else 0. This indicates a many-to-one mapping to a cell in A_1. By sequentially visiting the cells in A_1, a bit string (B_s) is generated. The number of bits, say n, in the bit string is the size of the array, *i.e.*, $n = W_x \times W_y$.

3.3.4 Generation of cancelable template

A Discrete Fourier Transform (DFT) is applied to the bit string B_s. Since the size of B_s is n, a n-point DFT is performed as follows:

$$V = \sum_{s=0}^{n-1} B_s e^{-j2\pi is/n}, i = 0, 1, \ldots.n-1 \tag{3.7}$$

Here, V is a complex vector of size $n \times 1$. Now, user chooses a random matrix R of size $p \times q$, where $q = n$, and $p < n$. R is multiplied by V to generate cancelable template T of size $p \times 1$ [99].

$$R \times V = T \tag{3.8}$$

The above procedure is repeated for all $kNNS = (kNNS_1, kNNS_2, \ldots kNNS_t)$. Now, we get t transformed templates, denoted by $T = (T_1, T_2, \ldots T_t)$ for t minutiae points in the fingerprint image. Hence, T is the final cancelable template enrolled into the system. During verification phase, the same transformation is applied using the same random matrix used at enrollment. Hence, the query template will be generated and matching is done in transformed domain [81].

3.3.5 Matching

The matching process is defined as comparing the enrolled and query templates resulting in a score between 0 to 1. 0 indicates a total mismatch and 1 indicates a perfect match. In our method, matching is a two-fold process: local matching and global matching.

3.3.5.1 Local matching

The distance between T_i and ϑ_j can be given by

$$d(T_i, \vartheta_j) = \frac{\left\| T_i - \vartheta_j \right\|_2}{\left\| T_i \right\|_2 + \left\| \vartheta_j \right\|_2} \tag{3.9}$$

where $\left\| . \right\|_2$ denotes the 2-norm. Now the local matching score between enrolled and query template can be given by

$$LMS(T_i, \vartheta_j) = 1 - d(T_i, \vartheta_j) \tag{3.10}$$

Let the enrolled fingerprint template be $T = (T_1, T_2 ... T_e)$ and query fingerprint template be $\vartheta = (\vartheta_1, \vartheta_2, ... \vartheta_q)$, where e and q are number of minutiae in enrolled and query fingerprint images. Figure 3.5 shows how enrolled and query fingerprint templates are matched.

3.3.5.2 Global Matching

The matching in Figure 3.5 results in a similarity matrix as follows:

$$S(T_i, \vartheta_j) = \begin{cases} max(LMS(T_i, \vartheta_j)) & \forall i \varepsilon [1, e], \forall j \varepsilon [1, q] \\ 0 & Otherwise \end{cases} \tag{3.11}$$

From this similarity matrix, the global matching score is generated by employing the method in [103] given by

$$GMS = \frac{\sum_{i=1}^{e} \sum_{j=1}^{q} S(T_i, \vartheta_j)}{\eta} \tag{3.12}$$

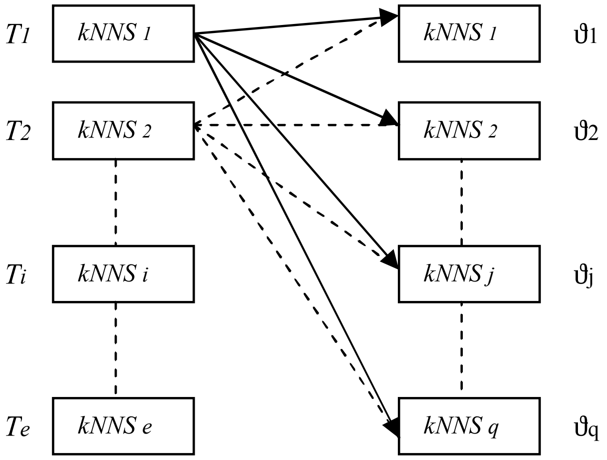


Figure 3.5: Local matching between enrolled and query fingerprint templates in kNNS algorithm

where η denotes the number of non-zero elements in $S(T_i, \vartheta_j)$. Algorithm 3.1 describes the step by step procedure of kNNS algorithm.

Algorithm 3.1 kNNS($x_i, y_i, \theta_i, k, A_1, C_X, C_Y, R$): To compute k-Nearest Neighborhood Structure

```
Input:  Minutiae locations $(x_i, y_i)$
          Orientation of minutiae points $\theta_i$
          Number of Neighbor points considered $k$
          Predefined 2D Array $A_1$ divided into cells of size $C_X, C_Y$
          Random matrix $R$
Output: Normalized match score

begin
Intialize minutia counter $i \leftarrow 0$;
$t \leftarrow$ Number of minutiae points in a fingerprint image;
while $i \leq t$ do
//Consider each minutia point as reference minutia.
        $r \leftarrow i$ ;
        Intialize counter $j \leftarrow 1$;
        while $j \leq k$ do
//Calculate distance from reference minutia r to $j^{th}$ nearest neighbor.
```

Algorithm 3.1 kNNS algorithm (continued)

```
8:              $\chi = (x_j - x_r)cos\theta_r + (y_j - y_r)sin\theta_r$;
9:              $\gamma = (x_j - x_r)sin\theta_r - (y_j - y_r)cos\theta_r$;
10:             $d_{jr} = \sqrt{\chi^2 + \gamma^2}$;
    //Calculate difference of orientation between reference minutiae r and $j^{th}$ nearest neighbor.
11:             $\theta_{jr} = \theta_j - \theta_r$;
12:             if ($\theta_{jr} < 0$) then
13:                 $\theta_{jr} = \theta_{jr} + 360$;
14:             end if;
15:             $kNNS(j,:) \leftarrow (d_{jr}, \theta_{jr})$;
16:             $j \leftarrow j + 1$;
17:         end while
    //Refer to section 3.3.2.
18:         Quantize kNNS by cell sizes $C_X, C_Y$, map to 2D array $A_1$
19:         Generate bit string $B_s$                    //Refer to section 3.3.3.
    // Generation of Cancelable template-Refer to section 3.3.4
20:         $V \leftarrow DFT(B_s)$
21:         $T(i) \leftarrow R \times V$
22:         $i \leftarrow i + 1$;
23: end while
    //Refer to section 3.3.5
24: Match score generation
25: return match_score
26: end
```

Computational Complexity of kNNS algorithm

Steps 7 to 17 repeated k times: $O(k)$

Steps 4 to 23 repeated t times: $O(t)$

Algorithm Complexity: $O(k * t)$

3.4 DTFS_INCIR and DTFS_AVGLO Algorithms

The generic framework of "Delaunay Triangle Feature Set (DTFS)" algorithms shown in Figure 3.6 contains the following steps:

(1) Constructing "Delaunay triangulation net" from fingerprint minutiae.

(2) Computation of "Delaunay Triangle Feature Set (DTFS)" using two methods.

> **Method-1:** Feature Set computation using the incircle center of the Delaunay triangle (DTFS_INCIR)
>
> **Method-2:** Feature Set computation using the average of edge lengths at each vertex in the Delaunay triangle (DTFS_AVGLO)

(3) Quantize DTFS, map to a 3D array.

(4) Generation of the bit string.

(5) Generation of the cancelable template.

(6) Matching.

The two methods DTFS_INCIR and DTFS_INCIR, differ in step 2, *i.e.*, feature set computation and the remaining steps are same for the generation of the cancelable template.

3.4.1 Constructing Delaunay triangulation net from fingerprint minutiae

From the minutiae points extracted, a Voronoi diagram is generated which divides the entire image region into several small partitions. In each cell, a minutia point m_i is located at the center. All other points in the cell are closer to m_i rather than any other minutia point in other cells. By connecting all m_i of Voronoi diagram cells the Delaunay triangulation net is formed [191], which contains N Delaunay triangles. The motivation for using Delaunay triangulation net is that it has some desirable local and global features. First, even if the elastic distortion occurs in the fingerprint image,

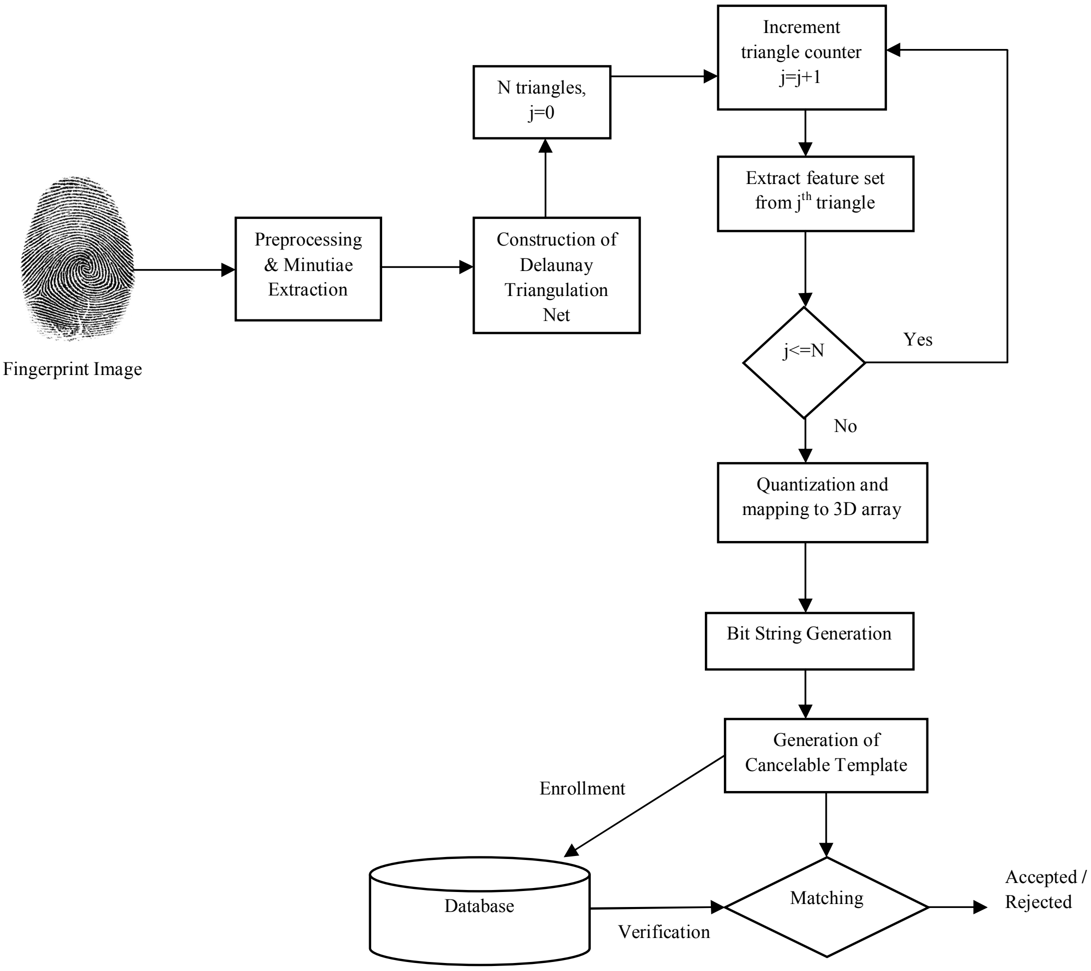

Fingerprint Image
Preprocessing & Minutiae Extraction
Construction of Delaunay Triangulation Net
N triangles, j=0
Increment triangle counter j=j+1
Extract feature set from jth triangle
j<=N
Yes
No
Quantization and mapping to 3D array
Bit String Generation
Generation of Cancelable Template
Enrollment
Database
Verification
Matching
Accepted / Rejected

Figure 3.6: Flow diagram for DTFS_INCIR and DTFS_AVGLO algorithms

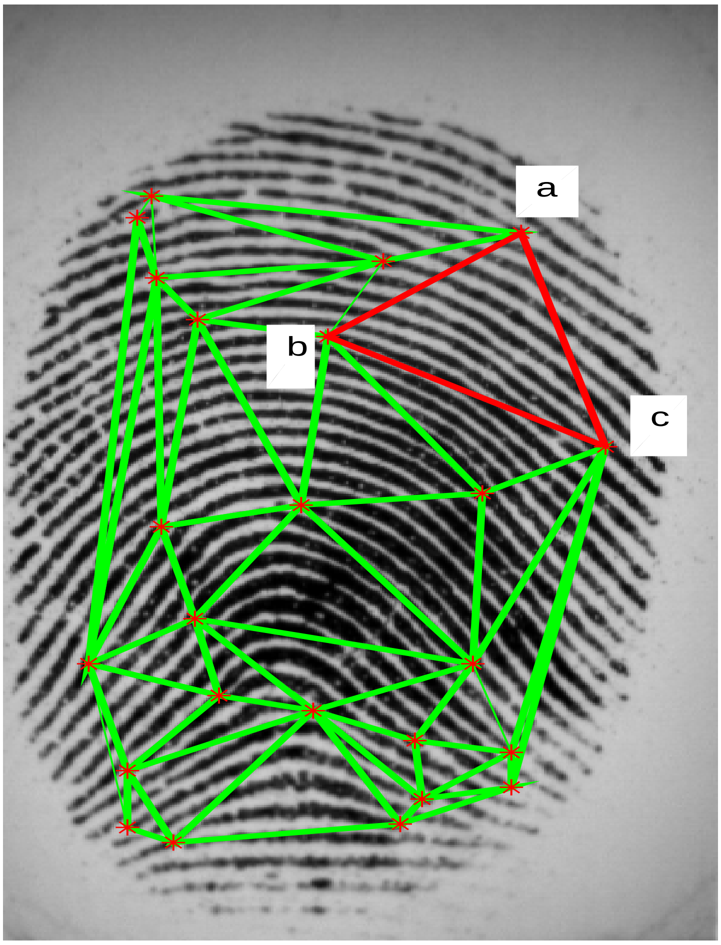


Figure 3.7: "Delaunay triangulation net" constructed from minutiae points of a fingerprint image.

every minutia keeps the same neighborhood structure as long as the minutia will not move to out of tolerance region under distortion. Second, insertion of spurious minutia or deletion of minutia effects only those triangles that contain the minutia [192]. An example of Delaunay triangulation formed from a fingerprint image is shown in the Figure 3.7. As shown in the Figure 3.7, if a minutia point b is missed, only those triangles that contain b will be affected.

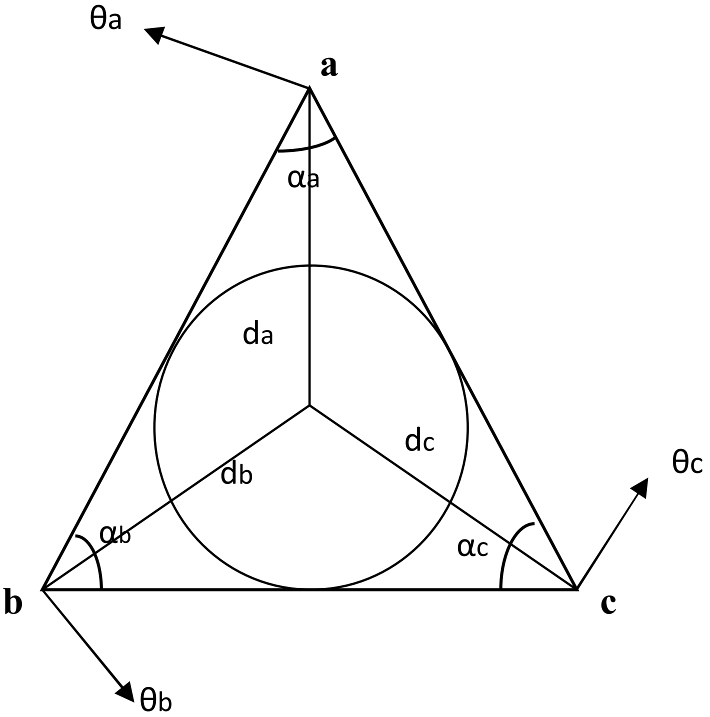


Figure 3.8: A triangle ΔT considered for computation of feature set in DTFS_INCIR

3.4.2 Computation of feature set from Delaunay triangles

Let ΔT be a triangle in the "Delaunay triangulation net". The feature set DTFS is computed as follows:

3.4.2.1 Method-1 : DTFS_INCIR

(1) Consider a triangle Δabc in the "Delaunay triangulation net" as shown in Figure 3.8. The distance from each vertex of triangle to the incircle center of the triangle (d_a, d_b, d_c), the orientation of vertices $(\theta_a, \theta_b, \theta_c)$ which are the original orientation of minutiae points and the internal angles of the triangle at each vertex $(\alpha_a, \alpha_b, \alpha_c)$ are considered for the computation of the feature set.

(2) Let the incircle center be (x_{in}, y_{in}). Then d_a, d_b, d_c can be computed as follows:

$$\begin{aligned} d_a &= \sqrt{(x_{in} - x_a)^2 + (y_{in} - y_a)^2}; \\ d_b &= \sqrt{(x_{in} - x_b)^2 + (y_{in} - y_b)^2}; \\ d_c &= \sqrt{(x_{in} - x_c)^2 + (y_{in} - y_c)^2}; \end{aligned} \tag{3.13}$$

(3) For one triangle $DTFS_1 = \{d_a, \theta_a, \alpha_a, d_b, \theta_b, \alpha_b, d_c, \theta_c, \alpha_c\}$

(4) For N Delaunay triangles, we compute the feature sets, hence the total feature set can be given by $DTFS = (DTFS_1, DTFS_2 .....DTFS_N)$, which consists of $N \times 9$ features.

3.4.2.2 Method-2 : DTFS_AVGLO

(1) Consider a triangle Δabc in Delaunay triangulation net as shown in Figure 3.9. The average length of edges connecting a vertex (d_a, d_b, d_c), the average of orientation difference between the vertex and remaining vertices in the triangle $(\theta_a, \theta_b, \theta_c)$, the internal angles of the triangle at each vertex $(\alpha_a, \alpha_b, \alpha_c)$ are considered for computation of feature set.

(2) From the triangle Δabc in Figure 3.9, (d_a, d_b, d_c) and $(\theta_a, \theta_b, \theta_c)$ are calculated using equation 3.14

$$\begin{aligned} d_a &= avg(d_1, d_3), \theta_a = avg(|\theta_1 - \theta_2|, |\theta_1 - \theta_3|) \\ d_b &= avg(d_1, d_2), \theta_a = avg(|\theta_2 - \theta_1|, |\theta_2 - \theta_3|) \\ d_c &= avg(d_2, d_3), \theta_c = avg(|\theta_3 - \theta_1|, |\theta_3 - \theta_2|) \end{aligned} \tag{3.14}$$

where d_1, d_2, d_3 are lengths of edges of triangle Δabc computed as follows:

$$\begin{aligned} d_1 &= \sqrt{(x_a - x_b)^2 + (y_a - y_b)^2} \\ d_2 &= \sqrt{(x_c - x_a)^2 + (y_c - y_b)^2} \\ d_3 &= \sqrt{(x_b - x_c)^2 + (y_b - y_c)^2} \end{aligned} \tag{3.15}$$

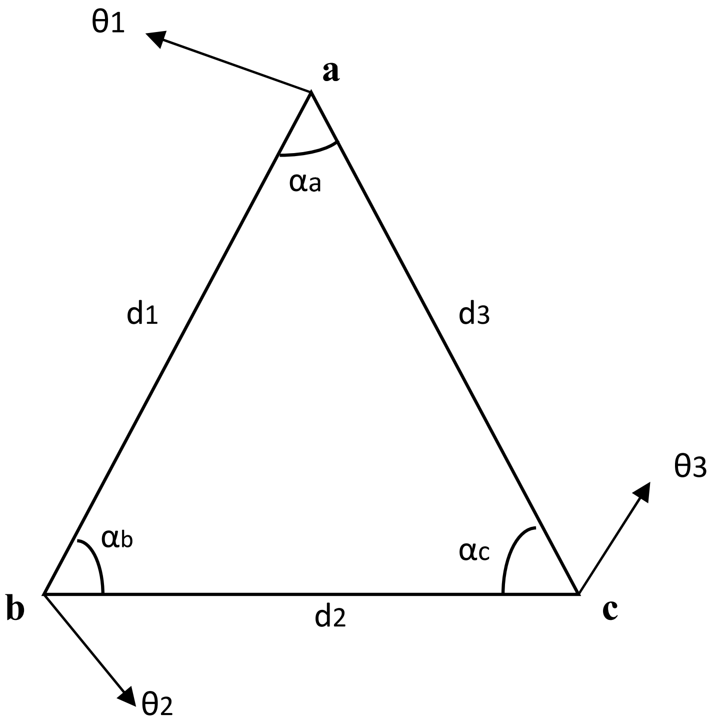


Figure 3.9: A triangle ΔT considered for computation of feature set in DTFS_AVGLO

(3) For one triangle $DTFS_1 = \{d_a, \theta_a, \alpha_a, d_b, \theta_b, \alpha_b, d_c, \theta_c, \alpha_c\}$

(4) For N Delaunay triangles we compute feature sets, hence the total feature set can be given by $DTFS = (DTFS_1, DTFS_2.....DTFS_N)$, which consists of $N \times 9$ features.

3.4.3 Quantize DTFS, and map to a 3D array

A 3D array A_1 divided into cells of size C_X, C_Y,C_Z is defined [94]. Now, quantize DTFS and map to A_1 by taking d_i on x-axis that ranges from 0 to maximum of distances computed, say $max(d_i)$, θ_i on y-axis that ranges from 0 to 2π, α_i on z-axis that ranges from 0 to π. The size of A_1 is $W_X \times W_Y \times W_Z$, where $W_x = \lfloor max(d_i)/C_X \rfloor$, $W_y = \lfloor 2\pi/C_Y \rfloor$ and $W_z = \lfloor \pi/C_Z \rfloor$. W_X, W_Y, W_Z denotes the number of cells in A_1, $\lfloor . \rfloor$ denote the floor function. The length, width and height of cells of A_1 is shown in Figure 3.10.

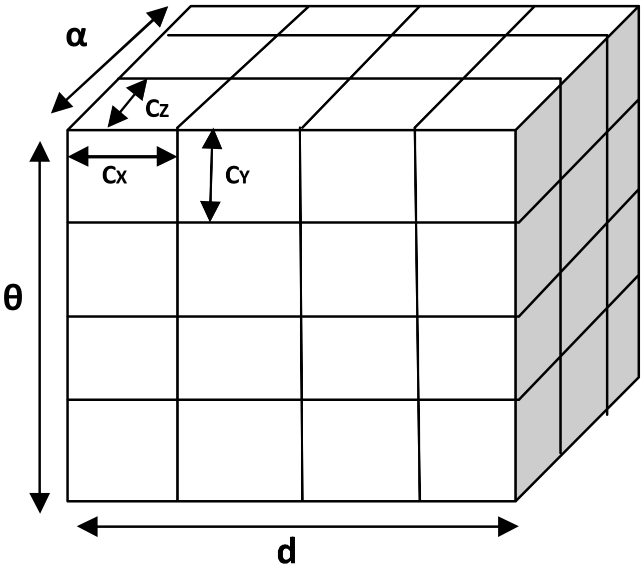


Figure 3.10: A_1 with cell sizes C_x, C_y, C_z where DTFS is mapped

3.4.4 Generation of bit string

The advantage of mapping DTFS to a 3D array is, whatever may be the size of DTFS, a fixed length bit string is produced. Now, we find the cells in A_1 that includes $(d_i, \theta_i, \alpha_i)$ as given below :

$$\begin{Bmatrix} x_i \\ y_i \\ z_i \end{Bmatrix} = \begin{Bmatrix} \lfloor d_i/c_X \rfloor \\ \lfloor \theta_i/c_Y \rfloor \\ \lfloor \alpha_i/c_Z \rfloor \end{Bmatrix} \tag{3.16}$$

where x_i, y_i, z_i represents indices of the 3D array. If one value or more than one value falls in a cell, the value of cell is set to 1 else 0. This indicates many-to-one mapping to a cell in the 3D array which ensures non-invertibility. Now, by sequentially visiting all the cells in array, a bit string (B_s) is generated. The number of bits, say n, in the bit string is the size of array, *i.e.*, $n = W_X \times W_Y \times W_Z$.

3.4.5 Generation of cancelable template

To protect bit string B_s we apply a two-fold process. First, a DFT is applied to the bit string to generate a complex vector V. Since the size of B_s is n, we perform a n-point

DFT as follows:

$$V = \sum_{s=0}^{n-1} B_s e^{-j2\pi is/n}, i = 0, 1, ...n-1 \tag{3.17}$$

By this we got a complex vector V of size $n \times 1$. Second, the complex vector V is secured by applying a non-invertible transformation to V as in [99]. A user-specific random matrix (A) of dimension $p \times q$ is chosen, where $q = n$. Now this A is multiplied by V to generate the final cancelable template T of size $p \times 1$ as follows:

$$A \times V = T \tag{3.18}$$

This template T is enrolled into the system. In equation 3.16 there are large number of solutions for V by considering the system of linear equations [99]. Thus the irreversibility requirement of template protection is quenched. During verification, the same transformation is applied using the same random matrix used at enrollment phase. Thus the query template is generated.

3.4.6 Matching

The distance between enrolled template T_i and query template T_j can be given by

$$d(T_i, T_j) = \frac{\left\| T_i - T_j \right\|_2}{\left\| T_i \right\|_2 + \left\| T_j \right\|_2} \tag{3.19}$$

where $\left\| . \right\|_2$ denotes the 2-norm. From this, the matching score $S(T_i, T_j)$ between the enrolled and query template is given by

$$S(T_i, T_j) = 1 - d(T_i, T_j) \tag{3.20}$$

The matching score 1 indicates that the match is perfect. 0 indicates a total mismatch between enrolled and query templates. Algorithm 3.2 describes DTFS_INCIR method and Algorithm 3.3 describes DTFS_AVGLO method.

Algorithm 3.2 DTFS_INCIR($x_i, y_i, \theta_i, A_1, C_X, C_Y, C_Z, R$): To compute Delaunay Triangle Feature Set using incircle center of each triangle

```
Input:  Minutiae locations $(x_i, y_i)$
          Orientation of minutiae points $\theta_i$
          Predefined 3D Array $A_1$ divided into cells of size $C_X, C_Y, C_Z$
          Random matrix $R$
Output: Normalized match score

begin                                                        // Refer to Figure 3.7
Construct Delaunay triangulation net from the fingerprint minutia;
$N \leftarrow$ Number of triangles in Delaunay triangulation net;
Initialize triangle counter $j \leftarrow 0$;
while $j \leq N$ do                                          //Refer to Figure 3.8
// Compute incircle center of each triangle
    $(x_{in}, y_{in}) \leftarrow$ incircle center of triangle $j$;
// Compute distance from vertices of the triangle to the incircle center
    $d_a \leftarrow \sqrt{(x_{in} - x_a)^2 + (y_{in} - y_a)^2}$;
    $d_b \leftarrow \sqrt{(x_{in} - x_b)^2 + (y_{in} - y_b)^2}$;
    $d_c \leftarrow \sqrt{(x_{in} - x_c)^2 + (y_{in} - y_c)^2}$;
    $\theta_a \leftarrow$ orientation of minutia point a;
    $\theta_b \leftarrow$ orientation of minutia point b;
    $\theta_c \leftarrow$ orientation of minutia point c;
// Compute the internal angles of each triangle
    $\alpha_a \leftarrow \cos^{-1}(\frac{d_b^2+d_c^2-d_a^2}{2*d_b*d_c})$, $\alpha_b \leftarrow \cos^{-1}(\frac{d_a^2+d_c^2-d_b^2}{2*d_a*d_c})$, $\alpha_c \leftarrow \cos^{-1}(\frac{d_a^2+d_3^b-d_2^c}{2*d_a*d_b})$
    $DTFS(j,:) \leftarrow (d_a, \theta_a, \alpha_a, d_b, \theta_b, \alpha_b, d_c, \theta_c, \alpha_c)$;
    $j \leftarrow j+1$;
end while
//Refer to section 3.4.3
Quantize DTFS by cell sizes $C_X, C_Y, C_Z$, map to 3D array $A_1$
Generate bit string;                                         //Refer to section 3.4.4
//Generation of Cancelable template-Refer to section 3.4.5
$V \leftarrow DFT(B_s)$
$T(i) \leftarrow R \times V$
```

Algorithm 3.2 DTFS_INCIR (continued)

```
19: Match score generation;                                   //Refer to section 3.4.6
20: return match_score;
21: end
```

Computational Complexity of DTFS_INCIR algorithm

Delaunay triangulation formation from t minutiae points: $O(t)$

Steps 5 to 14 repeated N times: $O(N)$

Algorithm Complexity: $O(t+N)$

Algorithm 3.3 DTFS_AVGLO(x_i,y_i, θ_i,A_1,C_X,C_Y,C_Z,R): To compute Delaunay Triangle Feature Set using average of edge lengths and orientation difference at each vertex

Input: Minutiae locations (x_i, y_i)
Orientation of minutiae points θ_i
Predefined 3D Array A_1 divided into cells of size C_X, C_Y, C_Z
Random matrix R

Output: Normalized match score

```
1: begin
   // Refer to Figure 3.7
2: Construct Delaunay triangulation net from the fingerprint minutia;
3: $N \leftarrow$ Number of triangles in Delaunay triangulation net;
4: Initialize triangle counter $j \leftarrow 0$;
5: while $j \leq N$ do                                        //Refer to Figure 3.9
   //Computation of edge lengths
6:         $d_1 \leftarrow \sqrt{(x_a - x_b)^2 + (y_a - y_b)^2}$;
7:         $d_2 \leftarrow \sqrt{(x_c - x_a)^2 + (y_c - y_b)^2}$;
8:         $d_3 \leftarrow \sqrt{(x_b - x_c)^2 + (y_b - y_c)^2}$;
```

Algorithm 3.3 DTFS_AVGLO (continued)

```
    //Compute average of edge lengths, orientation difference, internal angle at each vertex
9:          $d_a \leftarrow avg(d_1,d_3), \theta_a \leftarrow avg(|\theta_1 - \theta_2|, |\theta_1 - \theta_3|), \alpha_a \leftarrow \cos^{-1}(\frac{d_1^2+d_3^2-d_2^2}{2*d_1*d_3})$;
10:         $d_b \leftarrow avg(d_1,d_2), \theta_a \leftarrow avg(|\theta_2 - \theta_1|, |\theta_2 - \theta_3|), \alpha_b \leftarrow \cos^{-1}(\frac{d_1^2+d_2^2-d_3^2}{2*d_1*d_2})$;
11:         $d_c \leftarrow avg(d_2,d_3), \theta_c \leftarrow avg(|\theta_3 - \theta_1|, |\theta_3 - \theta_2|), \alpha_c \leftarrow \cos^{-1}(\frac{d_2^2+d_3^2-d_1^2}{2*d_2*d_3})$;
12:         $DTFS(j,:) \leftarrow (d_a, \theta_a, \alpha_a, d_b, \theta_b, \alpha_b, d_c, \theta_c, \alpha_c)$;
13:         $j \leftarrow j+1$;
14: end while
    //Refer to section 3.4.3
15: Quantize DTFS by cell sizes $C_X, C_X, C_Z$, map to 3D array $A_1$
16: Generate bit string $B_s$;                                  //Refer to section 3.4.4
    // Generation of Cancelable template-Refer to section 3.4.5
17: $V \leftarrow DFT(B_s)$
18: $T(i) \leftarrow R \times V$
19: Match score generation;                                     //Refer to section 3.4.6
20: return match_score;
21: end
```

Computational Complexity of DTFS_AVGLO algorithm

Delaunay triangulation formation from t minutiae points: $O(t)$

Steps 5 to 14 repeated N times: $O(N)$

Algorithm Complexity: $O(t+N)$

3.5 Performance measures of a biometric system

A similarity match score is known as a genuine or authentic score if it is a result of matching two samples of the same biometric trait of a user. A similarity match score is known as an imposter score if it involves comparing two biometric samples originating from different users [8]. Let $\rho(s|\omega_1)$, $\rho(s|\omega_0)$ be the probability density functions of the genuine and impostor scores, respectively.

The performance of a biometric system can be evaluated using the following measures [193]:

- The **False Accept Rate (FAR)** (or, the False Match Rate (FMR)) of a biometric system is defined as the fraction of impostor scores that are greater than or equal to the threshold η.

$$FAR(\eta) = \rho(s \geq \eta|\omega_0) = \int_{\eta}^{\infty} \rho(s|\omega_0)ds \tag{3.21}$$

- The **False Reject Rate (FRR)** (or, the False Non-Match Rate (FNMR)) of a biometric system is defined as the fraction of genuine scores that are less than the threshold η.

$$FRR(\eta) = \rho(s < \eta|\omega_1) = \int_{-\infty}^{\eta} \rho(s|\omega_1)ds \tag{3.22}$$

- The **Genuine Accept Rate (GAR)** is the fraction of genuine scores that are greater than or equal to the threshold η. Therefore, GAR=1-FRR.

$$GAR(\eta) = \rho(s \geq \eta|\omega_1) = 1 - FRR(\eta) \tag{3.23}$$

- When a linear, logarithmic or semi-logarithmic scale is used to plot the error rates, then the resulting graph is known as a **Receiver Operating Characteristic (ROC) curve**. In many instances, the ROC curve plots the GAR (rather than the FRR) against the FAR.

The performance of a biometric system may also be summarized using the following single-valued measures

- **Equal Error Rate (EER)** refers to that point where the FAR equals the FRR. A lower EER value indicates better performance [8].

- The **d-prime value** (d') measures the separation between the means of the genuine and impostor probability distributions in standard deviation units given by

$$d' = \frac{\left|\mu_{genuine} - \mu_{imposter}\right|}{\sqrt{(\sigma^2_{genuine} + \sigma^2_{imposter})/2}} \tag{3.24}$$

  where μ's and , σ's denote means and standard deviation of genuine and imposter distributions. A higher d' value indicates better performance [8].

3.6 Benchmark Databases

The following databases are used to test the performance of the system. Each database contain 800 images of 100 fingers, each finger with 8 samples.

- FVC 2002 DB1, DB2, DB3 [194]
- FVC 2004 DB1, DB2, DB3 [195]

We used Neurotechnology Verifinger SDK [196] for extracting the minutiae points from the fingerprint images in the databases.

3.7 Experimental Results and Discussion

3.7.1 Experiment Setup

For conducting experiments, we considered 2 samples of each finger. The proposed algorithms are tested in two scenarios:

(1) **Same key scenario:** Assume a case where a user has lost his key, then an attacker gains access to the system claiming that he is the genuine user. This is called a lost key or stolen key attack. To model this, we used same key (*i.e.*, same random matrix) to all users in equation 3.8 for kNNS algorithm and equation 3.18 for DTFS_INCIR and DTFS_AVGLO algorithms

(2) **Different key scenario:** We use different random matrix in equations 3.8 and 3.18 for different users.

Neighborhood parameter and cell size			**FVC 2002**			**FVC 2004**		
k	C_x	C_y	**DB1**	**DB2**	**DB3**	**DB1**	**DB2**	**DB3**
5	5	9	5.06	5.01	11.66	20.89	21.31	27.64
10	10	10	5.93	5.69	11.77	19.85	20.98	25.76
12	10	18	4.01	3.99	10.95	15.91	14.65	18.96
12	15	18	3.63	4.37	12.39	18.53	17.75	20.08
12	20	18	4.59	5.62	12.79	19.34	18.95	21.71
14	**15**	**15**	**4.71**	**3.44**	**8.79**	**15.36**	**14.25**	**18.27**
14	15	18	4.35	3.90	12.81	16.42	15.71	19.83

Table 3.1: EER obtained for kNNS algorithm for FVC 2002, 2004 databases in the same key scenario

3.7.2 Accuracy

The kNNS algorithm is tested by tuning the parameters like neighborhood parameter k, cell sizes C_x and C_y in the quantization step. Table 3.1 refers to the EER obtained for the kNNS algorithm in the “same key scenario” for “FVC 2002 and 2004 databases”. It is observed that, for $k = 14, C_x = 15, C_y = 15$, kNNS algorithm produces optimal result.

Figure 3.11 represents the “ROC curves” of kNNS algorithm for “FVC 2002 databases”. It is noticed that the recognition rate for “FVC 2002 DB2” is better than “FVC 2002 DB1, and DB3”. Figure 3.12 represents the FAR/FRR, and EER obtained for kNNS algorithm for “FVC 2002 databases”. In case of different key scenario, kNNS algorithm yields 0%, 0%, 3.65%, 3.67%, 4.92%, 6.73% for “FVC 2002 DB1, DB2, DB3, FVC 2004 DB1, DB2, DB3” respectively for the optimal cell sizes $C_x = 15, C_y = 15$ and $k = 14$.

DTFS_INCIR and DTFS_AVGLO algorithms are tested by tuning cell sizes C_X, C_Y, C_Z. Tables 3.2 and 3.3 shows EER obtained for the algorithms. From the tables, it is noticed that for cell sizes $C_X = 20, C_Y = 20, C_Z = 18$ DTFS_INCIR algorithm gives optimal result. For cell sizes $C_X = 10, C_Y = 10, C_Z = 18$ DTFS_AVGLO algorithm gives optimal result. Figure 3.14 and Figure 3.15 represents the “ROC curves” of the DTFS_INCIR and DTFS_AVGLO algorithms respectively. It is noticed that the recognition rate of the algorithms is better for “FVC 2002 DB2” than “FVC 2002 DB1,

Cell Size			FVC 2002			FVC 2004		
C_X	C_Y	C_Z	**DB1**	**DB2**	**DB3**	**DB1**	**DB2**	**DB3**
10	10	18	6.39	5.15	9.3	12.8	13.41	14.58
10	10	20	5.78	4.79	8.57	12.35	13.01	14.38
15	15	18	6.88	4.41	8.63	12.26	11.37	14.62
10	10	30	5.93	5.02	8.03	12.53	12.89	14.63
20	20	30	6.67	4.82	7.68	13.52	15.55	15.87
20	**20**	**18**	**6.25**	**3.84**	**8.16**	**13.06**	**12.95**	**15.23**

Table 3.2: EER obtained for DTFS_INCIR algorithm for FVC 2002, 2004 databases in the same key scenario

Cell Size			FVC 2002			FVC 2004		
C_X	C_Y	C_Z	**DB1**	**DB2**	**DB3**	**DB1**	**DB2**	**DB3**
5	5	18	5.45	4.63	8.78	13.56	13.02	17.27
10	**10**	**18**	**3.96**	**2.98**	**6.89**	**12.17**	**13.29**	**17.73**
10	10	20	4.68	3.58	7.42	12.55	13.07	15.6
15	15	18	5.46	3.32	6.39	12.38	11.77	15.73
10	10	30	4.24	3.86	6.2	12.66	12.4	15.68
20	20	30	7.08	5.08	7.27	12.77	12.24	17.1

Table 3.3: EER obtained for DTFS_AVGLO algorithm for FVC 2002, 2004 databases in same key scenario.

and DB3”. Figure 3.16 represents the FAR/FRR, and EER obtained for DTFS_INCIR and DTFS_AVGLO algorithms for “FVC 2002 DB2” database.

The genuine and imposter score distribution can be plotted to show their separation. Figure 3.13 refers to the score distribution of kNNS algorithm on “FVC 2002 databases”. Figure 3.17 represents the score distribution of DTFS_INCIR and DTFS_AVGLO algorithms on “FVC 2002 DB1” database. Table 3.4 refers to the d' values obtained for the proposed algorithms on FVC 2002 databases. From Table 3.4, it can be observed that the d' values are inversely proportional to the EER values of the proposed algorithms. We further used another factor, the “Kolmogorov-Smirnov (KS) test” to show the separability between genuine and imposter scores. The more closer this test results to 1, the better genuine and imposter scores are separated. Table 3.5 depicts the “KS test values” obtained for the proposed algorithms on “FVC 2002 databases”. It is observed that “KS test values” are closer to 1, which in turn indicates

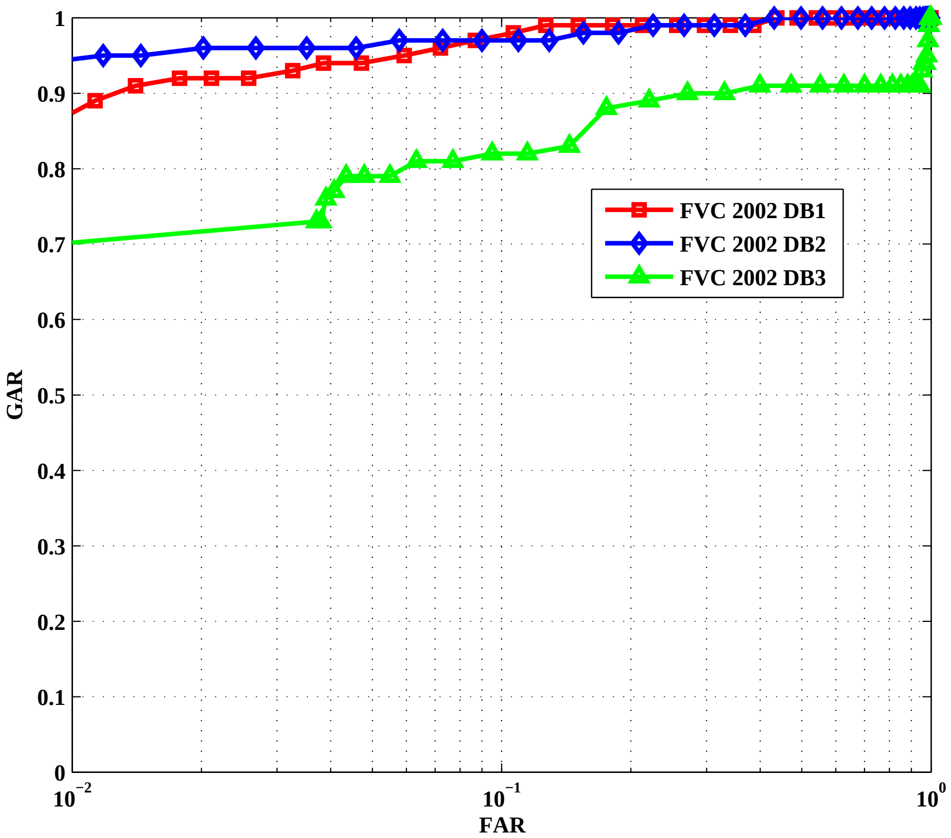


Figure 3.11: ROC curves for kNNS algorithm in same key scenario for FVC 2002 databases

better separation between the “genuine and imposter scores”.

3.7.3 Revocability

The revocability of proposed algorithms can be tested by plotting pseudo-imposter distribution. The pseudo-imposter scores are obtained by generating 100 transformed templates using the same fingerprint that are different from one another. This shows the non-linkability among transformed templates. Figure 3.18 plots the pseudo-imposter distribution of kNNS algorithm on “FVC 2002 DB2” database. Figure 3.17 (b) plots the pseudo-imposter distribution of DTFS_AVGLO algorithm on “FVC 2002 DB1” database. We can observe that the pseudo-imposter distribution is close to imposter

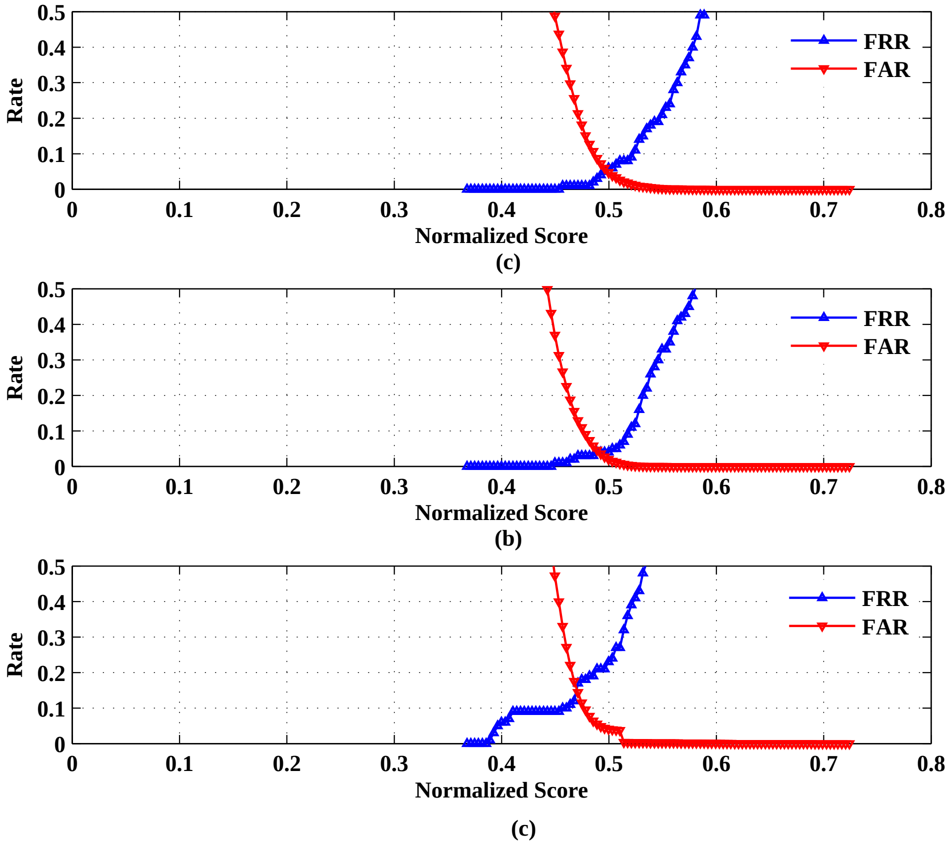
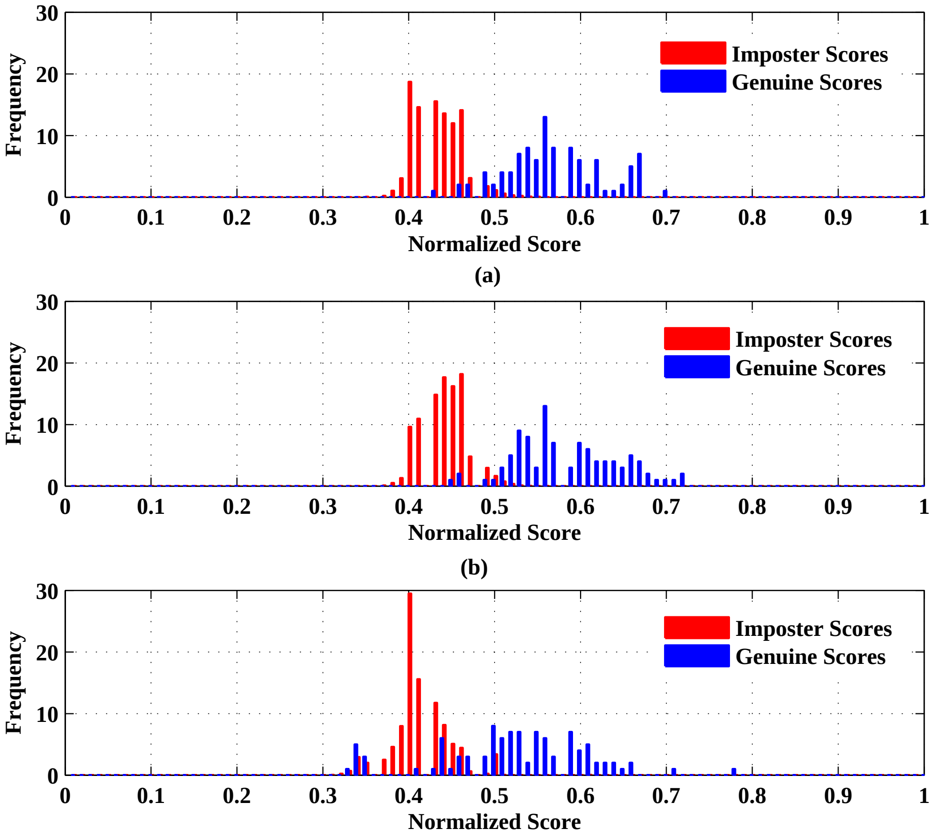


Figure 3.12: FAR/FRR obtained for kNNS algorithm in same key scenario for FVC 2002 (a)DB1 (b)DB2 (c)DB3

distribution and far from genuine distribution in the figures. Further, in kNNS algorithm the mean and standard deviation of imposter distribution is 0.4361 and 0.0288 which are close to mean and standard deviation of pseudo-imposter distribution, *i.e.*, 0.9395 and 0.0334. Hence, we can say that newly transformed templates are different to accorded ones although they are generated from the same fingerprint image.

3.7.4 Security Analysis

Non-invertibility of the transformation function ensures that it is computationally infeasible to reconstruct the original template from the transformed template. For performing the security analysis on the proposed algorithms, we considered the following



(a)

(b)

(c)

Figure 3.13: Genuine and imposter score distributions of kNNS algorithm in same key scenario for FVC 2002 (a)DB1, (b)DB2, (c)DB3

cases:

- **Case 1:** Suppose if kNNS or DTFS is attacked, the location of minutiae points can be guessed by considering the entire image. The image size of "FVC 2002 DB1" database is 388 × 374. The average distance between neighboring minutiae in a fingerprint image is 65 pixels in kNNS algorithm. Hence, the number of brute force attempts needed to guess the minutiae locations is $65 \times 2\pi \times 388 \times 374 \simeq 60$ million. For DTFS_INCIR algorithm, the average distance from minutiae to incircle center is 33 pixels. Hence, the number of brute force attempts required to guess the minutiae locations is $33 \times 2\pi \times 388 \times 374 \simeq 1.72$ billion.

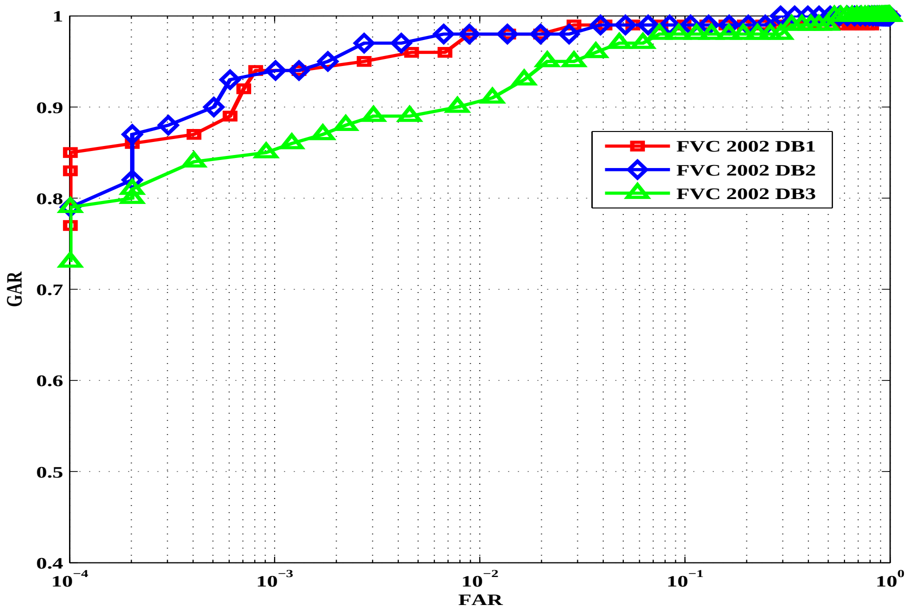


Figure 3.14: ROC curves in same key scenario for DTFS_INCIR algorithm

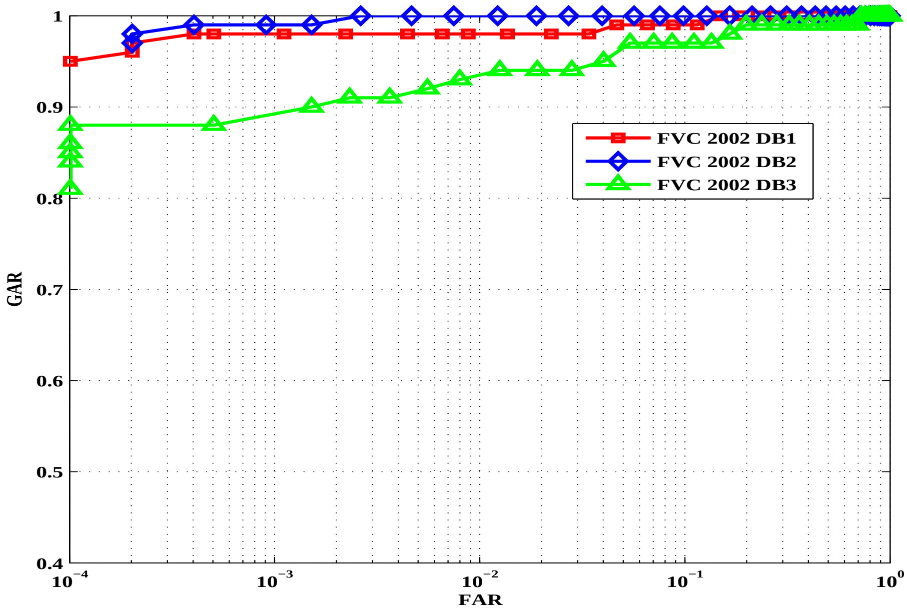
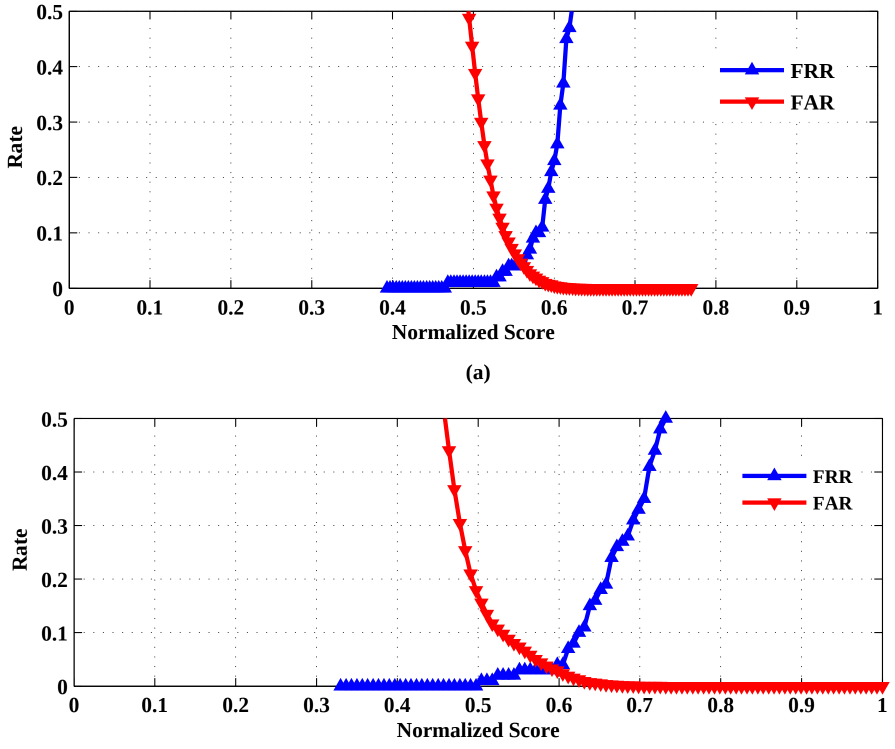


Figure 3.15: ROC curves in same key scenario for DTFS_AVGLO algorithm



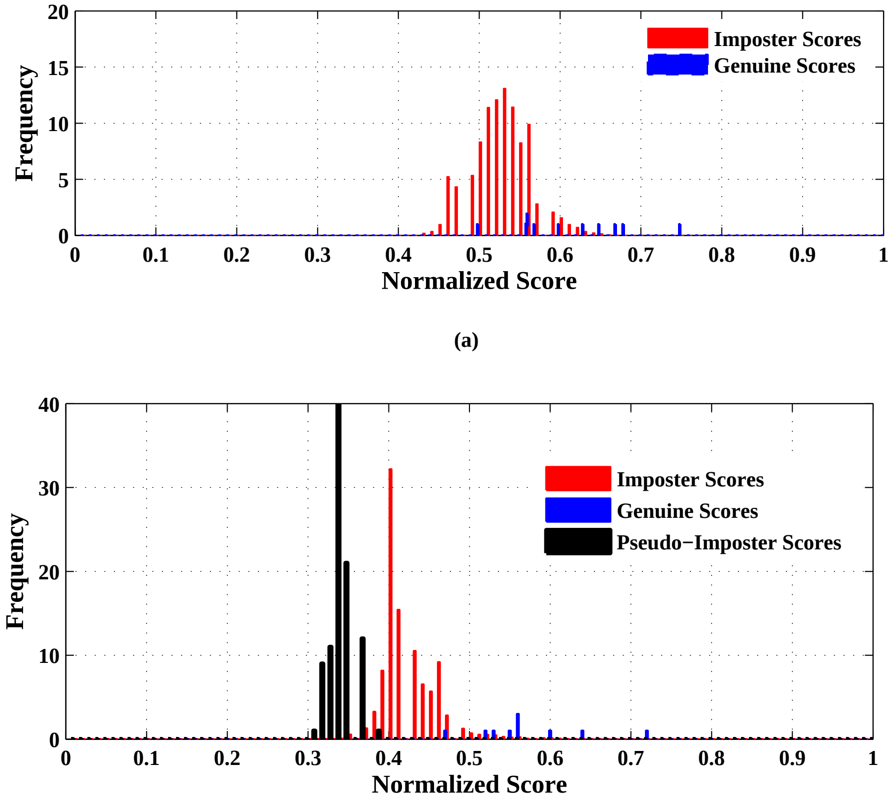
(a)

(b)

Figure 3.16: FAR/FRR obtained in same key scenario for database FVC 2002 DB2 for algorithm (a)DTFS_INCIR (b)DTFS_AVGLO

For DTFS_AVGLO algorithm, the average of edge lengths connecting vertex in Delaunay triangle is 48 pixels. Hence, the number of brute force attempts needed to guess the minutiae locations is $48 \times 2\pi \times 388 \times 374 \simeq 2.5$billion.

- **Case 2:** Assume that an attacker obtains the bit string B_s, the quantized cell sizes C_x, C_y and the neighborhood parameter k in kNNS algorithm. In the experiment, optimal result is obtained for $C_x = 15, C_y = 15$, so number of cells are $W_x = 25, W_y = 24$ and $k = 14$. Hence, for a 388×374 image, the number of brute force attempts needed to guess minutiae location from the bit string and quantized cell size are $388 \times 374 \times 14 \times 25 \times 24 \simeq 1.22$ billion. For DTFS_INCIR algorithm optimal result is obtained for $C_X = 20, C_Y = 20, C_Z = 18$. So, the number of



(a)

(b)

Figure 3.17: Genuine and imposter score distribution for FVC 2002 DB1 using algorithm (a)DTFS_INCIR (b)DTFS_AVGLO

cells in 3D array are $W_X = 8, W_Y = 18, W_Z = 10$. Hence for a 388×374 image, the number of brute force attempts needed to guess minutiae location are $388 \times 374 \times 8 \times 18 \times 10 \simeq 2.35$ billion. For DTFS_AVGLO algorithm, optimal result is obtained for $C_X = 10, C_Y = 10, C_Z = 18$. So, the number of cells in 3D array are $W_X = 16, W_Y = 36, W_Z = 10$. Hence for a 388×374 image, the number of brute force attempts needed to guess minutiae location are $388 \times 374 \times 16 \times 36 \times 10 \simeq 8.35$ billion.

- **Case 3:** Hill-climbing procedure described in [197] generate a fixed number of synthetic templates and attack the biometric system, then accumulates corresponding match scores and picks the best guess, modifies the initial set and

FVC 2002	DB1	DB2	DB3	DB1	DB2	DB3
	Same key scenario			Different key scenario		
kNNS	3.14	3.18	1.66	6.25	5.76	2.99
DTFS_INCIR	3.03	3.21	1.45	9.79	10.19	3.24
DTFS_AVGLO	3.96	3.63	1.56	6.61	7.74	2.52

Table 3.4: d' values obtained for kNNS, DTFS_INCIR, DTFS_AVGLO algorithms in same key and different key scenarios

FVC 2002	DB1	DB2	DB3	DB1	DB2	DB3
	Same key scenario			Different key scenario		
kNNS	0.91	0.93	0.75	1	1	0.93
DTFS_INCIR	0.8795	0.9233	0.8389	1	1	1
DTFS_AVGLO	0.9208	0.9404	0.8655	1	1	0.9671

Table 3.5: KS test values obtained for kNNS, DTFS_INCIR, DTFS_AVGLO algorithms in same key and different key scenarios

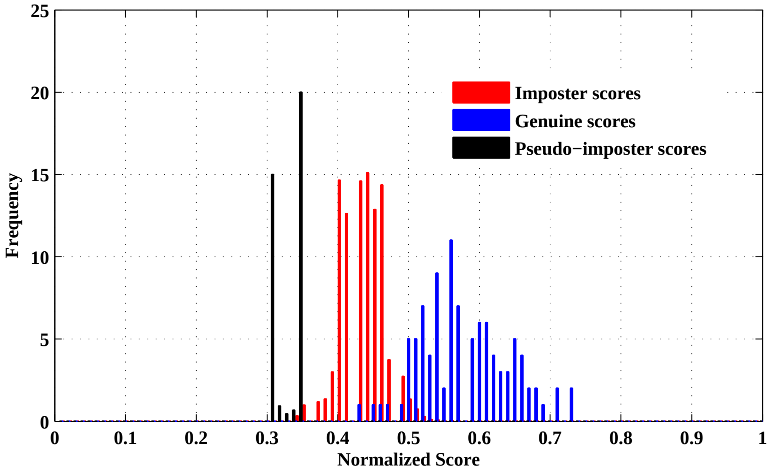


Figure 3.18: Genuine, imposter and pseudo-imposter distribution of kNNS algorithm for database “FVC 2002 DB2” in “same key scenario”

finds the best score accepted by the matcher. In proposed algorithms, once the feature set (DTFS and kNNS) is computed, it is hard to estimate the minutiae locations even through hill-climbing attacks since the transformed feature set does not store the location of minutiae points.

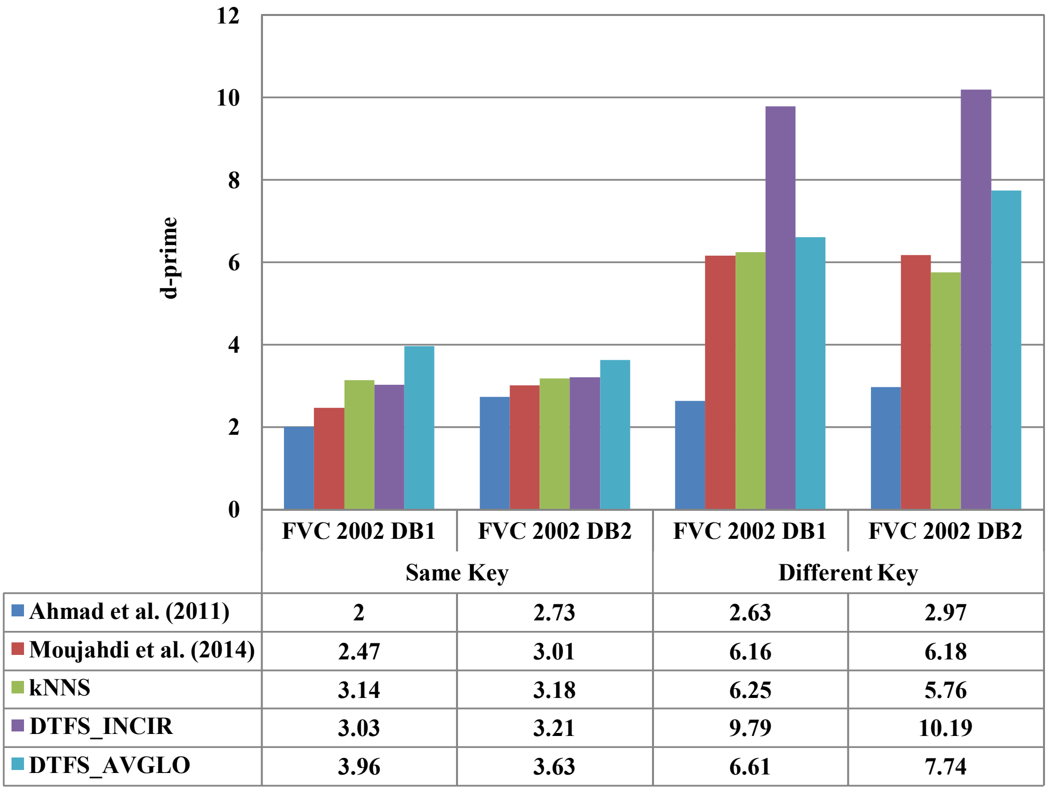
3.8 Performance Comparison with existing methods

For performance comparison of proposed algorithms with existing methods, we used EER, and d' values. The comparison of EER is shown in Table 3.6. From the table, is is observed that, for FVC 2002 DB1 database, kNNS algorithm shown less EER compared to [97, 105, 118], DTFS_INCIR algorithm shown less EER compared to [97], DTFS_AVGLO algorithm shown less EER compared to [18, 97, 105, 118]. For FVC 2002 DB2, all the three algorithms (kNNS, DTFS_INCIR, DTFS_AVGLO) shown less EER compared to [97, 99, 100, 105]. For FVC 2002 DB3, all the three algorithms (kNNS, DTFS_INCIR, DTFS_AVGLO) shown less EER compared to [97, 162]. For FVC 2004 databases, the proposed algorithms (kNNS, DTFS_INCIR, DTFS_AVGLO) outperformed the other methods shown in the Table 3.6.

EER	**FVC 2002**			**FVC 2004**		
	DB1	DB2	DB3	DB1	DB2	DB3
Ahmad et al. (2011) [97]	9	6	27	-	-	-
Das et al. (2012) [100]	2.27	3.79	-	-	-	-
Wang and Hu (2012) [99]	3.5	4	7.5	-	-	-
Jin et al. (2012) [118]	5.19	-	-	15.76	11.64	-
Yang et al. (2013) [105]	5.93	4	-	-	-	-
Yang et al. (2014)[162]	3.38	0.59	9.8	16.52	14.88	-
Jin et al. (2014) [18]	4.36	1.77	-	24.71	21.82	-
Proposed kNNS	4.71	3.44	8.79	15.36	14.25	18.27
Proposed DTFS_INCIR	6.25	3.84	8.16	13.06	12.95	15.23
Proposed DTFS_AVGLO	3.96	2.98	6.89	12.17	13.29	17.73

Table 3.6: EER comparison of kNNS, DTFS_INCIR, DTFS_AVGLO algorithms with existing methods in “same key scenario”

The comparison of d' values is made on FVC 2002 DB1 and DB2 databases in Figure 3.19. From the figure, it can be inferred that the separability is better for the



	FVC 2002 DB1	FVC 2002 DB2	FVC 2002 DB1	FVC 2002 DB2
	Same Key		Different Key	
■ Ahmad et al. (2011)	2	2.73	2.63	2.97
■ Moujahdi et al. (2014)	2.47	3.01	6.16	6.18
■ kNNS	3.14	3.18	6.25	5.76
■ DTFS_INCIR	3.03	3.21	9.79	10.19
■ DTFS_AVGLO	3.96	3.63	6.61	7.74

Figure 3.19: d' value comparison of kNNS, DTFS_INCIR, DTFS_AVGLO algorithms with existing methods

proposed algorithms compared to [97, 113] in both same key and different key scenarios. It is also inferred that d' value is inversely proportional to the EER value obtained, hence proves the credibility of the proposed kNNS, DTFS_INCIR, DTFS_AVGLO algorithms.

3.9 Summary

In this chapter, we presented three algorithms namely kNNS, DTFS_INCIR, and DTFS_AVGLO for generation of cancelable fingerprint templates. The kNNS algorithm is robust with regard to the Cartesian space, for instance, minutiae points need not be linearly separable. Moreover, to construct the kNNS for each minutia point there are only a few parameters to tune: distance metric and neighborhood parameter (k). The triangles in Delaunay triangulation net are structurally more stable under distortion. The usefulness of Delaunay triangulation opposed to other triangulation methods is the fact that triangles are well-shaped [198, 199]. Furthermore, the minutiae insertions and deletions affect the Delaunay triangulation structure locally.

The proposed computation of feature sets kNNS, DTFS_INCIR and DTFS_AVGLO includes all minutiae points in the fingerprint image. This led to maintain a good balance between security and performance of the biometric system. Accuracy, revocability, diversity and irreversibility are clearly analyzed for the proposed algorithms. Further, the experimental evaluation of proposed algorithms is done not only in terms of EER but also in terms of d' and KS test values. Experimental results prove the validity and significance of the proposed algorithms.

Chapter 4

Fusion Techniques for Protecting Fingerprint Templates

Biometric fusion is the use of multiple inputs or methods of processing of biometric samples [54]. It has many advantages such as improved accuracy, efficiency, applicability and robustness [50]. Biometric fusion can be done by using a single trait or multiple traits. A brief discussion of the information fusion in biometric systems and the levels of fusion is presented in section 1.12. Literature survey reveals that applying fusion to a biometric system has many advantages such as improved accuracy, efficiency, applicability and robustness [200, 201, 202, 203, 204, 205, 206, 207]. The contributions of this chapter are described below:

(1) Proposed a feature level fusion technique on transformed features of minutiae points of a fingerprint image namely Local Structure (LS) and Distant Structure (DS).

(2) Proposed a score level fusion technique for generating cancelable fingerprint templates that combine the match scores obtained from two algorithms kNNS and DTFS_AVGLO using T-operators and weighted sum rule.

This chapter is organized as follows: Section 4.1 presents the proposed feature level fusion method. Section 4.2 presents the proposed score level fusion methods. The experiments conducted and analysis is discussed in section 4.3. The performance comparison of proposed methods with existing methods is described in section 4.4. The chapter is concluded by presenting a summary in section 4.5.

4.1 Feature Level Fusion Method

The overall framework for the proposed feature level fusion method (FEATURELEVEL_LSDS) shown in Figure 4.1 contains the following steps :

(1) Computation of transformed features (Local Structure and Distant Structure) for minutiae points

(2) Representing Local Structure (LS) and Distant Structure (DS) as bit strings

(3) Generating fused bit string (B_{fus})

(4) Cancelable template generation

(5) Matching

4.1.1 Computation of transformed features for minutiae points

4.1.1.1 Computation of LS

We compute LS for the fingerprint image as follows :

(1) A minutia point of a fingerprint image is treated as reference minutia r. Consider p nearest minutiae points to r. Now, calculate the distance and orientation difference between r and p local (nearest) minutiae points as follows:

$$\begin{aligned} \chi &= (x_j - x_r)cos\theta_r + (y_j - y_r)sin\theta_r \\ \gamma &= (x_j - x_r)sin\theta_r - (y_j - y_r)cos\theta_r \end{aligned} \tag{4.1}$$

$$d_{jr} = \sqrt{\chi^2 + \gamma^2}, j = 1..p \tag{4.2}$$

$$\theta_{jr} = \begin{cases} \theta_j - \theta_r & if \theta_j > \theta_r \\ \theta_j - \theta_r + 360 & otherwise \end{cases} \tag{4.3}$$

(2) The LS of a reference minutia point r shown in Figure 4.2 can be represented by

$$LS_r = (d_{1r}, \theta_{1r}), (d_{2r}, \theta_{2r}) .... (d_{pr}, \theta_{pr}) \tag{4.4}$$

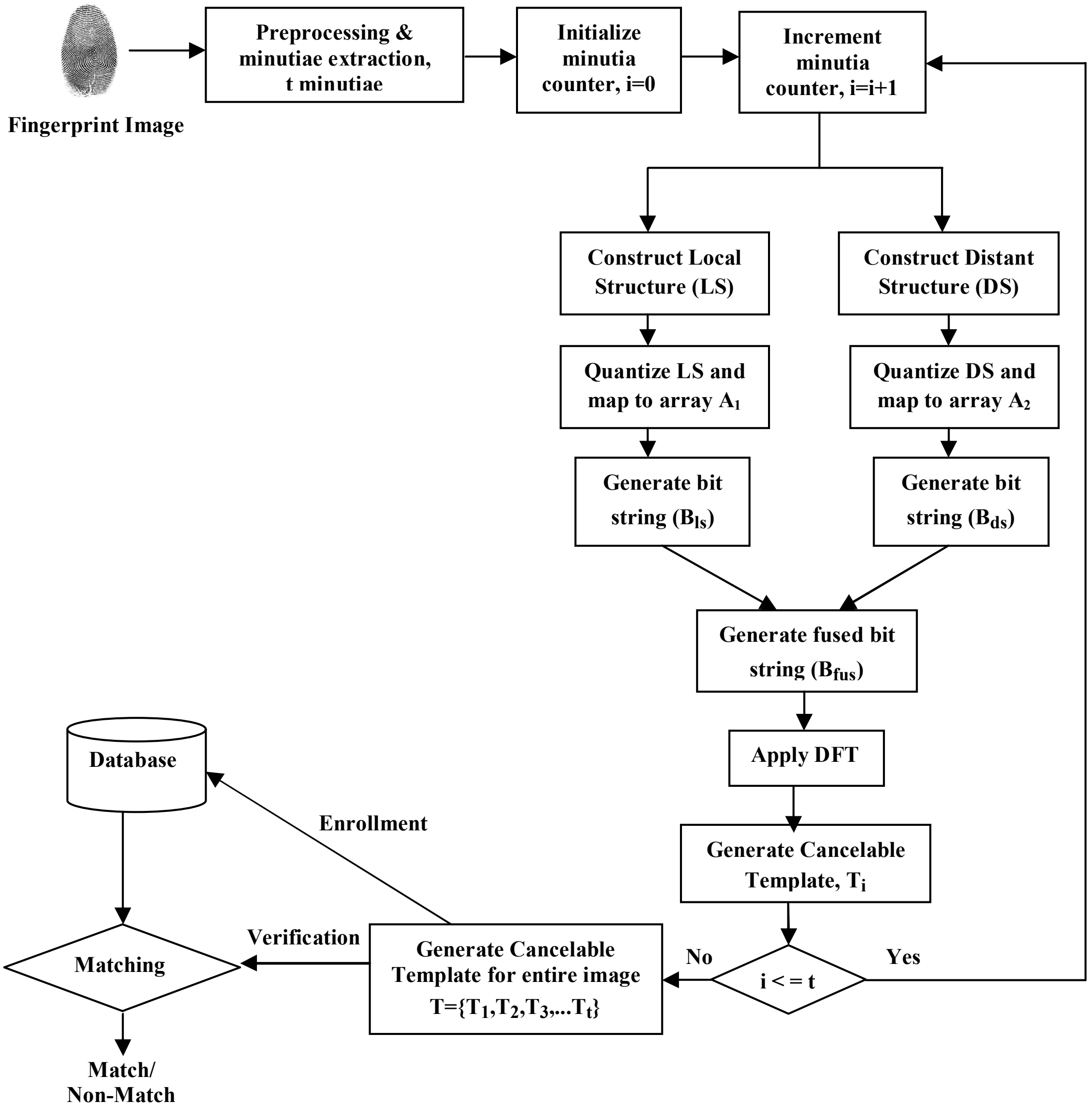

Fingerprint Image
Preprocessing & minutiae extraction, t minutiae
Initialize minutia counter, i=0
Increment minutia counter, i=i+1
Construct Local Structure (LS)
Construct Distant Structure (DS)
Quantize LS and map to array A1
Quantize DS and map to array A2
Generate bit string (Bls)
Generate bit string (Bds)
Generate fused bit string (Bfus)
Apply DFT
Generate Cancelable Template, Ti
i <= t
Yes
No
Generate Cancelable Template for entire image T={T1,T2,T3,...Tt}
Enrollment
Verification
Database
Matching
Match/ Non-Match

Figure 4.1: Overall framework of proposed feature level fusion method

where d_{jr}, θ_{jr} represents the distance and difference of orientation between r and j^{th} local (nearest) neighbor minutiae point respectively.

(3) Repeat steps 1 and 2 for remaining minutiae points in the fingerprint image. Hence, LS for entire fingerprint image can be given by

$$LS = (LS_1, LS_2, ...LS_t) \tag{4.5}$$

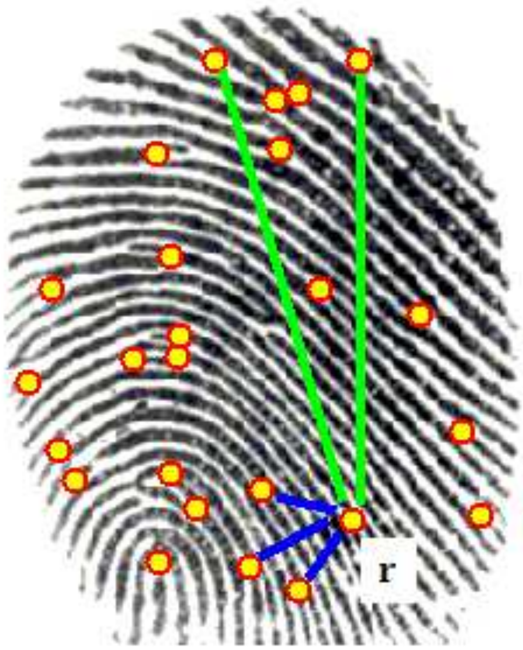


Figure 4.2: Representation of LS_r and DS_r for reference minutia point r with $p = 3$ and $q = 2$

4.1.1.2 Computation of DS

We compute DS for the minutiae points of the fingerprint image which is similar to section 4.1.1.1 but by considering q distant (farthest) minutiae points for r, *i.e.*, j=1...q. Hence, the DS of a reference minutia point r shown in Figure 4.2 can be given by

$$DS_r = (d_{1r}, \theta_{1r}), (d_{2r}, \theta_{2r}) .... (d_{qr}, \theta_{qr}) \tag{4.6}$$

The DS for entire image can be represented as follows :

$$DS = (DS_1, DS_2, ...DS_t) \tag{4.7}$$

4.1.2 Representing LS and DS as bit strings

We take predefined 2D arrays A_1 and A_2 as shown in Figure 4.3 and 4.4. Divide A_1 into cells of size c_x, c_y and A_2 into cells of size c_w, c_h [94]. Each LS_r is projected onto A_1 and each DS_r onto A_2 after quantization. We take d_{jr} on x-axis ranging in [0 $max(d_{jr})$] , θ_{jr} on y-axis ranging in [0 2π].

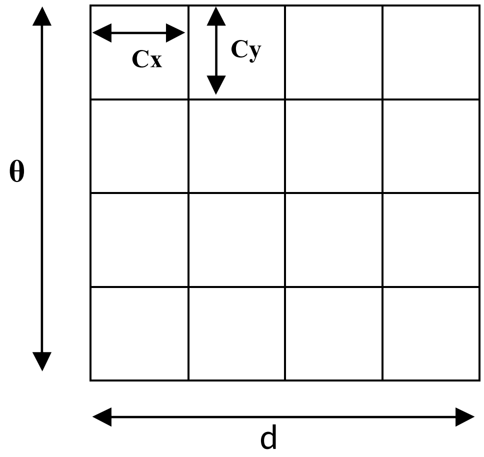


Figure 4.3: Array A_1 with cells of size c_x, c_y on which LS_r is projected.

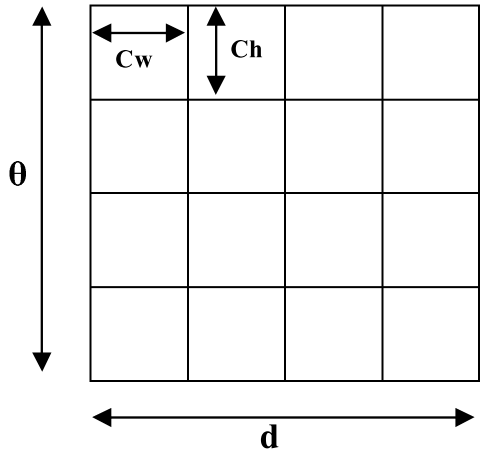


Figure 4.4: Array A_2 with cells of size c_w, c_h on which DS_r is projected.

Now, find the cells in A_1 and A_2 that includes (d_{jr}, θ_{jr}) of LS_r and DS_r by calculating (x_i, y_i) and (w_i, h_i) as follows:

$$\begin{Bmatrix} x_i \\ y_i \end{Bmatrix} = \begin{Bmatrix} \lfloor d_{jr}/c_x \rfloor \\ \lfloor \theta_{jr}/c_y \rfloor \end{Bmatrix} \tag{4.8}$$

$$\begin{Bmatrix} w_i \\ h_i \end{Bmatrix} = \begin{Bmatrix} \lfloor d_{jr}/c_w \rfloor \\ \lfloor \theta_{jr}/c_h \rfloor \end{Bmatrix} \tag{4.9}$$

where x_i, y_i represents indices of A_1 and w_i, h_i represents indices of A_2. The value of cell is 1, if one or more than one (d_{jr}, θ_{jr}) falls in a cell, otherwise 0. By sequentially visiting the cells in A_1 and A_2, fixed length bit strings B_{ls} and B_{ds} are generated.

4.1.3 Fusion of bit strings B_{ls} and B_{ds}

The bit strings B_{ls} and B_{ds} are fused as shown in Figure 4.5. The length of bit strings may vary because of different tuning parameters used, *i.e.*, (p, c_x, c_y) for projecting LS_r and (q, c_w, c_h) for projecting DS_r to a 2D array. Hence, padding of 0's to Most Significant Bits (MSB) of B_{ls} or B_{ds} is used to make the length of the bit strings equal.

```
if (length($B_{ls}$) < length($B_{ds}$)) then
    pad 0's to $B_{ls}$
else
    pad 0's to $B_{ds}$
end if
```

Now, perform XOR operation between B_{ls} and B_{ds}. Finally a circular shift by L bits on XOR'd bit string generates a fused bit string B_{fus}.

4.1.4 Generation of cancelable template

Let the length of bit string B_{fus} be n. We apply a n-point Discrete Fourier Transform (DFT) on B_{fus}. This generates a complex vector V as follows:

$$V = \sum_{s=0}^{n-1} B_{fus} e^{-j2\pi is/n}, i = 0, 1, ...n-1 \tag{4.10}$$

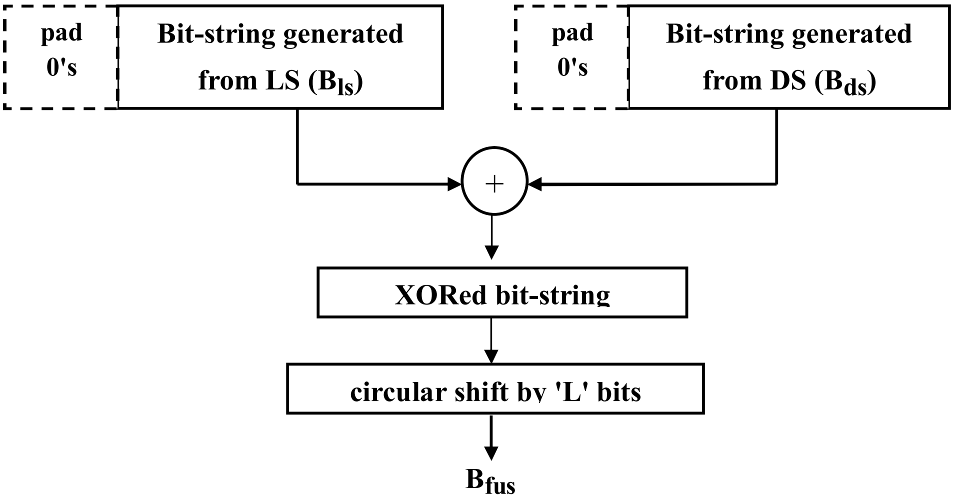


Figure 4.5: Generating fused bit string(B_{fus})

Hence, the size of V is $n \times 1$. Now, a random matrix (R) of dimension $m \times n$ is multiplied to V to obtain the cancelable template T of size $m \times 1$.

$$R \times V = T \tag{4.11}$$

Similarly, we produce t cancelable templates for the fingerprint image represented as $T = (T_1, T_2, ...T_t)$, where t is the mintuiae count.

4.1.5 Matching

Let $T = (T_1, T_2 ...T_u)$ and $\vartheta = (\vartheta_1, \vartheta_2, ...\vartheta_v)$ be the enrolled and query fingerprint templates, where u, v represents number of minutiae. For matching enrolled and query templates, we uses two scores: local match score and global match score.

4.1.5.1 Computing local match score

First, a local matching is done as shown in Figure 4.6. The distance between T_i and ϑ_j is given by

$$d(T_i, \vartheta_j) = \frac{\left\| T_i - \vartheta_j \right\|_2}{\left\| T_i \right\|_2 + \left\| \vartheta_j \right\|_2} \tag{4.12}$$

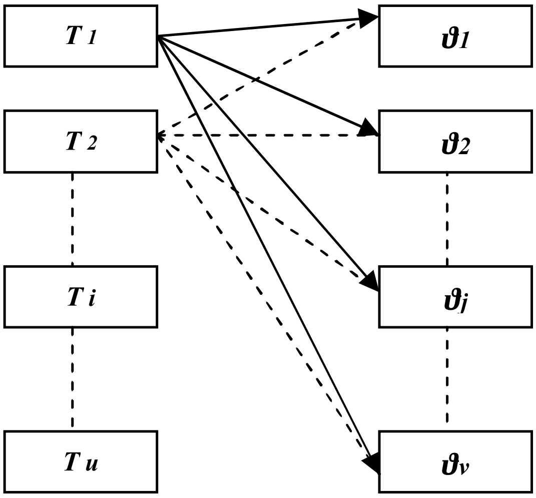


Figure 4.6: Matching between enrolled and query templates

Here, $\left\|.\right\|_2$ indicates a 2-norm. Now, the local match score (LMS) of (T_i, ϑ_j) is computed as follows:

$$LMS(T_i, \vartheta_j) = 1 - d(T_i, \vartheta_j) \tag{4.13}$$

4.1.5.2 Computing global match score

A local score is computed between every T_i in $T = (T_1, T_2 ... T_u)$ and every ϑ_j in $\vartheta = (\vartheta_1, \vartheta_2, ... \vartheta_v)$. This local score is represented as a similarity matrix as follows:

$$MAT(T_i, \vartheta_j) = \begin{cases} max(LMS(T_i, \vartheta_j)) & \forall i \varepsilon [1,u], \forall j \varepsilon [1,v] \\ 0 & Otherwise \end{cases} \tag{4.14}$$

Now, global match score (GMS) is computed as in [104] as follows:

$$GMS = \frac{\sum_{i=1}^{u} \sum_{j=1}^{v} MAT(T_i, \vartheta_j)}{\delta} \tag{4.15}$$

where δ represents number of non-zero entries in $MAT(T_i, \vartheta_j)$. Algorithm 4.1 describes the FEATURELEVEL_LSDS method.

Algorithm 4.1 FEATURELEVEL_LSDS $(x_i, y_i, \theta_i, p, q, A_1, A_2, c_x, c_y, c_w, c_h, L, R)$: To compute Local Structure and Distant Structure and fuse them

```
Input:  Minutiae locations $(x_i, y_i)$
        Orientation of minutiae points $\theta_i$
        Number of Neighbor points considered $p$
        Number of Distant points considered $q$
        Predefined 2D Array $A_1$ divided into cells of size $c_x, c_y$
        Predefined 2D Array $A_2$ divided into cells of size $c_w, c_h$
        Number of Circular shift bits $L$
        Random matrix $R$
Output: Normalized match score

begin
Intialize minutia counter $i \leftarrow 0$;
$t \leftarrow$ Number of minutiae points in a fingerprint image;
while $i \leq t$ do
        $r \leftarrow i$ ;        //Consider each minutia point as reference minutia.
        Intialize counter $j \leftarrow 1$;
        while $j \leq p$ do
//Calculate distance between reference minutia r to $j^{th}$ nearest neighbor.
            $\chi = (x_j - x_r)cos\theta_r + (y_j - y_r)sin\theta_r$;
            $\gamma = (x_j - x_r)sin\theta_r - (y_j - y_r)cos\theta_r$;
            $d_{jr} = \sqrt{\chi^2 + \gamma^2}$;
//Calculate orientation difference between minutiae point r and $j^{th}$ nearest point
            if $(\theta_j - \theta_r) > 0$
                $\theta_{jr} = \theta_j - \theta_r$;
            else
                $\theta_{jr} = \theta_j - \theta_r + 360$;
            end if
            $LS(j,:) \leftarrow (d_{jr}, \theta_{jr})$;
            $j \leftarrow j + 1$;
        end while
```

Algorithm 4.1 FEATURELEVEL_LSDS (continued)

```
        Intialize counter $l \leftarrow 1$;
        while $l \leq q$ do
  //Calculate distance between reference minutia r to $j^{th}$ distant point
            $\chi = (x_j - x_r)cos\theta_r + (y_j - y_r)sin\theta_r$;
            $\gamma = (x_j - x_r)sin\theta_r - (y_j - y_r)cos\theta_r$;
            $d_{jr} = \sqrt{\chi^2 + \gamma^2}$;
  //Calculate orientation difference between minutiae point r and $j^{th}$ distant point
            if $(\theta_j - \theta_r) > 0$
                $\theta_{jr} = \theta_j - \theta_r$;
            else
                $\theta_{jr} = \theta_j - \theta_r + 360$;
            end if
            $DS(j,:) \leftarrow (d_{jr}, \theta_{jr})$;
            $l \leftarrow l + 1$;
        end while
  //Refer to section 4.1.2.
        Quantize LS by cell sizes $c_x, c_y$, map to 2D array $A_1$
        Generate bit string $B_{ls}$                         //Refer to section 4.1.2.
  //Refer to section 4.1.2.
        Quantize DS by cell sizes $c_w, c_h$, map to 2D array $A_2$
        Generate bit string $B_{ds}$                         //Refer to section 4.1.2.
  //Fuse bit strings $B_{ls}$ and $B_{ds}$-Refer to section 4.1.3.
        if (length ($B_{ls}$) <length ($B_{ds}$))
            pad 0's to $B_{ls}$
        else
            pad 0's to $B_{ds}$
        end if
```

Algorithm 4.1 FEATURELEVEL_LSDS (continued)

```
41:             $B_{fus} \leftarrow B_{ls} \oplus B_{ds}$
42:             Circular shift $B_{fus}$ by L bits
   // Generation of Cancelable template-Refer to section 4.1.4
43:             $V \leftarrow DFT(B_{fus})$;
44:             $T(i) \leftarrow R \times V$;
45:             $i \leftarrow i + 1$;
46: end while
47: Match score generation                                   //Refer to section 4.1.5
48: return match_score;
49: end
```

Computational Complexity of FEATURELEVEL_LSDS algorithm
Steps 7 to 18 repeated p times: $O(p)$
Steps 20 to 31 repeated q times: $O(q)$
Hence, Steps 7 to 31: $O(p+q)$
Steps 4 to 46 repeated t times: $O(t)$
Algorithm Complexity: $O(t*(p+q))$

4.2 Score Level Fusion Method

The schematic diagram of proposed score level fusion method is given in Figure 4.7. The match scores obtained from two algorithms, kNNS and DTFS_AVGLO described in chapter 3 are fused using weighted-sum rule and T-operators. This system uses intra-modal fusion, *i.e.*, fusing systems using the same biometric modality but different features.

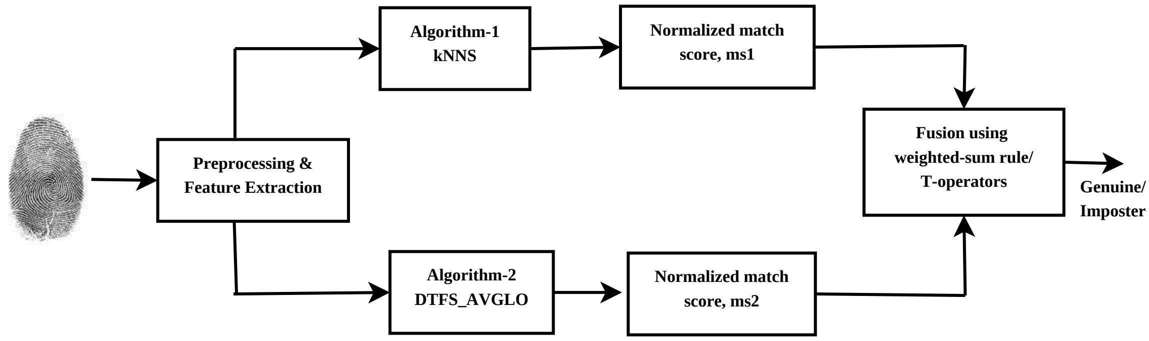


Figure 4.7: Schematic diagram of proposed score level fusion method.

4.2.1 Fusion of match scores of kNNS and DTFS_AVGLO algorithms using weighted sum rule

The normalized match scores (ms_1 and ms_2) obtained from kNNS and DTFS_AVGLO algorithms in chapter 3 are fused at score level using weighted-sum rule. Let w_1 and w_2 be the weights assigned to kNNS and DTFS_AVGLO algorithms respectively. The fused score (W_{fs}) can be given by:

$$W_{fs} = \sum_{i=1}^{n} (w_1 * ms_1(i)) + w_2 * ms_2(i)) \tag{4.16}$$

Weights w_1 and w_2 can be varied over in the range [0,1] such that the condition w_1+$w_2 = 1$ is satisfied [11]. Algorithm 4.2 explains how fusion of match scores returned from kNNS and DTFS_AVGLO algorithms is done using weighted-sum rule.

4.2.2 Fusion of match scores of kNNS and DTFS_AVGLO Algorithm using T-operators

T-operators (T-norms and T-conorms) are the binary functions that satisfy conjunction and disjunction operators respectively [208]. T-conorms are also called as S-norms. T-norms and T-conorms are two place functions that maps a unit square into unit interval defined by $T(S_1,S_2) : [0,1] \times [0,1] \to [0,1]$, $S(S_1,S_2) : [0,1] \times [0,1] \to [0,1]$ respectively. Here S_1 and S_2 represents the match scores ms_1 and ms_2 generated from

Algorithm 4.2 SCORELEVEL_WS(ms_1, ms_2): To compute fused scores from match scores returned from kNNS and DTFS_AVGLO algorithms using weighted-sum rule

```
Input:  Match Scores returned from kNNS algorithm $ms_1$
          Match Scores returned from DTFS_AVGLO algorithm $ms_2$
          Weight assigned to scores of kNNS algorithm $w_1$
          Weight assigned to scores of DTFS_AVGLO algorithm $w_2$
Output: Fused match scores $W_{fs}$

1: begin
2: $n \leftarrow$ length($ms_1$);
3: Initialize counter, $i \leftarrow 0$;
   //Fusion of match scores using weighted-sum rule
4: while $i \leq n$ do
5:         $W_{fs}(i) = w_1 * ms_1(i) + w_2 * ms_2(i)$;
6: end while
7: return $W_{fs}$;
8: end
```

Computational Complexity of SCORELEVEL_WS algorithm

kNNS algorithm complexity (refer to algorithm 3.1) +
DTFS_AVGLO algorithm complexity (refer to algorithm 3.3)

Steps 4 to 6 repeated n times: $O(n)$

Algorithm Complexity: $O(n) * (O(k * t) + O(t + N))$

algorithms kNNS and DTFS_AVGLO respectively. Both T-norms and T-conorms satisfies commutativity, monotonicity, associativity properties [209]. Number 1 acts as identity element for T-norms and number 0 acts as identity element for T-conorms. Since the T-operators (T-norms and T-conorms) are associative, the fusion of two or more algorithms can be in any order. The T-norms and T-conorms implemented in our fusion methods are given in Table 4.1 and 4.2 respectively, where p spans the space of T-operators.

S.No	T-norm	$T(S_1,S_2)$
1.	Zadeh (min rule) [210]	$min(S_1,S_2)$
2.	Goguen,Bandler (product rule) [211]	$S_1.S_2$
3.	Einstein Product [204]	$\frac{S1.S2}{2-(S1+S2-S1.S2)}$
4.	Hamacher [212]	$\frac{S1.S2}{S1+S2-S1.S2}$
5.	Yager $(0 < p < +\infty)$ [213]	$max(1-((1-S_1)^p+(1-S_2)^p)^{1/p},0)$
6.	Schweizer&Sklar $(-\infty < p < +\infty)$ [214]	$(max(S_1^p+S_2^p-1,0))^{1/p}$
7.	Frank $(0 < p < +\infty)$ [215]	$log_p\left(1+\frac{(p^{S_1}-1)(p^{S_2}-1)}{p-1}\right)$
8.	Dombi $(0 < p < +\infty)$ [216]	$\frac{1}{1+\left(\left(\frac{1-S_1}{S_2}\right)^p+\left(\frac{1-S_2}{S_2}\right)^p\right)^{1/p}}$
9.	Dubois $(p\varepsilon[0,1])$ [217]	$\frac{S_1.S_2}{max(S_1,S_2,p)}$
10.	Sugeno-Weber $(-1 < p < +\infty)$ [212]	$max\left(\frac{S_1+S_2-1+p.S_1.S_2}{1+p},0\right)$
11.	Yu Yandong $(-1 < p < +\infty)$ [218]	$max((1+p)(S_1+S_2-1)-p.S_1.S_2,0)$
12.	Aczel-Alsina $(0 < p < +\infty)$ [219]	$e^{-(\|logx\|^p+\|logy\|^p)^{1/p}}$

Table 4.1: T-norms implemented for fusion of match scores.

S.No	T-conorm	$S(S_1,S_2)$
1.	Zadeh (max rule) [210]	$max(S_1,S_2)$
2.	Goguen,Bandler [211]	$S_1+S_2-S_1.S_2$
3.	Dombi$(0 < p < +\infty$) [216]	$\frac{1}{1+\left(\left(\frac{1-S_1}{S_2}\right)^{-p}+\left(\frac{1-S_2}{S_2}\right)^{-p}\right)^{-1/p}}$
4.	Dubois $(p\varepsilon[0,1])$ [217]	$1-\frac{(1-S_1)(1-S_2)}{max(1-S_1,1-S_2,p)}$
5.	Sugeno-Weber $(-1 < p < +\infty)$ [212]	$min(S_1+S_2+p.S_1.S_2,1)$
6.	Yu Yandong $(-1 < p < +\infty)$ [218]	$min(S_1+S_2+p.S_1.S_2,1)$

Table 4.2: T-conorms implemented for fusion of match scores.

Algorithm 4.3 explains how fusion of match scores returned from kNNS and DTFS_AVGLO algorithms is performed using T-operators.

4.3 Experimental Results and Discussion

4.3.1 Experimental Setup

The experimental setup for the proposed fusion methods is same as described in section 3.7.

Algorithm 4.3 SCORELEVEL_TOP(ms_1, ms_2, p): To compute fused scores from match scores returned from kNNS and DTFS_AVGLO algorithms using T-operators

```
Input:  Match Scores returned from kNNS algorithm $ms_1$
        Match Scores returned from DTFS_AVGLO algorithm $ms_2$
        T-norm to be applied on $ms_1, ms_2$ T
        T-conorm to be applied on $ms_1, ms_2$ S
        Space of T-operator $p$
Output: Fused match scores using T-norms $T_{fs}$
        Fused match scores using T-conorms $S_{fs}$

1: begin
2: $n \leftarrow$ length($ms_1$);
3: Initialize counter, $i \leftarrow 0$;
   //Fusion of match scores using T-operators described in Tables 4.1 and 4.2
4: while $i \leq n$ do
5:         $T_{fs}(i) = T(ms_1(i), ms_2(i))$;
6:         $S_{fs}(i) = S(ms_1(i), ms_2(i))$;
7: end while
8: return $T_{fs}$, $S_{fs}$;
9: end
```

Computational Complexity of SCORELEVEL_TOP algorithm

kNNS algorithm complexity (refer to algorithm 3.1) +
DTFS_AVGLO algorithm complexity (refer to algorithm 3.3)

Steps 4 to 7 repeated n times: $O(n)$

Algorithm Complexity: $O(n) * (O(k * t) + O(t + N))$

4.3.2 Accuracy

The feature level fusion method, FEATURELEVEL_LSDS is examined by tuning parameters p, q, c_x,c_y,c_w, c_h and L.

Parameter tuning							FVC 2002			FVC 2004		
p	**q**	**Cx**	**Cy**	**Cw**	**Ch**	**L**	**DB1**	**DB2**	**DB3**	**DB1**	**DB2**	**DB3**
5	1	10	10	10	10	100	2.27	1.82	7.15	11.65	14.65	15.89
6	3	15	15	10	10	100	2.56	1.92	7.62	12.35	13.02	14.48
6	2	20	20	15	15	80	3.53	2.29	7.66	12.65	12.03	17.78
7	4	15	15	12	12	100	2.63	4.48	7.31	13.53	15.27	15.83
8	2	20	20	15	15	100	2.41	1.69	6.24	12.89	12.85	17.47
8	3	20	20	15	15	80	2.40	2.62	6.44	12.63	13.63	17.34
8	**4**	**20**	**20**	**15**	**15**	**100**	**2.19**	**1.6**	**6.14**	**11.89**	**12.71**	**17.60**
10	4	18	18	15	15	150	3.65	1.58	7.26	12.77	12.1	15.68
10	6	18	18	15	15	150	4.63	2.58	11.26	14.56	15.67	17.78

Table 4.3: EER obtained for FVC 2002 and 2004 databases by FEATURELEVEL_LSDS fusion method in same key scenario

- **Same key scenario:** The EER obtained for FVC 2002 and 2004 databases is shown in Table 4.3. It is observed that for $p = 8$, $q = 4$, $c_x = 20$, $c_y = 20$, $c_w = 15$, $c_h = 15$ and $L = 100$, our feature level fusion method yields less EER, *i.e.*, 2.19%, 1.6%, 6.14% EER for FVC 2002 DB1 through DB3. For FVC 2004, an EER of 11.89%, 12.71% and 17.60% was reported for DB1, DB2 and DB3 respectively. Figure 4.8 represents the "ROC curves" of FEATURELEVEL_LSDS method on "FVC 2002" databases. It is observed that the recognition rate for "FVC 2002 DB2" is better than "FVC 2002 DB1, and DB3". Figure 4.9 represents FAR/FRR, and EER obtained for "FVC 2002 databases".

- **Different key scenario** : In this scenario, FEATURELEVEL_LSDS method yields an EER of 0%, 0%, 1.65%, 5.69%, 6.72%, 8.93% for FVC 2002 DB1, DB2, DB3, FVC 2004 DB1, DB2, DB3 respectively for the optimal values $p = 8$, $q = 4$, $c_x = 20$, $c_y = 20$, $c_w = 15$, $c_h = 15$ and $L = 100$ shown in Table 4.3 .

The score distribution of FEATURELEVEL_LSDS method is plotted in Figure 4.10. The figure shows a clear separation of genuine scores from imposter scores in the "same key scenario". Tables 4.4 and 4.5 shows d' and KS test values obtained using FEATURELEVEl_LSDS method for the databases in FVC 2002 and FVC 2004 respectively.

The score level fusion method SCORELEVEL_WS is tested by tuning weights for both algorithms (kNNS and DTFS_AVGLO). A higher weight is assigned to the

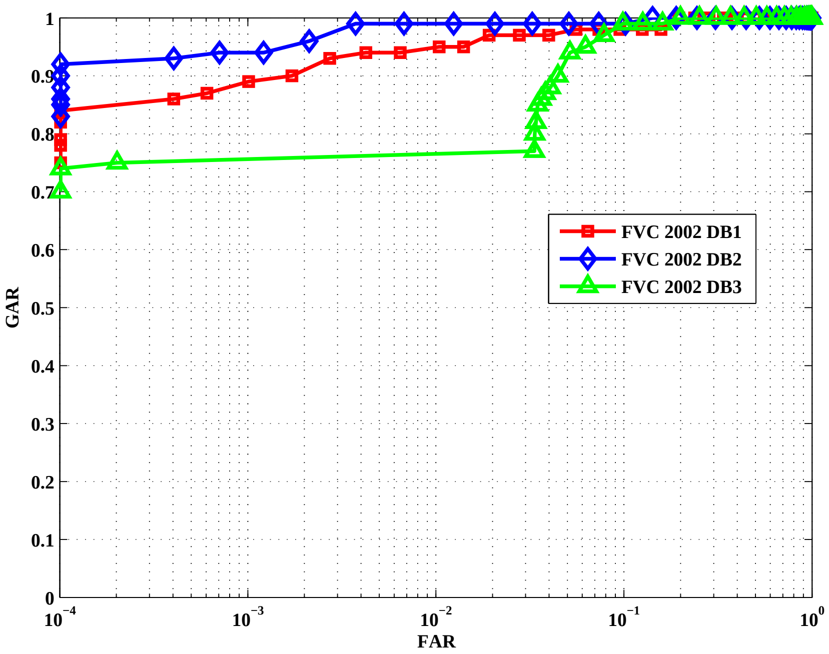


Figure 4.8: ROC curves of FEATURELEVEL_LSDS fusion method on FVC 2002 databases

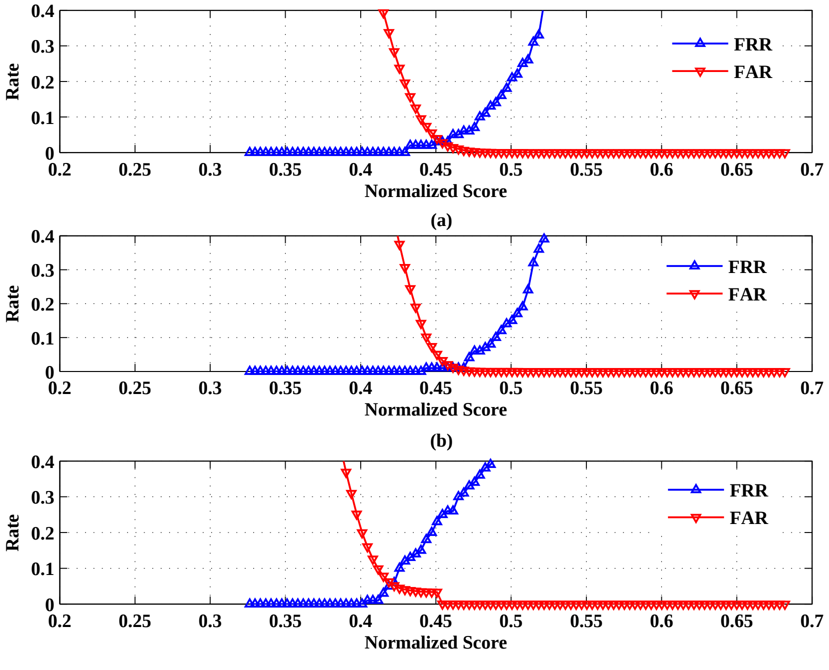
FVC 2002	Same key			Different key		
	DB1	DB2	DB3	DB1	DB2	DB3
d'	3.04	3.24	2.63	6.48	6.58	3.89
KS test	0.96	0.97	0.89	1	1	0.93

Table 4.4: d' and KS test values for FVC 2002 databases using FEATURELEVEL_LSDS fusion method

FVC 2004	Same key			Different key		
	DB1	DB2	DB3	DB1	DB2	DB3
d'	2.17	2.02	1.53	6.48	6.58	3.89
KS test	0.82	0.73	0.69	0.89	0.81	0.73

Table 4.5: d' and KS test values obtained for FVC 2004 databases using FEATURELEVEL_LSDS fusion method

algorithm with a lower mean of match scores. A lower weight is assigned to the algorithm with a higher mean of match scores. Table 4.6 refers to the EER obtained for the proposed score level fusion methods. SCORELEVEL_WS method gives an



(a)

(b)

(c)

Figure 4.9: FAR/FRR of FEATURELEVEL_LSDS fusion method for FVC 2002 (a)DB1 (b)DB2 (c)DB3

optimal result, *i.e.*, EER of 0.75%, 0.54%, 4.74% for “FVC 2002 DB1, DB2 and DB3” respectively when equal weights are assigned to both algorithms. Moreover, the EER is saturated at $w_1 = 0.7, w_2 = 0.3$ assigned to algorithms kNNS and DTFS_AVGLO respectively.

SCORELEVEL_TOP method gives an optimal result *i.e.*, EER of 0%, 0.059%, 3.93% for “FVC 2002 DB1, DB2 and DB3” respectively for Sugeno-Weber T-norm with $p = 0.4$. Figure 4.11 and 4.12 represents the ROC curves and FAR/FRR obtained for SCORELEVEL_WS method for FVC 2002 databases. Figure 4.13 and 4.14 shows the “ROC curves” and “FAR/FRR” obtained for SCORELEVEL_TOP using Sugeno-Weber T-norm with p=0.4 and T-conorm with p= -0.4 for FVC 2002 DB1 database.

Figure 4.15 shows the fused score distribution for SCORELEVL_WS method

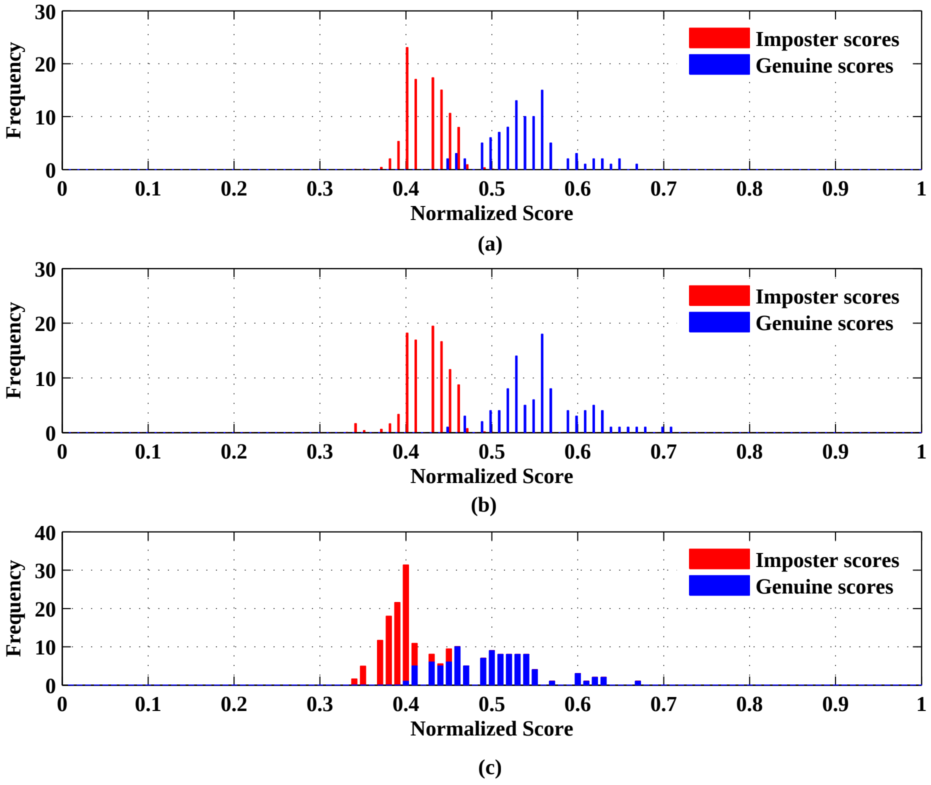


Figure 4.10: Score distributions of FEATURELEVEL_LSDS fusion method for FVc 2002 (a)DB1 (b)DB2 (c)DB3

and Figure 4.16 shows the fused score distribution for SCORELEVL_TOP method using Sugeno-weber T-operators for FVC 2002 databases. Table 4.7, 4.8 4.9 shows the d' and KS test values obtained for the proposed fusion methods FEATURELEVEL_LSDS, SCORELEVEL_WS, and SCORELEVEL_TOP respectively. From the tables, it is observed that the d' value is increased by fusion rather than individual algorithms (kNNS and DTFS_AVGLO) used for fusion. Also, the d' values are inversely proportional to the EER obtained. The KS test values are closer to 1 compared to KS test values obtained from individual algorithms. This resembles how the proposed score level fusion methods achieve better separation between "genuine and imposter" scores.

Algorithm	**weight (w1)**	**weight (w2)**	**p**	**FVC 2002**		
				DB1	**DB2**	**DB2**
kNNS	NA	NA	NA	4.71	3.44	8.79
DTFS_AVGLO	NA	NA	NA	3.96	2.98	6.89
SCORELEVEL_WS	0.5	0.5	NA	0.75	0.54	4.74
	0.55	0.45	NA	0.76	0.57	5.76
	0.6	0.4	NA	0.77	0.21	6.21
	0.65	0.35	NA	0.81	0.57	7.14
	0.7	0.3	NA	0.83	0.57	7.92
Score Level Fusion using T-norms (SCORELEVEL_TOP)						
Zadeh(min rule)	NA	NA	NA	2.91	2.43	8.18
Goguen &Bandler (product rule)	NA	NA	NA	0.84	0.54	5.13
Einstein Product	NA	NA	NA	0.72	0.53	4.93
Hamacher	NA	NA	NA	0.78	0.55	6.42
Yager	NA	NA	0.5	0.74	0.54	4.95
Schweizer &Sklar	NA	NA	-0.2	0.72	0.54	5.39
Frank	NA	NA	0.6	0.72	0.53	5.20
Dombi	NA	NA	0.4	0.69	0.56	4.91
Dubois	NA	NA	1	0.84	0.54	5.13
Sugeno-Weber	**NA**	**NA**	**0.4**	**0**	**0.059**	**3.93**
Yu Yandong	NA	NA	0.4	0.34	0.094	4.17
Aczel-Alsina	NA	NA	1	0.84	0.54	5.13
Score Level Fusion using T-conorms (SCORELEVEL_TOP)						
Zadeh (max rule)	NA	NA	NA	1.88	1.46	4.08
Goguen	NA	NA	NA	0.99	0.53	4.87
Dombi	NA	NA	0.4	0.79	0.56	4.73
Dubois	NA	NA	1	0.99	0.53	4.87
Sugeno-Weber	NA	NA	-0.4	0.72	0.52	4.69
Yu-Yandong	NA	NA	-0.4	0.72	0.52	4.69

Table 4.6: EER obtained for SCORELEVEL_WS and SCORELEVEL_TOP methods for FVC 2002 databases

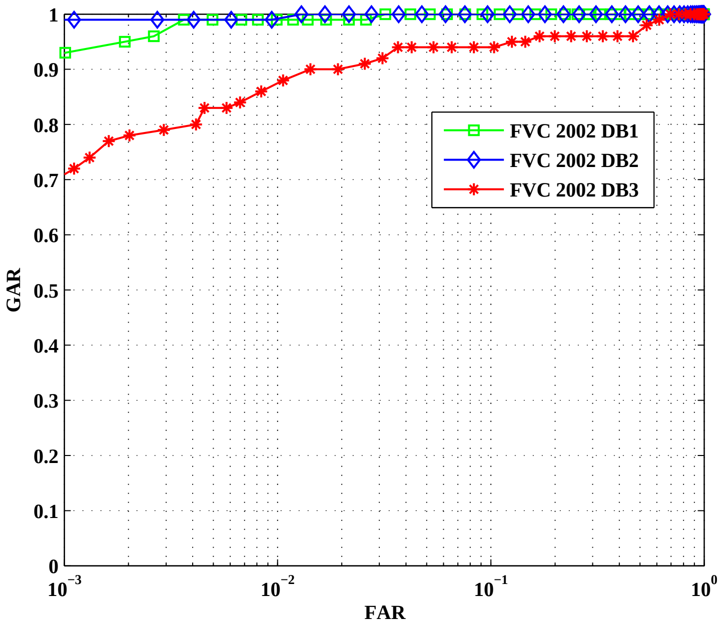


Figure 4.11: ROC curves for SCORELEVEL_WS method with weights $w_1 = 0.5$ and $w_2 = 0.5$

Algorithm	weight (w1) kNNS	weight (w2) DTFS_AVGLO	FVC 2002					
			DB1		DB2		DB3	
			d'	KS test	d'	KS test	d_1	KS test
kNNS	NA	NA	3.14	0.91	2.95	0.93	1.66	0.75
DTFS_AVGLO	NA	NA	3.96	0.92	3.63	0.94	1.56	0.87
Fusion of kNNS & DTFS_AVGLO by weighted-sum	**0.5**	**0.5**	**4.63**	**0.9865**	**4.39**	**0.9896**	**3.05**	**0.9071**
	0.55	0.45	4.61	0.9861	4.31	0.9857	2.91	0.8952
	0.6	0.4	4.58	0.9857	4.22	0.9859	2.79	0.8854
	0.65	0.35	4.47	0.9851	4.28	0.9851	2.50	0.8856
	0.7	0.3	4.35	0.9835	3.95	0.9857	2.49	0.8559

Table 4.7: d' and KS test values of SCORELEVEL_WS method for FVC 2002 databases

4.3.3 Revocability

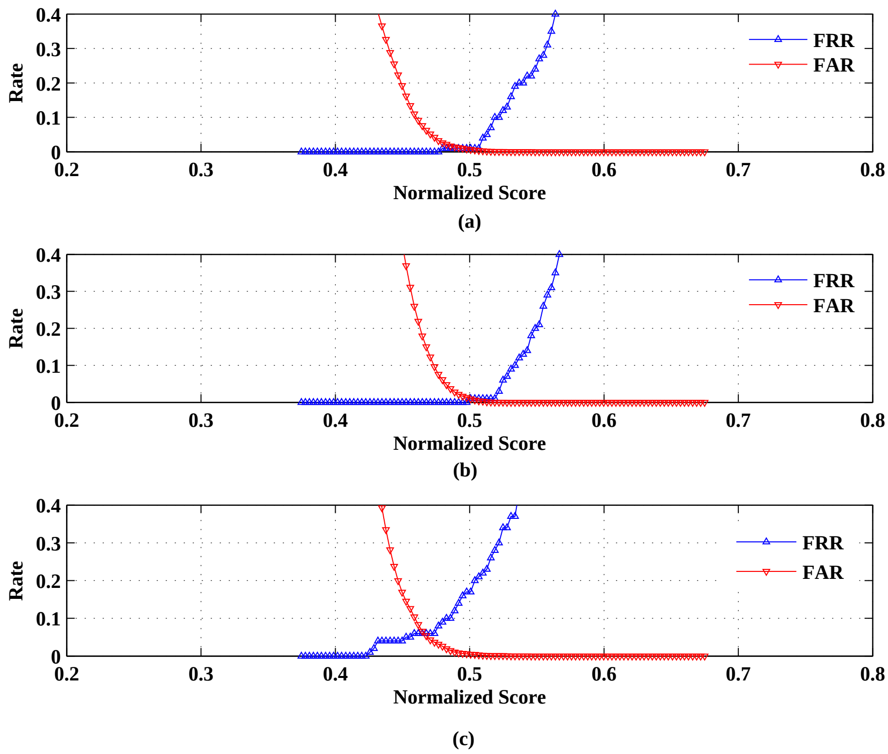
To investigate the revocability of FEATURELEVEL_LSDS method, we calculated pseudo-imposter score distribution. As shown in Figure 4.17, the pseudo-imposter score distribution is clearly separated from genuine score distribution and close to imposter score distribution. This ensures the non-linkability of transformed templates.

T-norm	**p**	**FVC 2002**					
		DB1		**DB2**		**DB3**	
		d'	**KS test**	d'	**KS test**	d'	**KS test**
Zadeh(min rule)	NA	3.8381	0.9421	3.5213	0.9525	1.9286	0.8448
Gugoen & Bandler	NA	4.2509	0.9858	4.0159	0.9896	2.7431	0.9012
Einstein product	NA	4.08	0.9861	3.86	0.9896	2.68	0.9031
Hamacher	NA	4.3082	0.9844	4.0599	0.9895	2.6784	0.8718
Yager	0.5	4.9757	0.9857	4.6566	0.9894	3.2377	0.9022
Schweizer & Sklar	-0.2	4.26	0.9857	4.0224	0.9894	2.7311	0.8921
Frank	0.6	4.2886	0.9857	4.0498	0.9894	2.7515	0.8995
Dombi	0.4	3.9741	0.9862	3.7391	0.9896	2.6161	0.9025
Dubois	1	4.2509	0.9858	4.0159	0.9896	2.7431	0.9012
Sugeno-Weber	0.4	4.1147	0.9867	4.1106	0.9896	3.5213	0.9525
Yu-Yandong	-0.4	4.4837	0.9866	4.3663	0.9896	2.8167	0.9067
Aczel-Alsina	1	4.2509	0.9858	4.0159	0.9896	2.7431	0.9012

Table 4.8: d' and KS test values SCORELEVEL_TOP method using T-norms for databases in FVC 2002.

T-conorm	**p**	**FVC 2002**					
		DB1		**DB2**		**DB3**	
		d'	**KS test**	d'	**KS test**	d'	**KS test**
Zadeh(max rule)	NA	4.1413	0.9644	4.1389	0.971	3.3072	0.9187
Gugoen & Bandler	NA	4.9565	0.9847	4.7166	0.9895	3.3317	0.9052
Dombi	0.4	4.9319	0.9852	4.6687	0.9896	3.3196	0.9038
Dubois	1	4.9565	0.9847	4.7166	0.9895	3.3317	0.9052
Sugeno-Weber	-0.4	5.9814	0.9857	5.921	0.9896	3.6626	0.9066
Yu-Yandong	-0.4	5.9814	0.9857	5.921	0.9896	3.6626	0.9066

Table 4.9: d' and KS test values SCORELEVEL_TOP method using T-conorms for databases in FVC 2002.



(a)

(b)

(c)

Figure 4.12: FAR/FRR obtained for SCORELEVEL_WS method with weights $w_1 = 0.5$ and $w_2 = 0.5$

Similarly, the pseudo-imposter distribution of kNNS and DTFS_AVGLO algorithms are shown in chapter 3. Since, SCORELEVEL_WS and SCORELEVEL_TOP methods are built by kNNS and DTFS_AVGLO algorithms, we can say the score level fusion methods are revocable.

4.3.4 Security analysis of the FEATURELEVEL_LSDS method

Consider the following cases for analyzing security of feature level fusion method

- **Case 'A' :** The number of transformed features mapped onto arrays A_1 and A_2 changes for every reference minutia. Therefore, the reconstruction of LS_r and DS_r is hard to achieve. Hence, the proposed method is secure.

- **Case 'B':** If the biometric system using FEATURELEVEL_LSDS method is attacked by hill-climbing attacks discussed in [197], the minutiae points can not

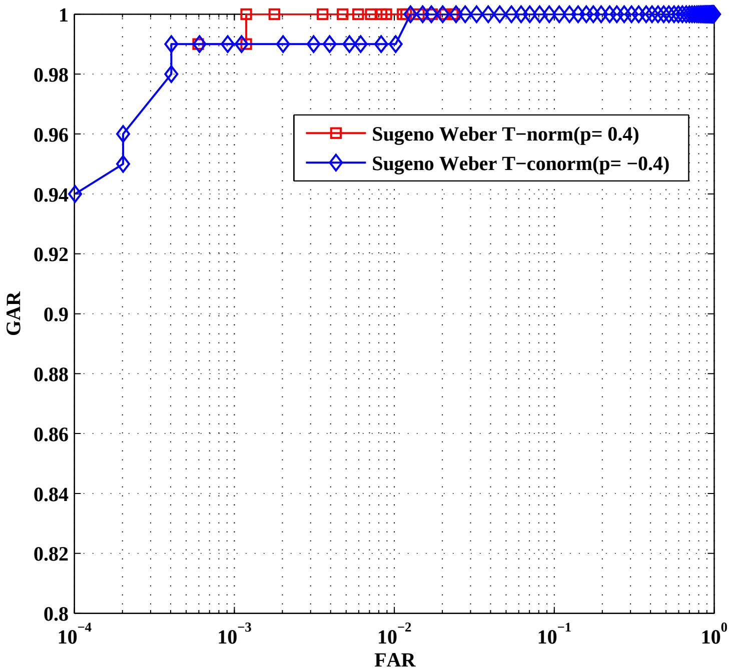
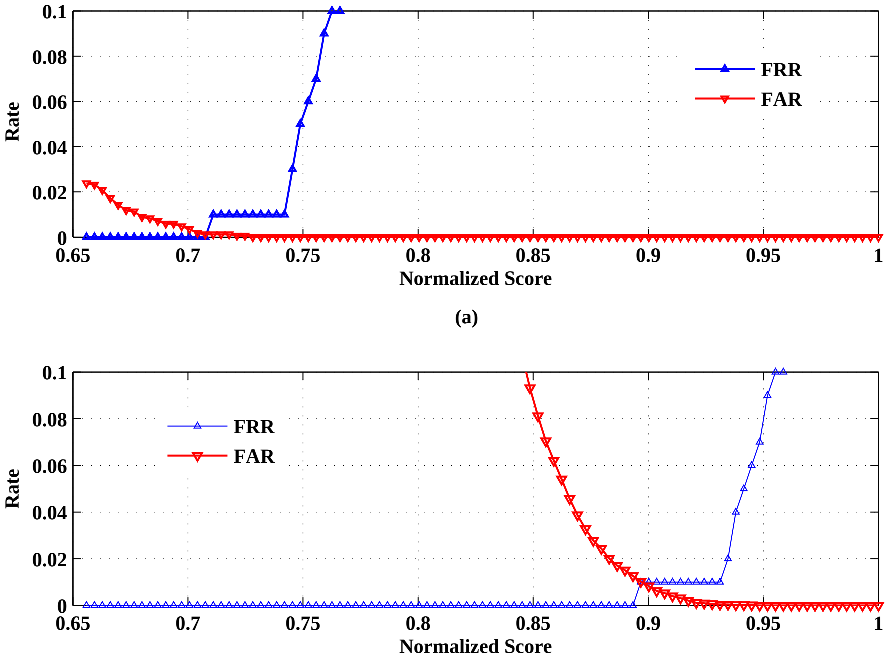


Figure 4.13: ROC curves for SCORELEVEL_TOP method using Sugeno weber T-operators for FVC 2002 DB1

be estimated because the location of minutia points is not stored in the structures, LS, and DS.

- **Case 'C' :** Let us presume that the hacker perceives the bit string B_{fus}, the quantized cell sizes c_x, c_y, c_w, c_h, the parameters p, q, and circular shift bits L. In our experiment the optimal result is for $p = 8$, $q = 4$, $c_x = 20$, $c_y = 20$, $c_w = 15$, $c_h = 15$ and $L = 100$. For a 296×560 image in FVC 2002 DB2 database, the $max(d_{jr})$ for LS is 377, $max(d_{jr})$ for DS is 529. Hence, number of cells in array A_1 is $377/20 \times 360/20 = 18 \times 18 = 324$, number of cells in array A_2 is $529/15 \times 360/15 = 35 \times 24 = 720$. Hence, the bit string length after fusion is $B_{fus} = 720$. So, total number of brute force attempts needed to guess minutiae location from quantized cell sizes, neighborhood parameters p, q, shift bits L, and bit string B_{fus} are $296 \times 560 \times 8 \times 4 \times 720 + 1 \simeq 3.81$ billion.



(a)

(b)

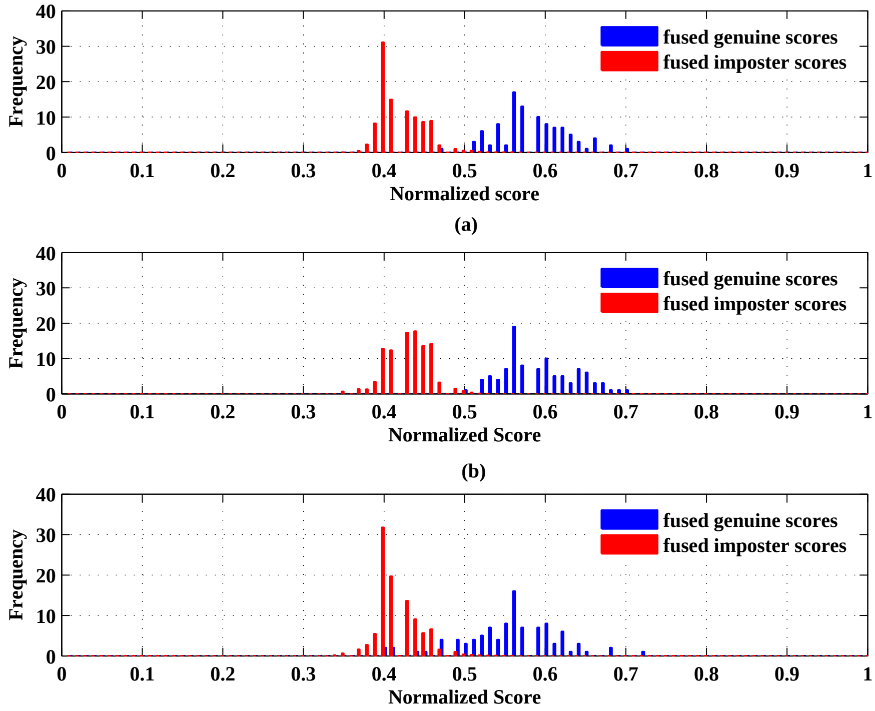
Figure 4.14: FAR/FRR obtained for SCORELEVEL_TOP method using(a)Sugeno-Weber T-norm with p=0.4 (b)Sugeno-Weber T-conorm with p= -0.4 for FVC 2002 DB1

4.3.5 Security Analysis of score level fusion methods

Consider the following cases for performing security analysis on score level fusion methods, SCORELEVEL_WS, and SCORELEVEL_TOP

- **Case 'A':** If an intruder got the bit string B_s, it is hard to reconstruct the feature set in kNNS or DTFS_AVGLO because we used quantization of cells before mapping to a 2D array or 3D array. Also, many to one mapping are done to a cell in the arrays.

- **Case 'B':** Consider the case where attacker got feature vector kNNS or DTFS_AVGLO. The size of an image in FVC 2002 DB1 database is 388×374. The average distance between neighboring minutiae in a fingerprint image is 26 pixels for kNNS algorithm. Hence, $26 \times 2\pi \times 388 \times 374 \simeq 1.35$ billion attempts are needed to guess minutiae locations in kNNS algorithm. We considered the



(a)

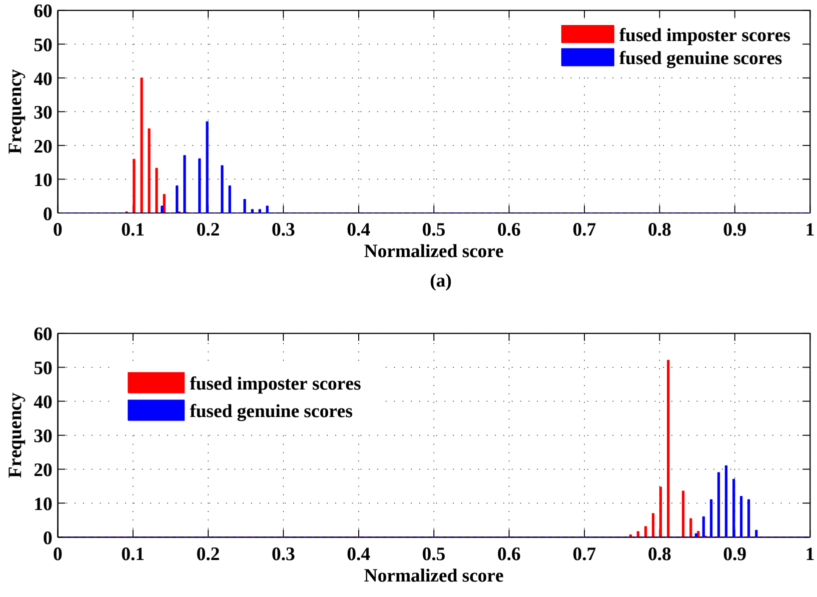
(b)

(c)

Figure 4.15: Fused score distribution of SCORELEVEL_WS method with weights w_1=0.5 and w_2=0.5 for FVC 2002 (a)DB1 (b)DB2 (c)DB3

average of edge lengths at a minutiae point in DTFS_AVGLO algorithm which is 48 pixels. Hence, $48 \times 2\pi \times 388 \times 374 \simeq 2.5$ billion attempts are needed to guess minutiae locations in NNFS algorithm. Hence, the fusion method needs $1.35 + 2.5 \simeq 3.85$ billion attempts to guess minutiae locations.

- **Case ’C’ :** Assume that attacker got the bit string B_s, the quantized cell sizes c_x, c_y and the neighborhood parameter t in kNNS algorithm. The optimal result is obtained for cell sizes $c_x = 15, c_y = 15$ in kNNS algorithm. So, number of cells are $w_x = 25, w_y = 24$ and $t = 14$. Hence, for a 388×374 image, $388 \times 374 \times 14 \times 25 \times 24 \simeq 1.22$ billion attempts are needed to guess a minutiae location. Assume that an attacker knows B_s and quantized cell sizes c_x, c_y, c_z in DTFS_AVGLO algorithm. The optimal result is obtained for cell sizes $c_x = 10, c_y = 10, c_z = 18$, so the number of cells are $w_x = 16, w_y = 36, w_z = 10$. Hence, for a 388×374,



(a)

(b)

Figure 4.16: Fused score distribution of SCORELEVEL_TOP method for FVC 2002 DB1 using (a)Sugeno-Weber T-norm with p=0.4 (b)Sugeno-Weber T-conorm with p= -0.4

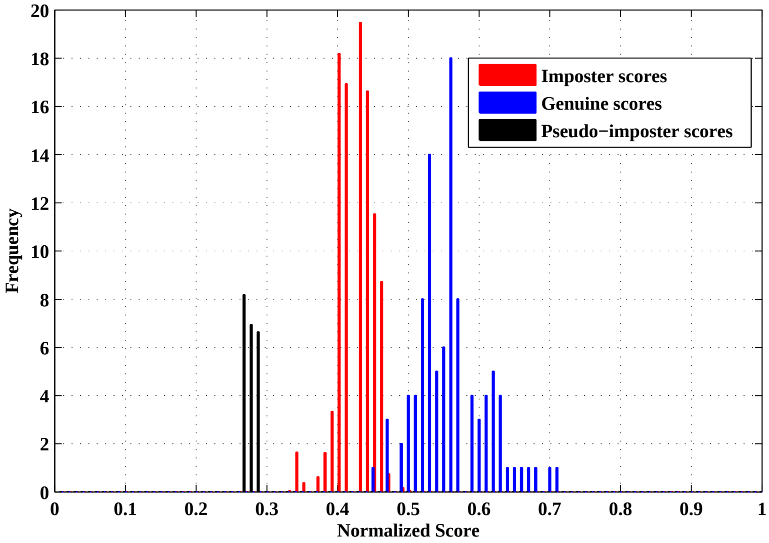


Figure 4.17: Pseudo imposter distribution of FEATURELEVEl_LSDS method for FVC 2002 DB2

$388 \times 374 \times 16 \times 36 \times 10 \simeq 0.835$ billion attempts are needed to guess a minutiae locations. Hence, for score level fusion methods, attacker needs $1.22 + 0.835 \simeq 2.055$ billion attempts to guess the minutiae locations.

- **Case 'D':** If the biometric system using SCORELEVEL_WS or SCORELEVEL_TOP method is attacked by hill-climbing attacks discussed in [197], the minutiae points can not be estimated because the location of minutia points are not stored.

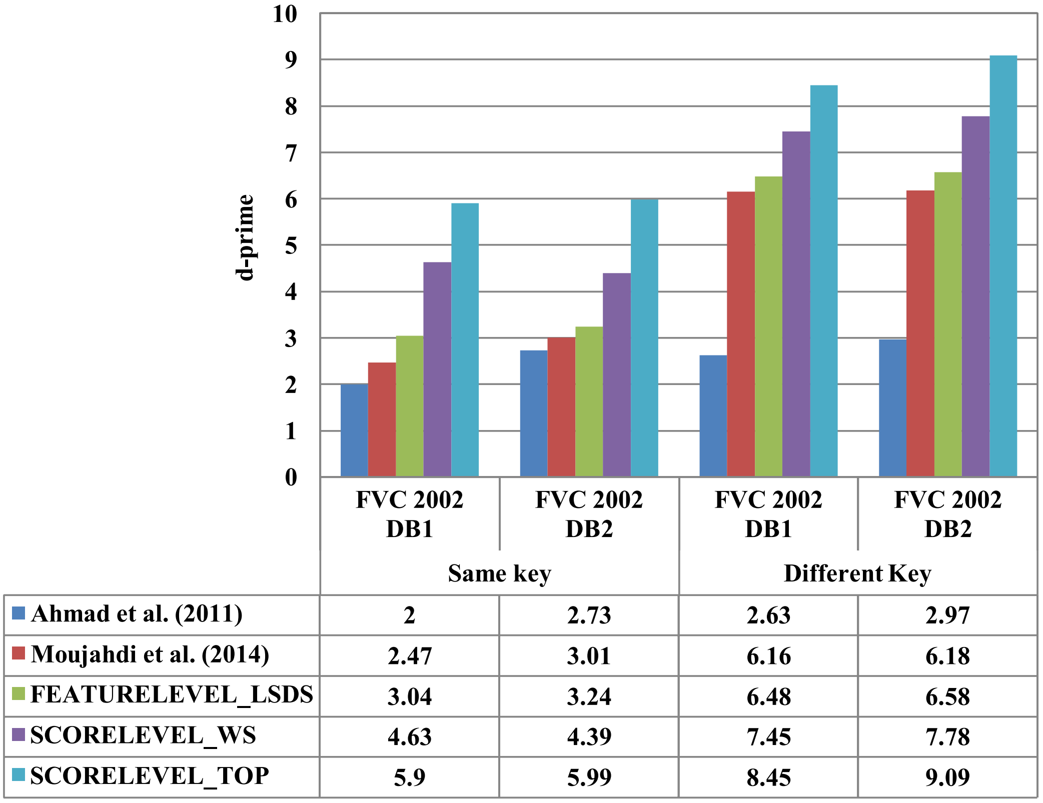
4.4 Performance Comparison with existing methods

The EER comparison of proposed fusion methods with existing methods of cancelable fingerprint generation methods is shown in Table 4.10. We run our proposed fusion methods on FVC 2004 databases and the obtained EERs are reported in Table 4.10. Except the FEATURELEVEL_LSDS method on FVC 2002 DB2 database, the proposed fusion methods shown less EER compared to [97, 99, 100, 105, 118, 162] for all databases shown in Table 4.10.

EER Comparison	**FVC 2002**			**FVC 2004**		
	DB1	**DB2**	**DB3**	**DB1**	**DB2**	**DB3**
Ahmad et al. (2011) [97]	9	6	27	-	-	-
Das et al.(2012) [100]	2.27	3.79	-	-	-	-
Wang and Hu (2012) [99]	3.5	4	7.5	-	-	-
Jin et al. (2012) [118]	5.19	-	-	15.76	11.64	-
Yang et al. (2013) [105]	5.93	4	-	-	-	-
Yang et al.(2014) [162]	3.38	0.59	9.80	24.71	21.82	-
FEATURELEVEL_LSDS	**2.19**	**1.6**	**6.14**	**11.89**	**12.71**	**17.60**
SCORELEVEL_WS	**0.75**	**0.54**	**4.74**	**8.73**	**9.32**	**11.67**
SCORELEVEL_TOP	**0**	**0.059**	**3.93**	**7.13**	**8.49**	**10.95**

Table 4.10: EER comparison of FEATURELEVEL_LSDS, SCORELEVEL_WS, SCORELEVEL_TOP methods in same key scenario with existing methods

The comparison of d' values of proposed methods with the existing methods is made on FVC 2002 DB1 and DB2 databases. It is shown in Figure 4.18 that the separability is better (higher) for the proposed fusion methods compared to [97, 113] in both same key and different key scenarios.



	FVC 2002 DB1	FVC 2002 DB2	FVC 2002 DB1	FVC 2002 DB2
	Same key		Different Key	
■ Ahmad et al. (2011)	2	2.73	2.63	2.97
■ Moujahdi et al. (2014)	2.47	3.01	6.16	6.18
■ FEATURELEVEL_LSDS	3.04	3.24	6.48	6.58
■ SCORELEVEL_WS	4.63	4.39	7.45	7.78
■ SCORELEVEL_TOP	5.9	5.99	8.45	9.09

Figure 4.18: d' value comparison of FEATURELEVEL_LSDS, SCORELEVEL_WS, SCORELEVEL_TOP methods with existing methods

4.5 Summary

In this chapter, we presented feature level and score level fusion methods for generating cancelable fingerprint templates namely FEATURELEVEL_LSDS, SCORELEVEL_WS, and SCORELEVEL_TOP. FEATURELEVEL_LSDS fusion method is built using fused structures at the feature level. The fused structure led to maintain a good balance between security and performance. Experimental results prove the tenability of proposed feature level fusion method.

Score level fusion methods are built by fusing kNNS and DTFS_AVGLO algorithms discussed in chapter 3. Since the algorithm kNNS uses local and global matching between enrolled and query templates, it results in better matching accuracy. By using DTFS_AVGLO, the minutia insertions and deletions affect the feature set computation locally which leads to better matching accuracy. Further, these two algorithms are combined by performing score-level fusion using weighted sum rule and T-operators. Experimental results clearly state that the proposed fusion methods when compared to each of the individual algorithms used for fusing shown better performance. From the experimental analysis, the potential of using T-operators in the generation of cancelable fingerprint templates is extinguished. Also, the intramodal fusion yielded better accuracy compared to individual algorithms used for fusion.

Chapter 5

Bio-Cryptosystems and Hybrid Techniques for Securing Fingerprint Templates

Biometric Cryptosystems which combines research in cryptography and biometrics provides security by binding cryptographic key to biometric features or generating the key using biometric features. The identity (*i.e.*, the biometric features) of a person will be stored in encrypted domain [162]. A detailed discussion on building a key binding biometric cryptosystem using fuzzy commitment scheme is given in section 1.9.1.1. Literature survey reveals that the hybrid template protection approaches (Refer to Figure 2.3), which have been paid little attention so far need to be developed to make use of advantages of multiple schemes, viz. cancelable biometrics and biometric cryptosystems. aThe contributions of this chapter are described below:

(1) A bio-cryptosystem for fingerprints is proposed using Delaunay Neighbor Structures (DNS) computed from fingerprint minutiae points. The fuzzy commitment scheme is employed to secure the template.

(2) Proposed a hybrid technique for protecting fingerprint templates by combining Delaunay Neighbor Structures and fuzzy commitment scheme.

(3) Proposed a hybrid technique for protecting fingerprint templates by combining multiple spiral curves and fuzzy commitment scheme.

(4) A hybrid technique was developed by employing bit string from DTFS_INCIR algorithm in chapter 3. Further, these transformed features are encrypted using convolution coding in the fuzzy commitment scheme.

This chapter is organized as follows: Section 5.1 presents a bio-cryptosystem designed for fingerprints using Delaunay Neighbor Structures. Section 5.2 presents a hybrid technique to protect fingerprint templates using Delaunay Neighbor Structures. Section 5.3 gives a detailed discussion about a hybrid method built for protecting fingerprint templates using multiple spiral curves. Section 5.4 describes a hybrid method for protecting fingerprint templates using convolution coding. Section 5.5 presents a performance comparison of proposed methods with existing methods. The chapter is concluded by presenting a summary in section 5.6.

5.1 Fingerprint Cryptosystem using Delaunay Neighbor Structures

We name the method of devising a fingerprint cryptosystem using "Delaunay Neighbor Structures" as BIOCRYP_DNS. The flow diagram for BIOCRYP_DNS is shown in Figure 5.1. It contains the following steps :

(1) Computation of "Delaunay Neighbor Structures (DNS)"

(2) Bit string generation from DNS

(3) Apply Dimensionality reduction matrix to the bit string

(4) Employing fuzzy commitment scheme

(5) Matching

5.1.1 Computation of DNS

Construct a "Delaunay triangulation net" from the minutiae points of a fingerprint image as shown in Figure 5.2. The DNS centered around minutia *r* contains neighboring

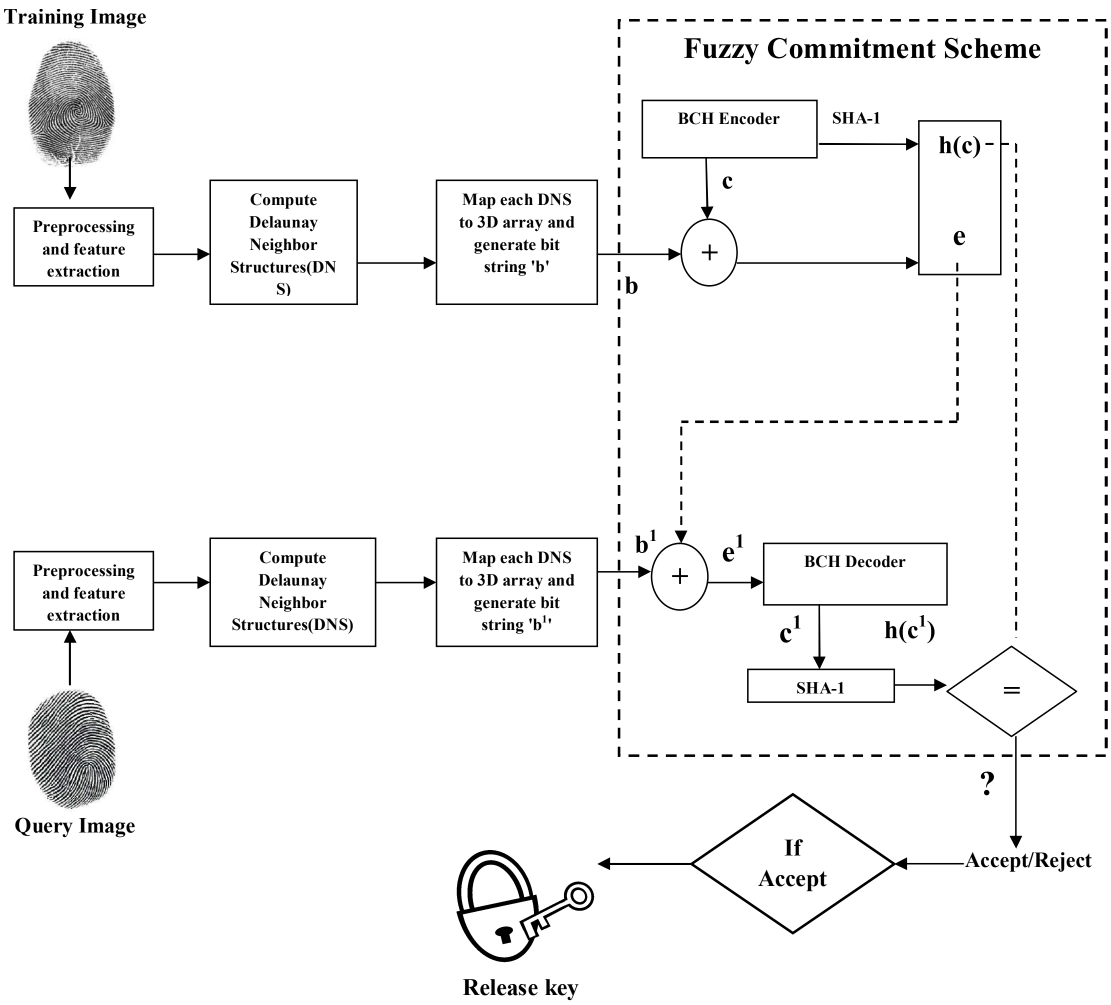

Training Image
Fuzzy Commitment Scheme
BCH Encoder
SHA-1
h(c)
c
Preprocessing and feature extraction
Compute Delaunay Neighbor Structures(DNS)
Map each DNS to 3D array and generate bit string 'b'
+
e
b
b1
e1
+
BCH Decoder
Preprocessing and feature extraction
Compute Delaunay Neighbor Structures(DNS)
Map each DNS to 3D array and generate bit string 'b1'
c1
h(c1)
SHA-1
=
?
Query Image
If Accept
Accept/Reject
Release key

Figure 5.1: Flow diagram for the BIOCRYP_DNS method

minutiae as shown in Figure 5.3(a) denoted by

$$DNS_r = (m_r, m_1, m_2, m_3, m_4, m_5, m_6) \tag{5.1}$$

where m_i denotes the minutia point i. The DNS is computed for all minutiae points in the entire image and denoted by

$$DNS = (DNS_1, DNS_2, DNS_3 .....DNS_t) \tag{5.2}$$

where t is number of minutiae points in the image.

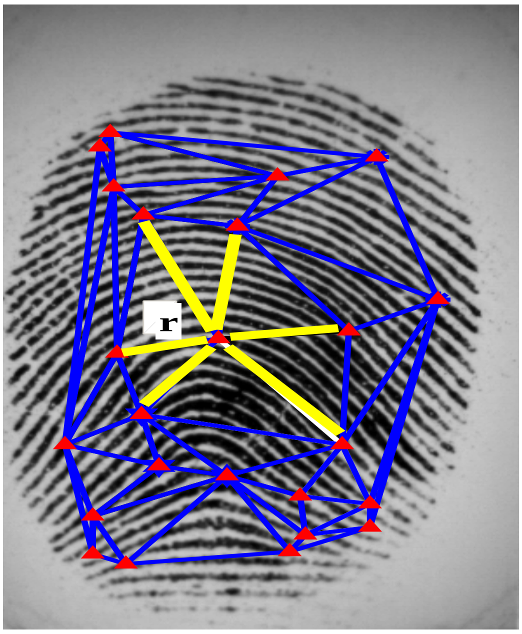


Figure 5.2: Delaunay triangulation formed from fingerprint minutiae and DNS for minutia point *r*

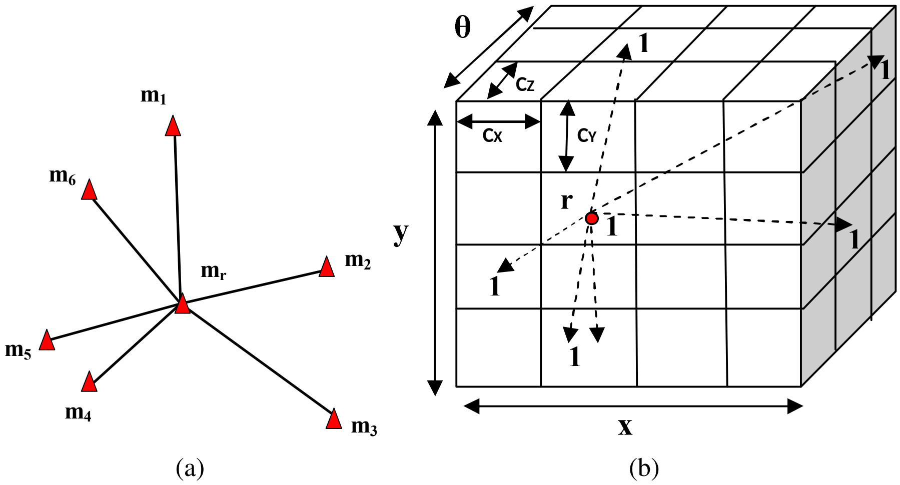


Figure 5.3: (a)DNS of minutia point *r* (b) Mapping DNS to a 3D array

5.1.2 Bit string generation from DNS

We take a 3D array of size $W_x \times W_y \times W_z$ as shown in Figure 5.3(b), where W_x ranges from 0 to max(x-coordinate value), W_y ranges from 0 to max(y-coordinate value), W_z ranges from 0 to 2π. Divide the array into cells of size c_x, c_y, c_z. The central minutia r of each DNS is taken as reference. It is located at center of first layer of 3D array, *i.e.*, $(W_x/2, W_y/2, 0)$. The remaining minutiae in the DNS are rotated and transformed to $(x_i^1, y_i^1, \theta_i^1)$ as follows :

$$\begin{bmatrix} x_i^1 \\ y_i^1 \end{bmatrix} = \begin{bmatrix} cos\theta_r & -sin\theta_r \\ sin\theta_r & cos\theta_r \end{bmatrix} \begin{bmatrix} (x_i - x_r) \\ -(y_i - y_r) \end{bmatrix} + \begin{bmatrix} W_x/2 \\ W_y/2 \end{bmatrix} \tag{5.3}$$

$$\theta_i^1 = \begin{Bmatrix} \theta_i - \theta_r & if \theta_i \geq \theta_r \\ 2\pi + \theta_i - \theta_r & if \theta_i < \theta_r \end{Bmatrix} \tag{5.4}$$

Now, a quantization process is applied by the cell sizes c_x, c_y, c_z to $(x_i^1, y_i^1, \theta_i^1)$ as follows :

$$\begin{bmatrix} x_i^{11} \\ y_i^{11} \\ \theta_i^{11} \end{bmatrix} = \begin{bmatrix} x_i^1/c_x \\ y_i^1/c_y \\ \theta_i^1/c_z \end{bmatrix} \tag{5.5}$$

Now, we project $(x_i^{11}, y_i^{11}, \theta_i^{11})$ onto 3D array. If a point or more than one point falls in a cell its value is set to 1 otherwise 0 as shown in the Figure 5.3(b). Visiting the cells in a sequential manner produce a bit string. Similarly, for t DNSs computed, t bit strings are generated as follows:

$$b_i = (b_1, b_2, b_3 ... b_t), i = 1..t \tag{5.6}$$

5.1.3 Apply Dimensionality reduction matrix to each bit string

Dimensionality reduction is required at this stage to reduce the size of bit string to perform Bose, Chaudhuri, and Hocquenghem (BCH) encoding. Let the bit strings computed are of size $m \times 1$. Multiply each bit string b_i with a random matrix (with 0's and 1's) of size $n \times m$, where n is the value used in [n,k,λ] BCH encoder discussed in section 5.1.4.

5.1.4 Employing Fuzzy Commitment Scheme

The BCH error correcting code is a random cyclic code represented as BCH (n,k,λ), where n is the message, k is the message word, λ is the error correction capability [220]. For prime power q, positive integers m and d such that $d <= q^m - 1$, BCH code defined over finite field $GF(q)$ with code length $n = q^m - 1$ and distance at least d is given as follows :

(1) Let α be a primitive element of $GF(q^m)$.

(2) Let $m_i(x)$ be the minimal polynomial of α_i over GF(q) for positive integer 'i'.

(3) The generator polynomial of the BCH code is given by the least common multiple $g(x) = LCM(m_1(x),\ldots,m_{d-1}(x))$.

(4) Now, g(x) is a polynomial with coefficients in GF(q) and divides $x^n - 1$. Hence, the polynomial code defined by g(x) is cyclic. This is called narrow sense BCH code.

A fuzzy commitment scheme [27] takes a binary feature vector b_i as input. In the encoding phase, a random key is taken and encoded using BCH coding to produce a codeword c. Now an encrypted template e is obtained by performing XOR operation between b_i and c as follows:

$$e_i = b_i \oplus c, i = 1..t \tag{5.7}$$

The hash value of c, say $h(c)$ is computed and stored along with e_i. For BIOCRYP_DNS method we used cryptographic hash function SHA-1 to compute the hash value.

In the decoding phase, a binary feature vector b' of query image is presented. b'_i is XOR'd with e_i as follows:

$$e'_i = b'_i \oplus e'_i, i = 1..t \tag{5.8}$$

A decoding algorithm is applied to e' to get c'. The hash value of c' is computed. If $h(c) = h(c')$, which shows that the decoded codeword c' is equal to codeword c. Hence

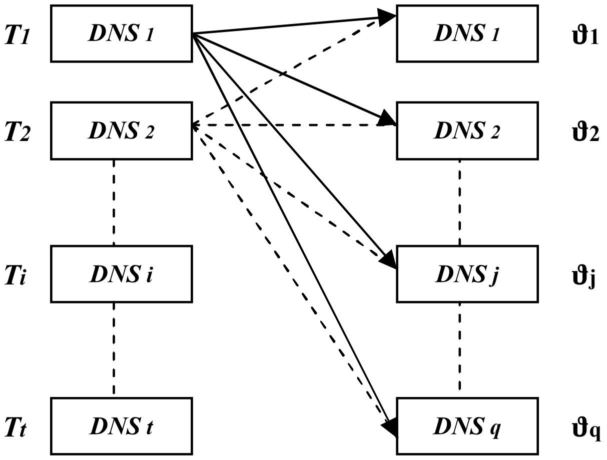


Figure 5.4: Computing local score between encrypted templates of trained and testing images

the user is considered to be genuine and accepted, else considered as imposter and rejected as shown in Figure 5.1.

5.1.5 Matching

Let $T = (T_1, T_2 ... T_t)$ be the trained image template and $\vartheta = (\vartheta_1, \vartheta_2, ...\vartheta_q)$ be the testing image template. Matching is the process of comparing the T and ϑ. The hamming measure, say HM is the percentage of bits that differ between T_i and ϑ_j. We employ a local matching procedure as shown in Figure 5.4. The local match score between T_i and ϑ_j can be given by

$$LMS(T_i, \vartheta_j) = HM(T_i, \vartheta_j) \tag{5.9}$$

We matched every T_i in $T = (T_1, T_2 ... T_t)$ with every ϑ_j in $\vartheta = (\vartheta_1, \vartheta_2, ...\vartheta_q)$ that results in a similarity matrix as follows:

$$S(T_i, \vartheta_j) = \begin{cases} max(LMS(T_i, \vartheta_j)) & \forall i\varepsilon[1,t], \forall j\varepsilon[1,q] \\ 0 & Otherwise \end{cases} \tag{5.10}$$

From this similarity matrix, the final matching score is generated as follows:

$$MS = \frac{\sum_{i=1}^{t} \sum_{j=1}^{q} S(T_i, \vartheta_j)}{\eta} \tag{5.11}$$

where η represents the number of non-zero elements in $S(T_i, \vartheta_j)$. Algorithm 5.1 describes the BIOCRYP_DNS method.

Algorithm 5.1 BIOCRYP_DNS ($x_i, y_i, \theta_i, A_1, c_x, c_y, c_z, n, k, \lambda$): To compute Delaunay Neighbor Structures and generate secure template

```
Input:  Minutiae locations $(x_i, y_i)$
        Orientation of minutiae points $\theta_i$
        Predefined 3D Array $A_1$ divided into cells of size $c_x, c_y, c_z$
        Input to BCH encoder $n, k, \lambda$
Output: Normalized match score

begin
t ← number of minutiae points in the fingerprint image;
// Refer to Figure 5.2
Construct Delaunay triangulation net from the fingerprint minutia;
Initialize minutia counter $i \leftarrow 0$;
while $i \leq t$ do
//Find neighbor minutiae in Delaunay triangulation net for each minutia point-Refer to Figure 5.3(a)
        $DNS(i) \leftarrow$ Neighboring minutiae of $i$ in Delaunay triangulation net;
        Quantize $DNS(i)$ by cell sizes $c_x, c_y, c_z$, map to 3D array $A_1$;
        Generate bit string $B_s$;                    //Refer to section 5.1.2.
// Apply Dimensionality reduction matrix to bit string-Refer to section 5.1.3
        $m \leftarrow$ length($B_s$);
        Define random matrix R of size (m,n) filled with 0's and 1's;
        $b \leftarrow B_s \times R$;
        Codeword $c \leftarrow BCH(n, k, \lambda)$;   //Refer to section 5.1.4
        $e \leftarrow b \oplus c$;
//Hash value generation
        $h \leftarrow$ SHA-1(c);
// Store encrypted template e
        $Encrypt\_temp(i) \leftarrow e$;
```

Algorithm 5.1 BIOCRYP_DNS (continued)

```
16:             $i \leftarrow i+1$;
17:     **end while**
18:     Match score generation;                                   //Refer to section 5.1.5
19:     **return** match_score;
20:     **end**
```

Computational complexity of BIOCRYP_DNS algorithm

Delaunay triangulation formation from t minutiae points: $O(t)$

Steps 5 to 17 repeated t times: $O(t)$

Algorithm Complexity: $O(t+t) = O(t)$

5.1.6 Experimental Results and Discussion

The BIOCRYP_DNS method is tested on FVC 2002 databases by taking 2 samples of each fingerprint image, one for training and one for testing. BIOCRYP_DNS method is tested for different values of (n,k,λ) of BCH encoder and by tuning the cell sizes c_x,c_y and c_z in the quantization step.

5.1.6.1 Accuracy

Table 5.1 represents the EER obtained for BIOCRYP_DNS method. For cell sizes $c_x = 20$,$c_y = 20$ and $c_z = 20$, the method show optimal result, *i.e.*, an EER of 1.43%, 1.79% and 5.89% for FVC 2002 DB1, DB2 and DB3 respectively for (1023,11,255) BCH encoder.

Figure 5.5, 5.6 and 5.7 depicts the ROC curves, EER obtained and score distribution for BIOCRYP_DNS method on FVC 2002 databases respectively.

5.1.6.2 Revocability

The Dimensionality reduction matrix used in section 5.1.3 reduces the size of the bit string. This is revocable. The use of a key to generate the codeword through BCH

Cell sizes			**(n,k,λ)**								
			(255,63,33)			**(511,259,30)**			**(1023,11,255)**		
c_x	c_y	c_z	**DB1**	**DB2**	**DB3**	**DB1**	**DB2**	**DB3**	**DB1**	**DB2**	**DB3**
15	15	15	4.64	5.13	14.96	6.57	5.89	12.41	4.12	4.15	11.87
15	20	15	5.76	7.67	15.87	5.13	6.12	17.97	3.84	3.44	11.86
20	**20**	**20**	**3.26**	**3.98**	**9.85**	**3.14**	**3.48**	**8.89**	**1.43**	**1.79**	**5.89**
20	25	20	5.14	7.63	10.43	4.63	4.93	14.96	1.89	2.12	6.89
25	25	25	6.57	5.89	12.43	5.74	7.63	15.82	2.98	3.02	7.16

Table 5.1: EER resulted for BIOCRYP_DNS method for FVC 2002 databases

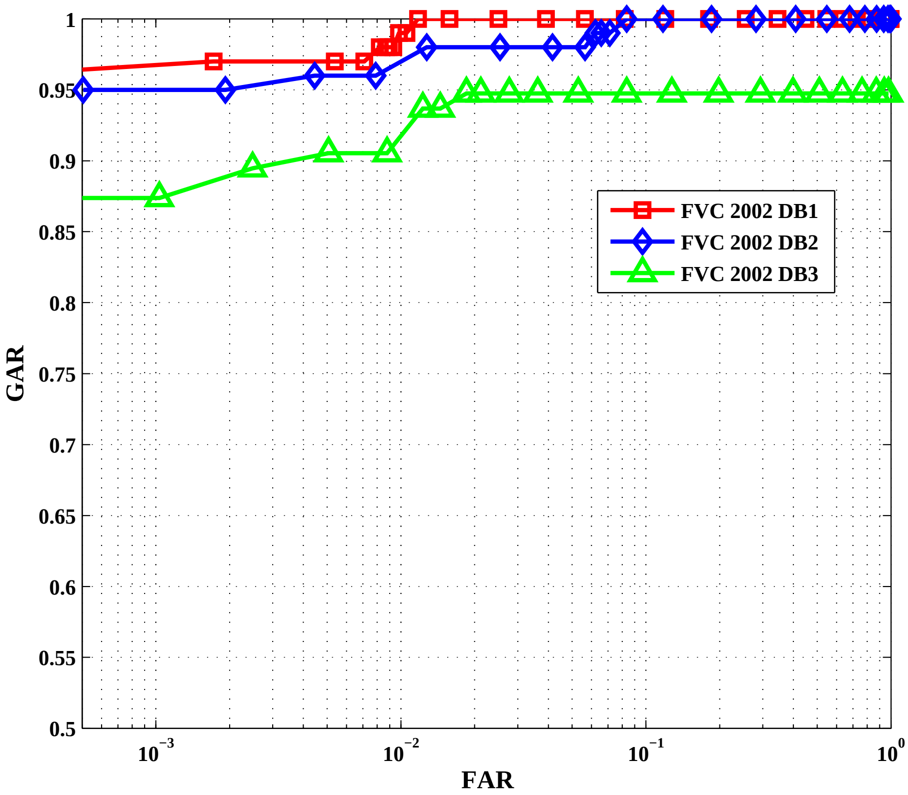


Figure 5.5: ROC curves for BIOCRYP_DNS method for FVC 2002 databases

encoding is also revocable. Hence, for same fingerprint image, we can get many secure templates that are different from one another.

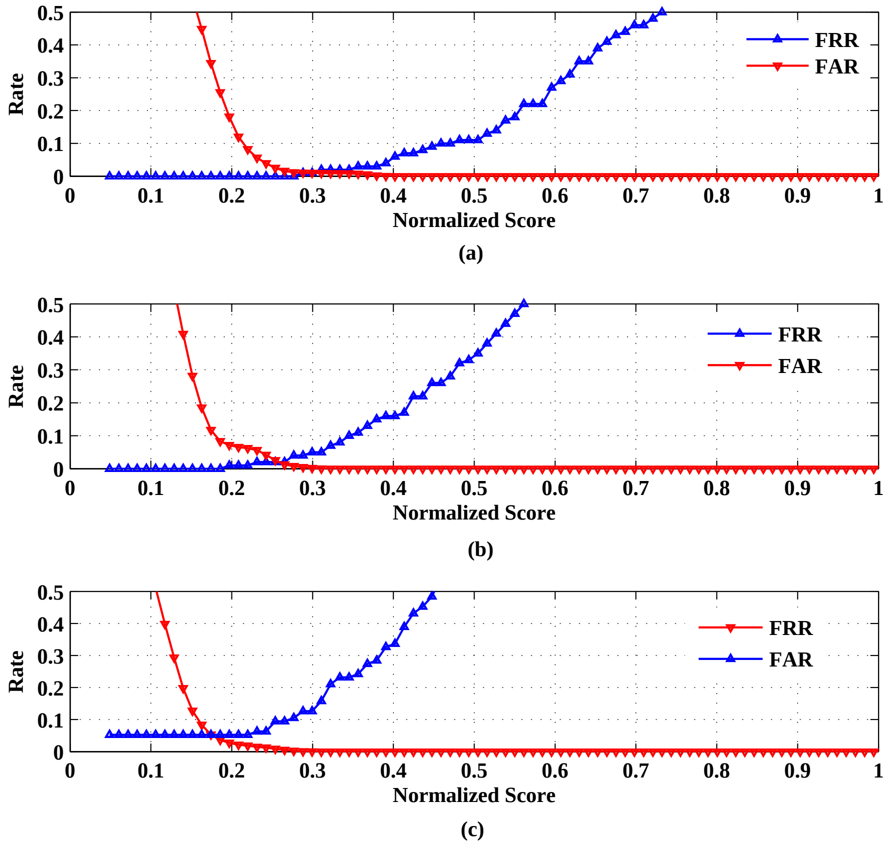
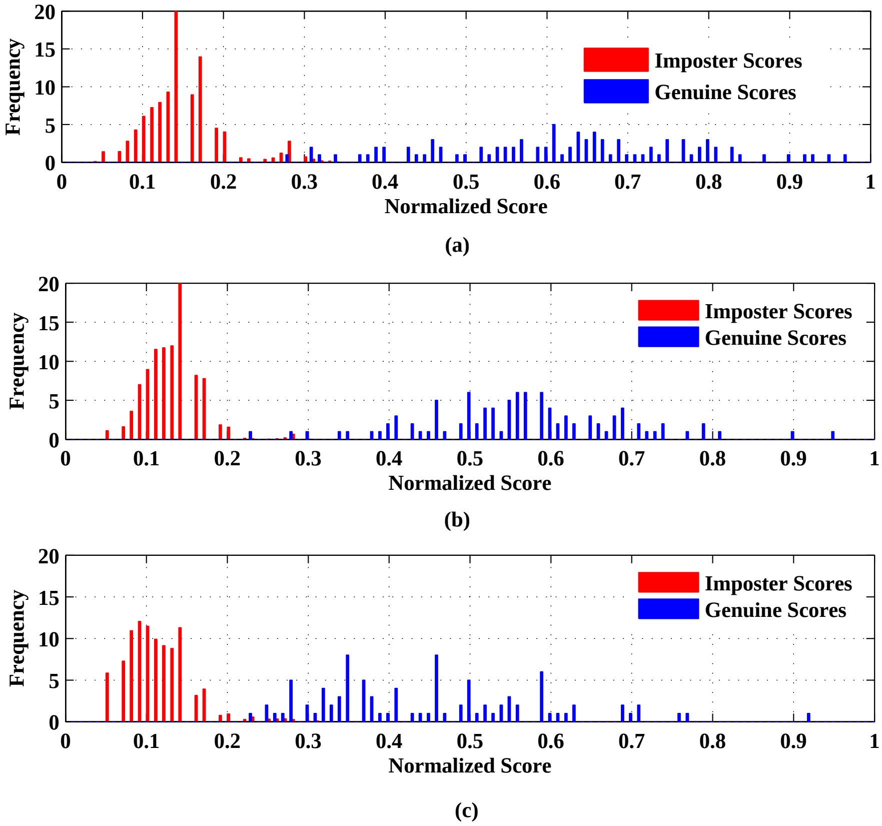


Figure 5.6: FAR/FRR of BIOCRYP_DNS method for FVC 2002 (a)DB1 (b)DB2 (c)DB3 databases

5.1.6.3 Security Analysis

Assume that the intruder obtained information stored in the template of the valid user, *i.e.*, the threshold value that is user-specific, the XOR'd binary vector and the hash value of codeword. It is hard for him to guess the original fingerprint minutiae information because a many-to-one mapping is done to a cell in 3D array after quantization of cells. Also, the Dimensionality reduction matrix discussed in section 5.1.3 is user specific. Hence, the proposed feature generation process is irreversible.

The hill-climbing method described in [197] estimates the minutiae locations by generating synthetic templates. For BIOCRYP_DNS method, it is hard to estimate



(a)

(b)

(c)

Figure 5.7: Genuine and imposter score distribution of BIOCRYP_DNS method for FVC 2002 (a)DB1 (b)DB2 (c)DB3 databases

the minutiae locations since the bit string generated is reduced in size through Dimensionality reduction matrix and hence does not store the location of minutiae points.

5.2 Hybrid technique using Delaunay Neighbor Structures

Cancelability is added to the BIOCRYP_DNS method discussed in preceding section. Instead of directly mapping x coordinate, y coordinate and orientation of minutia point to an array, we transformed DNS by taking distance and orientation difference between reference minutia point (center) and other minutia points in DNS. These transformed features are mapped to a 2D array instead of 3D array. Hence, a hybrid technique that combines cancelable biometrics and bio-cryptosystems was built for securing fingerprint templates. We name the hybrid method as HYBRID_DNS. The framework of HYBRID_DNS method is given in Figure 5.8 contains following steps:

(1) Construct “Delaunay Neighbor Structures (DNS)” from fingerprint minutiae

(2) Transform DNS and generate bit string

(3) Apply Dimensionality reduction matrix to the bit string

(4) Employing fuzzy commitment scheme

(5) Matching

5.2.1 Constructing DNS for fingerprint minutiae

The construction of DNS for each minutia point in the fingerprint image is similar to that of BIOCRYP_DNS method given section 5.1.1.

5.2.2 Transform DNS and generate bit string

The transformed feature vector TFV_r=(d_{jr}, θ_{jr}) for DNS_r is computed as follows:

$$\begin{aligned} \chi &= (x_j - x_r)cos\theta_r + (y_j - yr)sin\theta_r \\ \gamma &= (x_j - x_r)sin\theta_r - (y_j - yr)cos\theta_r \end{aligned} \tag{5.12}$$

$$d_{jr} = \sqrt{\chi^2 + \gamma^2}, j = m_1, m_2, m_3, m_4, m_5, m_6 \tag{5.13}$$

$$\theta_{jr} = \begin{cases} \theta_j - \theta_r & if \theta_j > \theta_r \\ \theta_j - \theta_r + 360 & otherwise \end{cases} \tag{5.14}$$

where d_{jr}, θ_{jr} represents the distance, orientation difference between minutiae point r and neighboring minutia of r in DNS_r, *i.e.*, $j = m_1, m_2, m_3, m_4, m_5, m_6$ for DNS_r shown

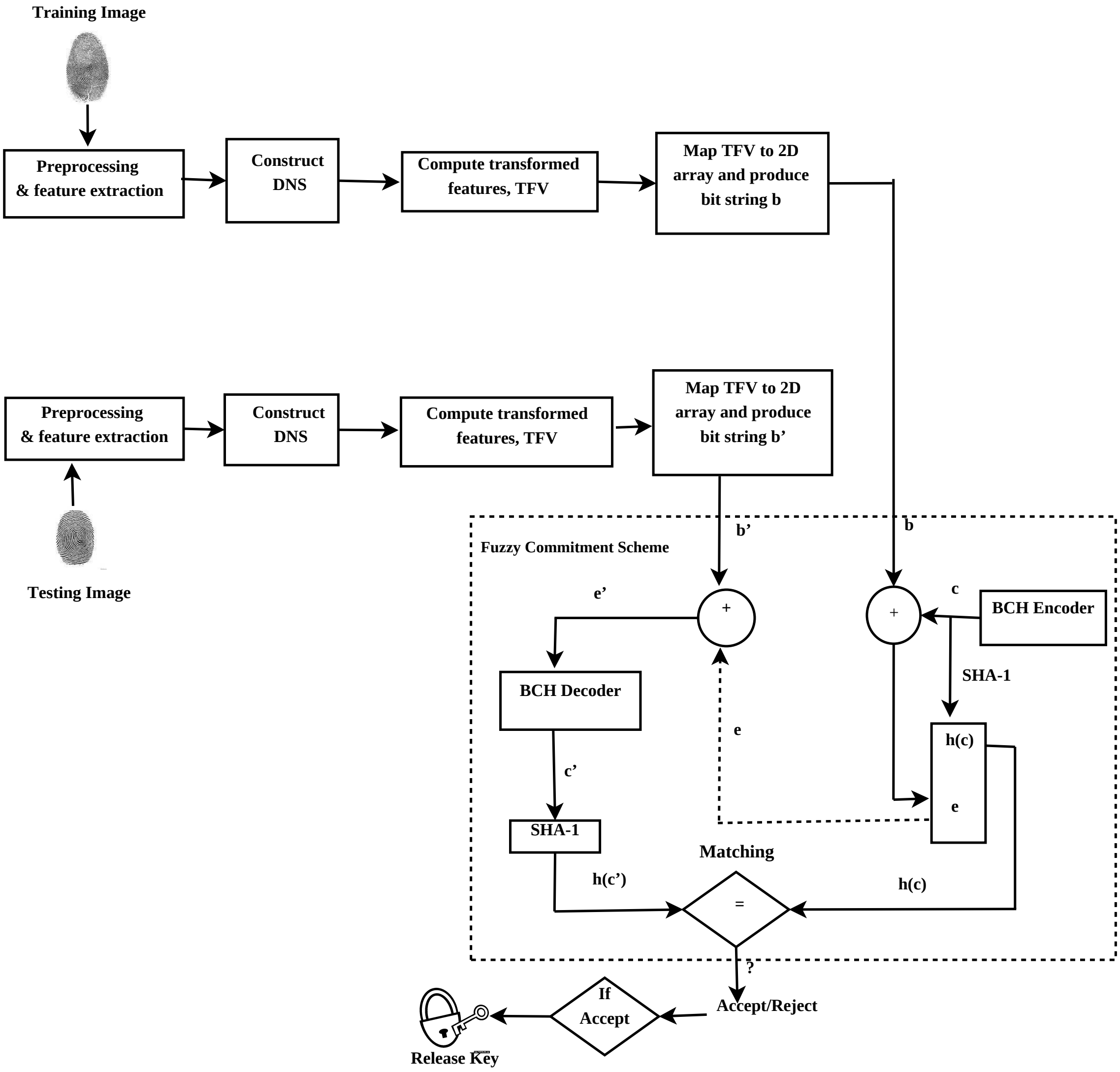

Training Image
Preprocessing & feature extraction
Construct DNS
Compute transformed features, TFV
Map TFV to 2D array and produce bit string b
Preprocessing & feature extraction
Construct DNS
Compute transformed features, TFV
Map TFV to 2D array and produce bit string b'
Testing Image
Fuzzy Commitment Scheme
b'
b
e'
+
+
c
BCH Encoder
BCH Decoder
SHA-1
e
h(c)
e
c'
SHA-1
Matching
h(c')
h(c)
=
?
Accept/Reject
If Accept
Release Key

Figure 5.8: Overall framework of the HYBRID_DNS method

in Figure 5.3(a). The transformed feature vector, say TFV, computed for all DNS is represented as follows:

$$TFV = (TFV_1, TFV_2, TFV_3, ...TFV_t) \tag{5.15}$$

We choose a 2D array and divide it into cells of size c_l, c_h as shown in Figure 5.9. Each TFV_i is mapped onto 2D array after quantization by taking d_{jr} on x-axis that ranges from 0 to $max(d_{jr})$, θ_{jr} on y-axis that ranges from 0 to 2π. Now, we find cells in array

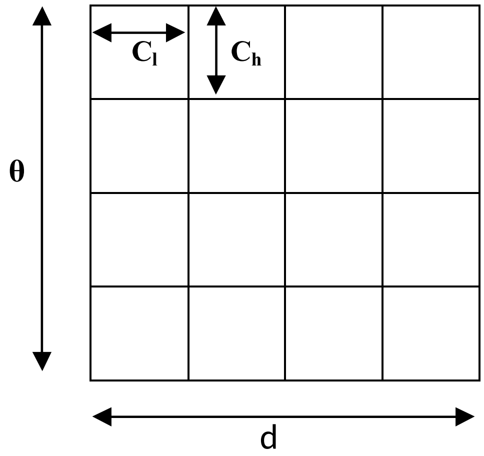


Figure 5.9: 2D array on which TFV_i is mapped

that includes (d_{jr}, θ_{jr}) as follows:

$$\begin{Bmatrix} x_i \\ y_i \end{Bmatrix} = \begin{Bmatrix} \lfloor d_{jr}/c_l \rfloor \\ \lfloor \theta_{jr}/c_h \rfloor \end{Bmatrix} \tag{5.16}$$

where (x_i, y_i) are indices of 2D array. If a point or more than one point falls in a cell its value is set to 1 otherwise 0. Visiting the cells in a sequential manner produce a bit string. Similarly, for t TFVs computed in equation 5.15 for a given fingerprint image, t bit strings are produced.

5.2.3 Apply Dimensionality reduction matrix to the bit string

The application of dimensionality reduction matrix to the bit string in HYBRID_DNS method is similar to BIOCRYP_DNS method discussed in section 5.1.3.

5.2.4 Employing fuzzy commitment scheme

The BCH encoding and decoding in fuzzy commitment scheme is applied to the bit string of HYBRID_DNS method similar to that of BIOCRYP_DNS discussed in Section 5.1.4.

5.2.5 Matching

The matching process of HYBRID_DNS method is similar to the matching process of BIOCRYP_DNS discussed in Section 5.1.5. Algorithm 5.2 describes the HYBRID_DNS method.

Algorithm 5.2 HYBRID_DNS ($x_i, y_i, \theta_i, A_1, c_l, c_h, n, k, \lambda$): To compute Transformed DNS and generate secure template

Input: Minutiae locations (x_i, y_i)
Orientation of minutiae points θ_i
Predefined 2D Array A_1 divided into cells of size c_l, c_h
Input to BCH encoder n, k, λ
Output: Normalized match score

```
begin
t ← number of minutiae points in the fingerprint image;        // Refer to Figure 5.2
Construct Delaunay triangulation net from the fingerprint minutia;
Initialize minutia counter $i \leftarrow 1$;
while $i \leq t$ do
//Find neighbor minutiae in Delaunay triangulation net for each minutia point-Refer to Figure 5.3(a)
        $DNS(i) \leftarrow$ Neighboring minutiae of i in Delaunay triangulation net;
        $r \leftarrow$ center of $DNS(i)$;
// Transform $DNS(i)$
        $p \leftarrow$ number of neighboring minutiae in $DNS(i)$;
        while $j \leq p$ do
//Calculate distance from center minutia r to $j^{th}$ neighbor in $DNS(i)$
                $\chi = (x_j - x_r)cos\theta_r + (y_j - yr)sin\theta_r$;
                $\gamma = (x_j - x_r)sin\theta_r - (y_j - yr)cos\theta_r$;
                $d_{jr} = \sqrt{\chi^2 + \gamma^2}$;
//Calculate orientation difference center minutia r and $j^{th}$ neighbor in $DNS(i)$
                if $(\theta_j - \theta_r) > 0$
                        $\theta_{jr} = \theta_j - \theta_r$;
```

Algorithm 5.2 HYBRID_DNS (continued)

```
15:                 else
16:                     $\theta_{jr} = \theta_j - \theta_r + 360$;
17:                 end if
18:                 $TFV(i,:) \leftarrow (d_{jr}, \theta_{jr})$;
19:                 $j \leftarrow j + 1$;
20:         end while;
21:         Quantize $TFV(i)$ by cell sizes $c_l, c_h$, map to 2D array $A_1$;
22:         Generate bit string $B_s$;                     //Refer to section 5.1.2.
  // Apply Dimensionality reduction matrix to bit string-Refer to section 5.1.3
23:         $m \leftarrow$ length($B_s$);
24:         Define random matrix R of size (m,n) filled with 0's and 1's;
25:         $b \leftarrow B_s \times R$;
26:         Codeword $c \leftarrow BCH(n,k,\lambda)$;                //Refer to section 5.1.4
27:         $e \leftarrow b \oplus c$;
  //Hash value generation
28:         $h \leftarrow$ SHA-1(c);
  // Store encrypted template e
29:         $Encrypt\_temp(i) \leftarrow e$;
30:         $i \leftarrow i + 1$;
31: end while
32: Match score generation;                                //Refer to section 5.1.5
33: return match_score;
34: end
```

Computational complexity of HYBRID_DNS algorithm
Delaunay triangulation formation from t minutiae points: $O(t)$
Steps 9 to 20 repeated p times: $O(p)$
Steps 5 to 31 repeated t times: $O(t)$
Algorithm Complexity: $O(t + t * p) = O(t * p)$

5.2.6 Experimental Results and Discussion

We tested HYBRID_DNS technique by tuning the cell sizes c_l,c_h in the quantization step, and different values of (n,k,λ) in BCH encoding.

5.2.6.1 Accuracy

Table 5.2 refers to the EER acquired for HYBRID_DNS method on “FVC 2002 databases”. An EER of 1.15%, 1.49% and 5.81% was reported for “FVC 2002 DB1, DB2 and DB3” respectively for cell sizes $c_l = 10$,$c_h = 10$ and “(1023,11,255) BCH encoder”. Table 5.3 refers to the EER acquired for HYBRID_DNS method on “FVC 2004 databases”. An EER of 10.76%, 11.54% and 15.21% was reported for “FVC 2004 DB1, DB2 and DB3” respectively for cell sizes $c_l = 15$,$c_h = 15$ and “(1023,11,255) BCH encoder”.

Cell Sizes		**(n,k,λ)**								
		(255,63,33)			**(511,259,30)**			**(1023,11,255)**		
c_l	c_h	**DB1**	**DB2**	**DB3**	**DB1**	**DB2**	**DB3**	**DB1**	**DB2**	**DB3**
5	5	3.54	4.03	12.75	5.48	4.17	10.38	2.15	3.17	10.87
10	**10**	4.53	5.03	11.75	6.38	5.27	11.31	**1.15**	**1.49**	**5.81**
10	15	4.76	6.61	13.82	4.13	6.12	15.79	2.39	2.98	11.86
15	15	3.29	4.98	8.89	4.74	3.17	8.89	2.43	2.79	6.99
20	15	5.14	7.63	10.43	4.63	4.93	14.96	1.89	2.12	6.89
20	20	6.57	5.89	12.43	5.74	7.63	15.82	2.98	3.02	7.16
25	25	6.54	6.09	11.53	4.52	6.21	13.18	3.45	4.07	8.03

Table 5.2: EER resulted for HYBRID_DNS method on FVC 2002 databases

Figure 5.10 depicts the ROC curves of HYBRID_DNS method. From Figure 5.10, it is noticed that the FAR and FRR are low for FVC 2002 DB1 and DB2, that results in a low EER. Figure 5.11 represents the error rates (FAR and FRR) acquired for the HYBRID_DNS method on the databases in FVC 2002 and 2004. From Figure 5.11, it is observed that a low EER is reported for FVC 2002 DB1 and DB2. Figure 5.12 depicts the genuine and imposter score distribution for HYBRID_DNS method on FVC 2002 and 2004 databases.

Cell Sizes		**(n,k,λ)**								
		(255,63,33)			**(511,259,30)**			**(1023,11,255)**		
c_l	c_h	**DB1**	**DB2**	**DB3**	**DB1**	**DB2**	**DB3**	**DB1**	**DB2**	**DB3**
5	5	23.31	17.76	22.13	17.31	18.72	20.17	12.51	14.63	20.32
10	10	14.35	15.32	21.75	16.83	15.72	21.13	11.51	11.94	15.91
10	15	14.67	16.16	23.28	14.31	16.21	15.97	12.93	12.89	19.64
15	**15**	3.29	4.98	8.89	4.74	3.17	8.89	**10.76**	**11.54**	**15.21**
20	15	15.25	17.74	20.54	14.74	14.04	17.07	11.90	12.23	26.90
20	20	16.68	15.90	22.54	15.85	17.74	25.93	12.09	19.93	27.27
25	25	16.74	16.79	21.63	14.62	16.32	23.29	13.56	14.18	28.84

Table 5.3: EER resulted for HYBRID_DNS method for FVC 2004 databases

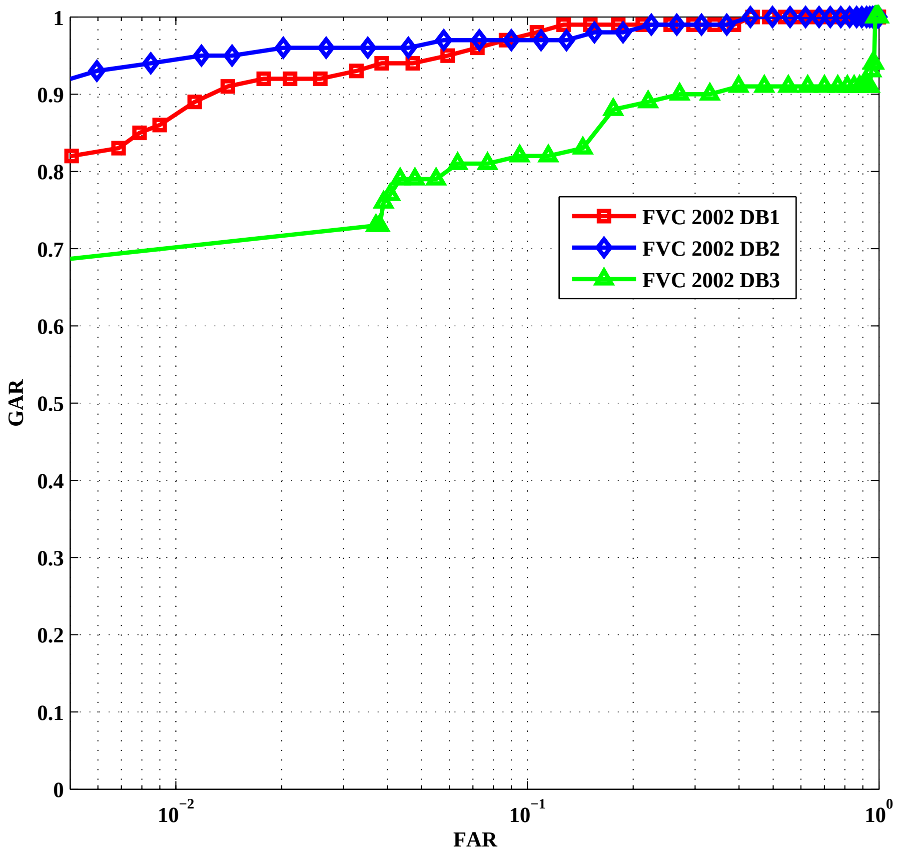


Figure 5.10: ROC curves of HYBRID_DNS method for FVC 2002 databases

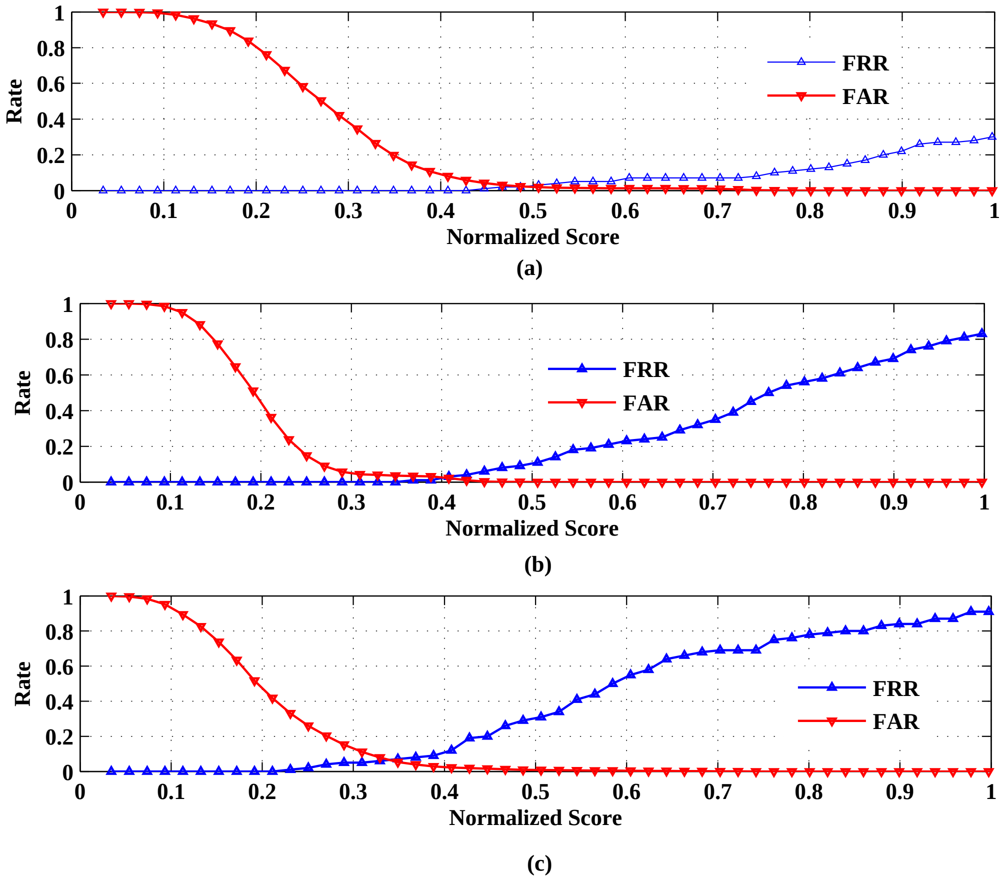
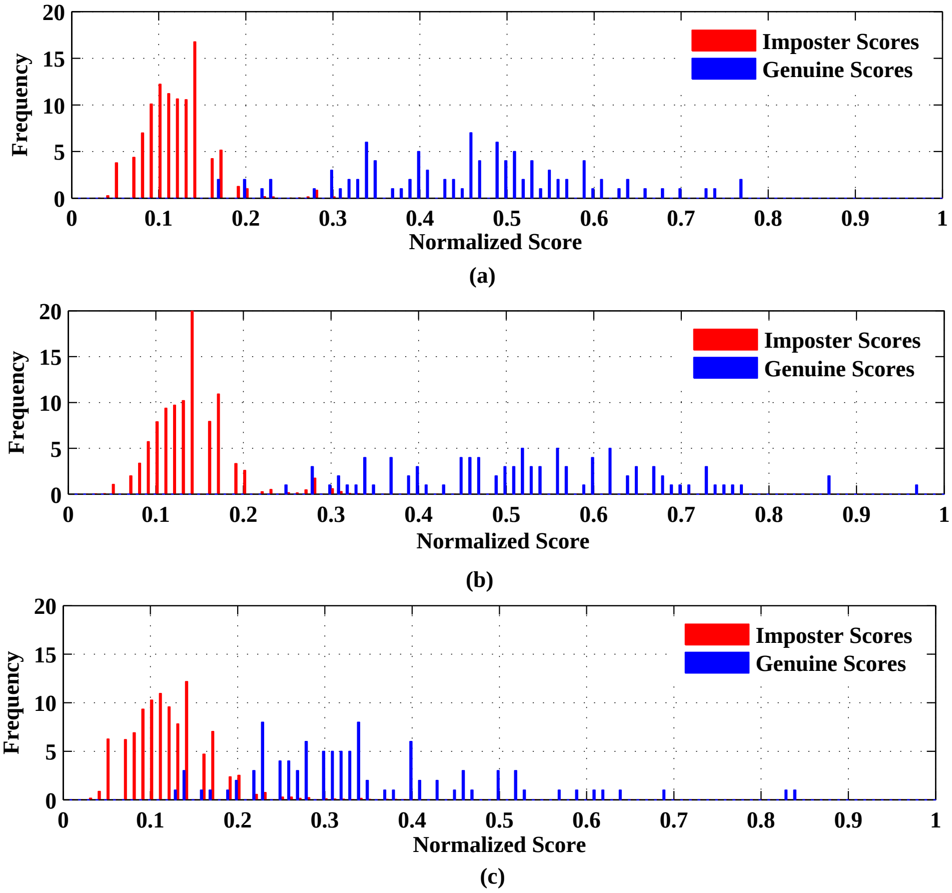


Figure 5.11: FAR/FRR of HYBRID_DNS method for FVC 2002 (a)DB1 (b)DB2 (c)DB3

The d' values obtained for HYBRID_DNS method for the optimal result shown in tables 5.2 and 5.3 is shown in Figure 5.13. From the figure, it is noticed that d' value is more for FVC 2002 DB1, which illustrates that the separation between "genuine scores and imposter scores" is more. Figure 5.14 shows the KS test values acquired for the HYBRID_DNS method. From the figure, we can observe that the KS test values are closer to 1 for FVC 2002 DB1 and DB2.

5.2.6.2 Revocability and Security Analysis

The discussion given for revocability and security analysis for BIOCRYP_DNS method in section 5.1.6.2 and 5.1.6.3 respectively is also applicable for HYBRID_DNS method.



(a)

(b)

(c)

Figure 5.12: Genuine and Imposter score distribution of HYBRID_DNS method for FVC 2002 (a)DB1 (b)DB2 (c)DB3

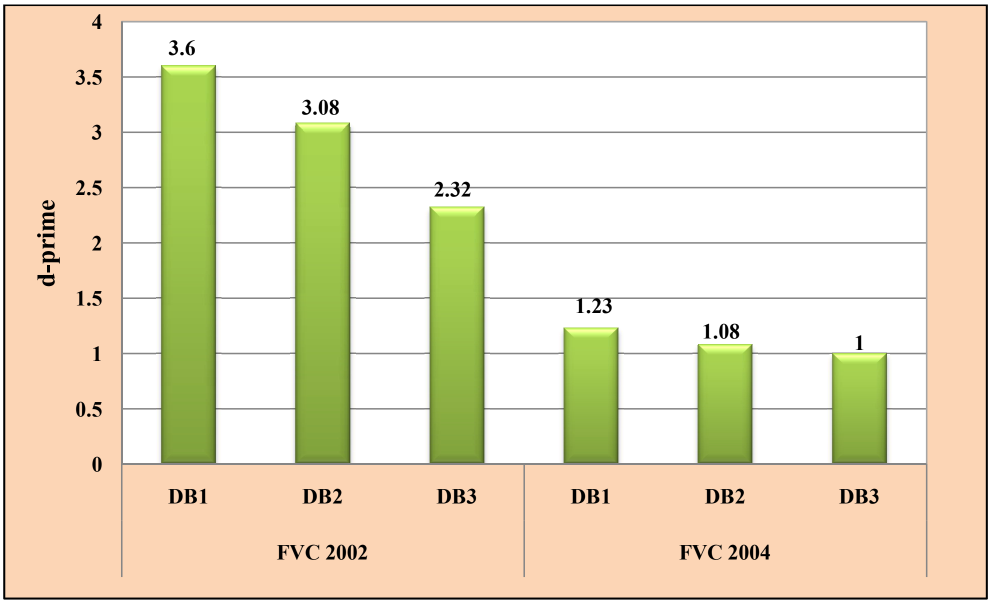


Figure 5.13: d' values for the HYBRID_DNS method on FVC 2002 and 2004 databases

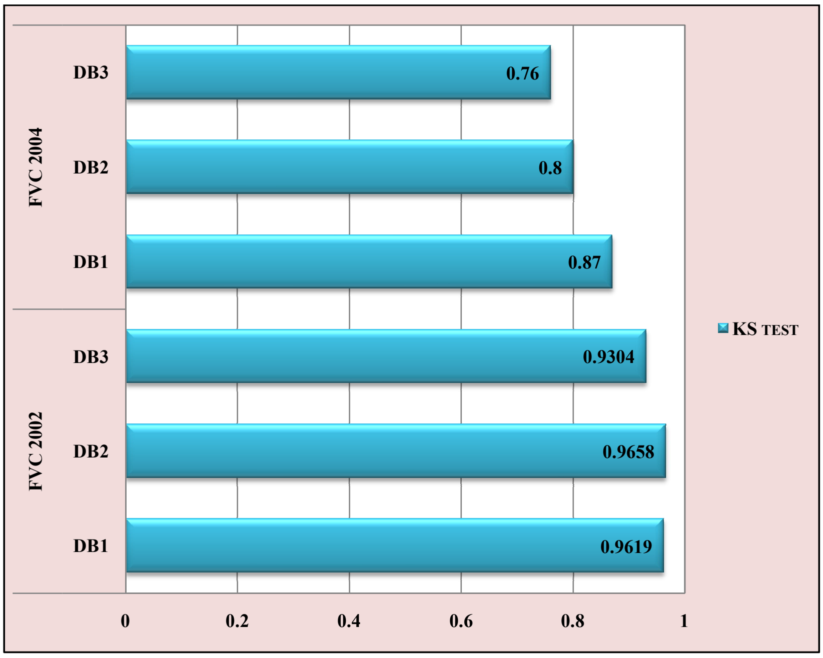


Figure 5.14: KS test values for HYBRID_DNS method on FVC 2002 and 2004 databases

5.3 Hybrid technique using Multiple Spiral Curves

We name the hybrid technique for securing fingerprint templates using multiple spiral curves as HYBRID_MSC. The overall flow of HYBRID_MSC method is given in Figure 5.15. It contains following steps:

(1) Construction of multiple spiral curves

(2) Bit string representations of multiple spiral curves

(3) Apply Dimensionality reduction matrix to the bit string

(4) Employing fuzzy commitment scheme

(5) Matching

5.3.1 Construction of multiple spiral curves

Treat each minutia point in the fingerprint image as reference minutia (x_r, y_r, θ_r) and perform the following steps:

(1) Compute the distance between reference minutia and remaining minutiae points in the image.

$$\begin{bmatrix} x_i^T \\ y_i^T \end{bmatrix} = \begin{bmatrix} cos\theta_r & -sin\theta_r \\ sin\theta_r & cos\theta_r \end{bmatrix} \begin{bmatrix} (x_i - x_r) \\ -(y_i - y_r) \end{bmatrix} \tag{5.17}$$

From x_i^T and y_i^T, distance is calculated as

$$d_{ir} = \sqrt{{x_i^T}^2 + {y_i^T}^2} \tag{5.18}$$

(2) For a reference minutiae point r, sort d_{ir} in ascending order. Now, we take m least distances from the list. With these distances, contiguous right angled triangles are drawn taking distances as hypotenuses of triangles as in [113]. For the first triangle, the base distance d_0 is chosen and the same is added to every distance as shown in Figure 5.16(a). Here, d_0 acts as a key.

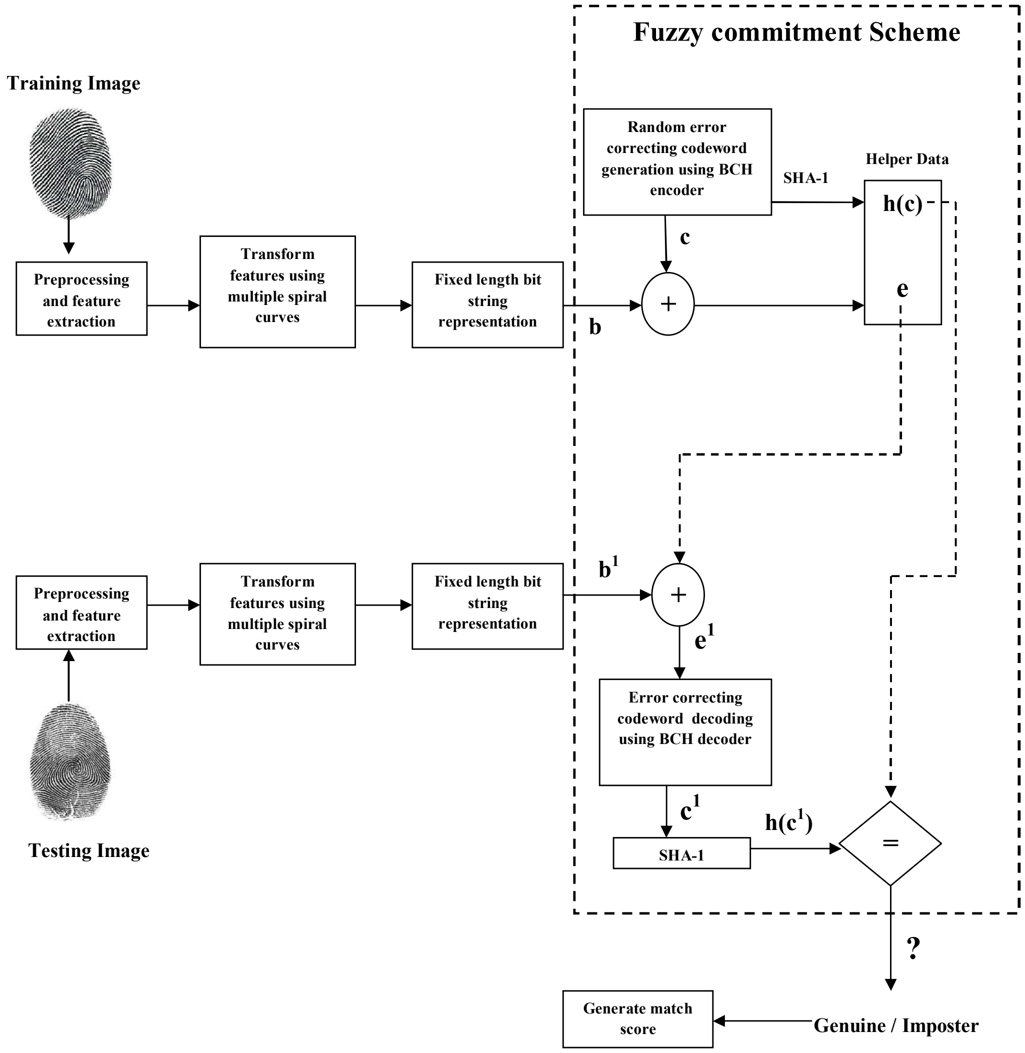
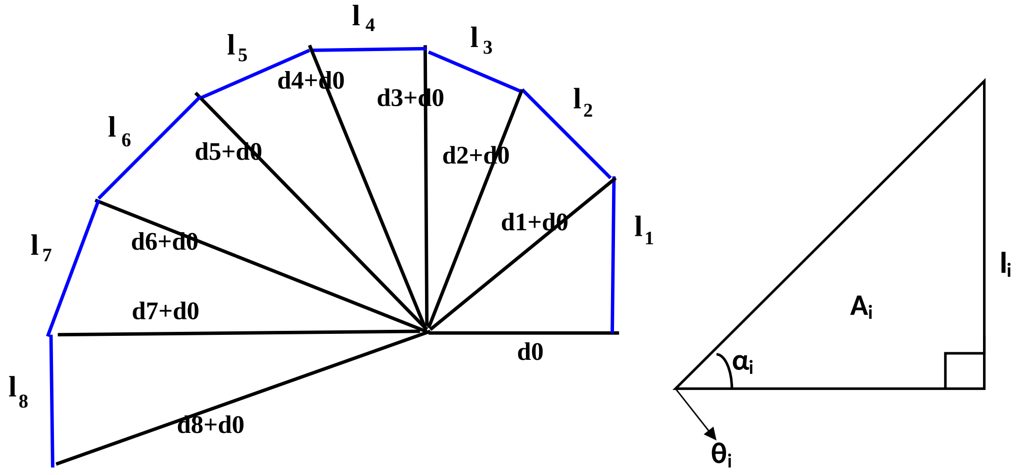

Fuzzy commitment Scheme
Training Image
Preprocessing and feature extraction
Transform features using multiple spiral curves
Fixed length bit string representation
Random error correcting codeword generation using BCH encoder
SHA-1
Helper Data
h(c)
c
b
+
e
b1
e1
Testing Image
Error correcting codeword decoding using BCH decoder
c1
h(c1)
SHA-1
=
?
Generate match score
Genuine / Imposter

Figure 5.15: Flow diagram of HYBRID_MSC method

(3) Pythagoras theorem is used to compute the other side of the triangle, l_i from Figure 5.16(a). Now, we store l_i, the angle at vertex α_i, area A_i of each triangle, and the orientation of respective minutiae θ_i for deriving a feature set as shown in Figure 5.16(b).

(4) Hence, for a reference minutia(x_r, y_r, θ_r), the feature set is represented as

$$\begin{aligned} L_r = [[l_{1r}, \theta_1, \alpha_{1r}, A_{1r}], [l_{2r}, \theta_2, \alpha_{2r}, A_{2r}], \\ \dots[l_{mr}, \theta_m, \alpha_{mr}, A_{mr}]] \end{aligned} \tag{5.19}$$



(a) (b)

Figure 5.16: (a)Spiral curve constructed for a reference minutia point(b)The metrics considered for a triangle to build the feature set L_r

where $l_{ir}, \alpha_{ir}, A_{ir}$ represents the metrics of a triangle i, θ_i represents the orientation of minutia i, m is the number of least distances considered.

(5) Repeat steps 1 to 4 for all minutiae points in the fingerprint image. Thus, the transformed template is represented as $L = [L_1, L_2, L_3, ..., L_t]$, t represents the minutiae count.

The transformed feature set stores the features derived from the contiguous right angled triangles and original orientation of the minutiae points. Thus, it is infeasible to find the location of minutiae points from the transformed feature set L, which in turn ensures the non-invertibility of transformed template.

5.3.2 Bit string representations of multiple spiral curves

Each $L_r = (l_{ir}, \theta_i, \alpha_{ir}, A_{ir})$ is a vector of order 4. L_r is mapped to a 4D-space. The 4 axes are taken as distance that ranges in [0, $max(d_{ir})$], orientation ranging in [0. 360], angle ranging in [0 90] and area ranging in [0, $max(A_i)$] respectively. The 4D-space is quantized along 4-axes by using cell sizes c_x, c_y, c_z, c_w [94]. The transformed template L_r is projected to 4D space. Now, we find the cells that contain the points of L_r as follows:

$$\begin{bmatrix} x_i \\ y_i \\ z_i \\ w_i \end{bmatrix} = \begin{bmatrix} \lfloor d_{ir}/c_x \rfloor \\ \lfloor \theta_r/c_y \rfloor \\ \lfloor \alpha_{ir}/c_z \rfloor \\ \lfloor A_{ir}/c_w \rfloor \end{bmatrix} \tag{5.20}$$

where x_i, y_i, z_i and w_i represents indices of 4D-space. Now, if one or more points of L_r falls in cell a cell of 4D-space its value is taken as 1 else 0. Visit the cells of 4D-space sequentially to obtain a fixed-length bit string. The process is repeated for t L_rs in L. Hence t bit strings are obtained.

5.3.3 Apply Dimensionality reduction matrix to the bit string

The application of dimensionality reduction matrix to the bit string in HYBRID_MSC method is similar to BIOCRYP_DNS method discussed in section 5.1.3.

5.3.4 Employing fuzzy commitment scheme

The BCH encoding and decoding in fuzzy commitment scheme is applied to the bit string of HYBRID_MSC method similar to BIOCRYP_DNS discussed in Section 5.1.4.

5.3.5 Matching

The matching process of HYBRID_MSC method is similar to the matching process of BIOCRYP_DNS discussed in Section 5.1.5. Algorithm 5.3 describes the HYBRID_MSC method.

5.3.6 Experimental Results and Discussion

5.3.6.1 Accuracy

We tested HYBRID_MSC method by tuning the parameters like number of least distances considered for drawing spiral curves m, cell sizes c_x,c_y, c_z and c_w in the quantization step and different values of (n,k,λ) in BCH encoding as shown in Table 5.4. For cell sizes $c_x = 20$,$c_y = 25$, $c_z = 20$, $c_w = 150$, the HYBRID_MSC method shown

Algorithm 5.3 HYBRID_MSC $(x_i, y_i, \theta_i, S_1, c_x, c_y, c_z, c_w, n, k, \lambda)$: To compute Multiple Spiral Curves and generate secure template

```
Input:   Minutiae locations $(x_i, y_i)$
          Orientation of minutiae points $\theta_i$
          Predefined 4D Space $S_1$ divided into cells of size $c_x, c_y, c_z, c_w$
          Input to BCH encoder $n, k, \lambda$
Output: Normalized match score

begin
t ← number of minutia in the fingerprint image;
Initialize minutia counter $i \leftarrow 1$;
while $i \leq t$ do
        $r \leftarrow i$;                     // Consider each minutia point as reference minutia
//Calculate distance from reference minutia r to $i^{th}$ neighbor in $DNS(i)$
        $\chi = (x_i - x_r)cos\theta_r + (y_i - y_r)sin\theta_r$;
        $\gamma = (x_i - x_r)sin\theta_r - (y_i - y_r)cos\theta_r$;
        $d_{ir} = \sqrt{\chi^2 + \gamma^2}$;
        Sort $d_{ir}$ in ascending order and take $m$ least distances;
// Build right angled triangles - Refer to section 5.3.1 and Figure 5.16(a)
        Calculate $lri, \alpha_{ir}, A_{ir}$
        $L(i,:) \leftarrow (l_{ir}, \theta_r, \alpha_{ir}, A_{ir})$;
        Quantize $L(i)$ by cell sizes $c_x, c_y, c_z, c_w$, map to 4D space $S_1$;
        Generate bit string $B_s$;                       //Refer to section 5.1.2.
// Apply Dimensionality reduction matrix to bit string       //Refer to section 5.1.3.
        $m \leftarrow$ length($B_s$);
        Define random matrix $R$ of size (m,n) filled with 0's and 1's;
        $b \leftarrow B_s \times R$;
        Codeword $c \leftarrow BCH(n, k, \lambda)$;                 //Refer to section 5.1.4
        $e \leftarrow b \oplus c$;
//Hash value generation
        $h \leftarrow$ SHA-1(c);
// Store encrypted template e
        $Encrypt\_temp(i) \leftarrow e$;
```

Algorithm 5.3 HYBRID_MSC (continued)

```
21:         $i \leftarrow i+1$;
22: end while
23: Match score generation                                  //Refer to section 5.1.5
24: return match_score
25: end
```

Computational Complexity of HYBRID_MSC algorithm

Steps 4 to 22 repeated t times: $O(t)$

Sorting t distances: $O(t)$

Algorithm Complexity: $O(t*t) = O(t^2)$

m & Cell Sizes					**(n,k,λ)**								
					(127,57,11)			**(255,63,33)**			**(1023,11,255)**		
m	c_x	c_y	c_z	c_w	**DB1**	**DB2**	**DB3**	**DB1**	**DB2**	**DB3**	**DB1**	**DB2**	**DB3**
3	15	15	15	100	4.74	5.76	13.78	2.08	2.85	10.43	3.74	3.63	11.43
3	20	25	20	150	4.91	6.43	11.98	3.88	2.32	10.67	2.70	2.83	10.65
4	15	15	15	100	3.98	4.67	10.75	2.83	3.14	8.63	2.89	3.13	16.34
4	**20**	**25**	**20**	**150**	**2.72**	**3.68**	**8.34**	**1.17**	**2.46**	**8.51**	**1.52**	**2.35**	**10.43**
5	10	20	10	100	3.43	4.16	11.02	3.02	4.48	12.75	6.41	3.68	14.76
5	20	25	20	200	5.71	7.12	17.69	6.45	5.34	11.47	7.43	6.23	16.51

Table 5.4: EER for HYBRID_MSC method on FVC 2002 databases

optimal result, *i.e.*, an EER of 1.17%, 2.46% and 8.51% for FVC 2002 DB1 through DB3 for (255,63,33) BCH encoder.

Figures 5.17 depicts the ROC curves for the HYBRID_MSC method. From Figure 5.17, it can be seen that FVC 2002 DB2 has better recognition rate compared to DB1 and DB3. Some fingerprint images may not have singular points. Hence, instead of singular points [113], we used all minutiae points as a center for constructing spiral curves. To reduce complexity, we considered a set of least distances m as shown in Table 5.4. Figure 5.18 plots the FAR/FRR obtained for HYBRID_MSC method on FVC 2002 databases.

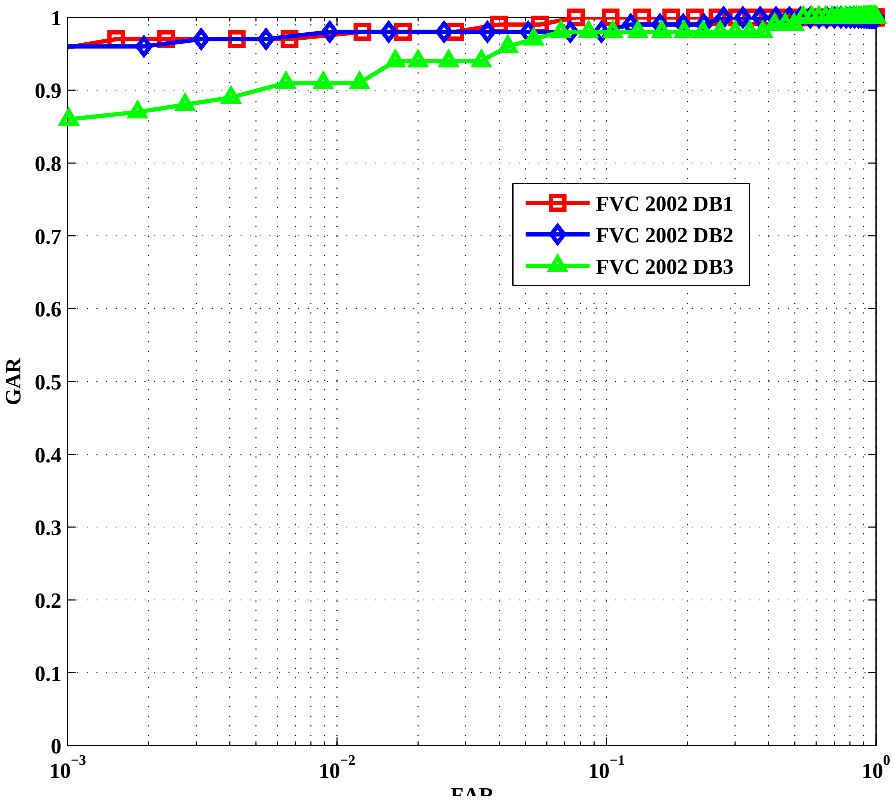
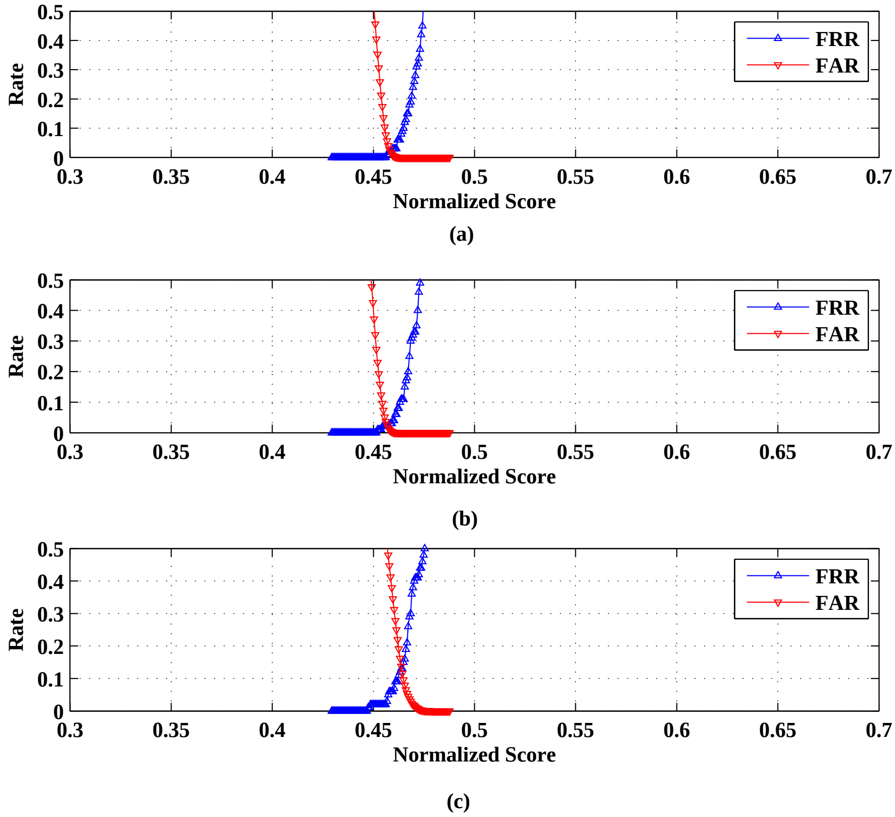


Figure 5.17: ROC curves of HYBRID_MSC method on FVC 2002 databases

The genuine and imposter score distribution of HYBRID_MSC method is shown in Figure 5.19. The figure shows a clear separation between genuine and imposter distribution. Figure 5.20 shows the d' values obtained for HYBRID_MSC method. It can be noticed that d' value is better to DB2 than DB1 and DB3 of FVC 2002. From Figure 5.21, it is observed that KS test value is closer to 1 for DB2 than DB1 and DB3 of FVC 2002.

5.3.6.2 Revocability

Using different base distances d_0, we generate transformed templates for a fingerprint image in FVC 2002 DB2. These transformed templates are matched with the enrolled template. Hence, pseudo-imposter scores are generated. The pseudo-imposter distribution is plotted in Figure 5.22. It is clearly separated from genuine distribution and



(a)

(b)

(c)

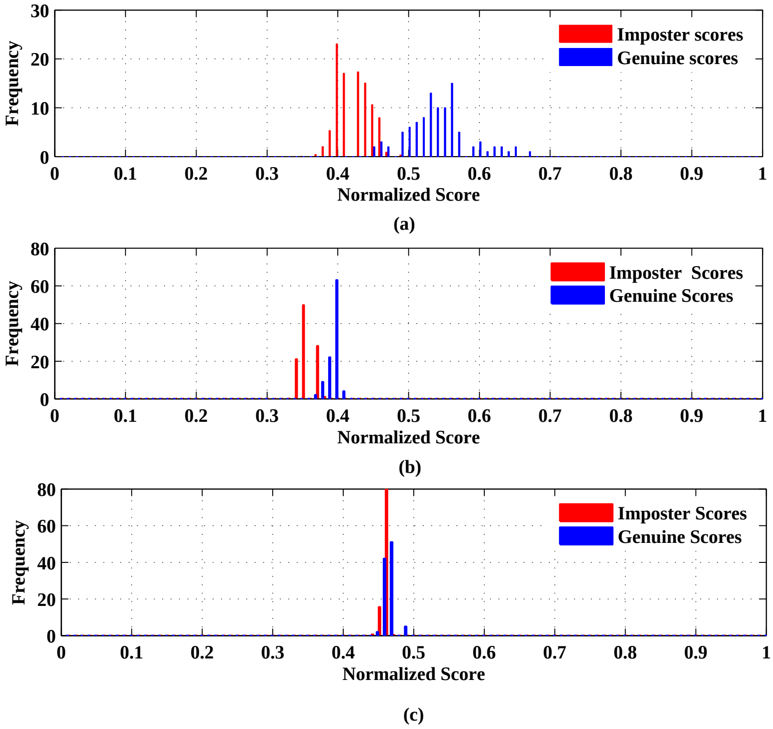
Figure 5.18: FAR/FRR of HYBRID_MSC method for FVC 2002 databases (a) DB1 (b) DB2 (c) DB3

close to imposter distribution.

5.3.6.3 Security Analysis

We considered the following cases for performing security analysis of HYBRID_MSC method

- **Case 1:** Assume attacker obtained information stored in the template of a valid user, *i.e.*, user-specific least distances considered *m*, the hash value of codeword, the XOR'd binary vector. From these values, it is hard to guess the minutia location because of many-to-one mapping to a cell in the 4D-space after quantization. Hence, the proposed feature generation process is irreversible.



(a)

(b)

(c)

Figure 5.19: Score distribution for HYBRID_MSC method on FVC 2002 databases (a) DB1 (b) DB2 (c) DB3

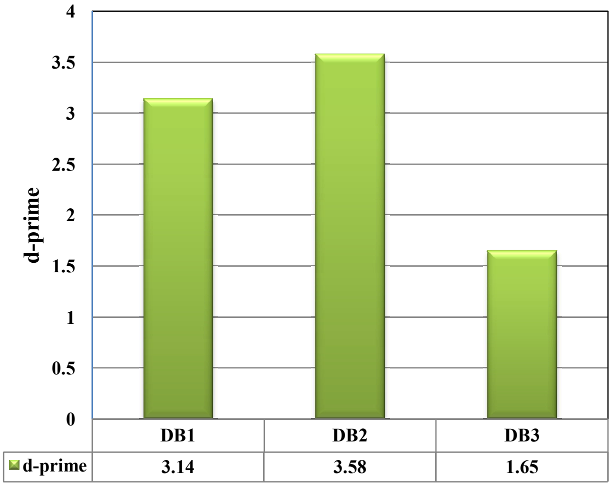


Figure 5.20: d' values of HYBRID_MSC method for databases of FVC 2002

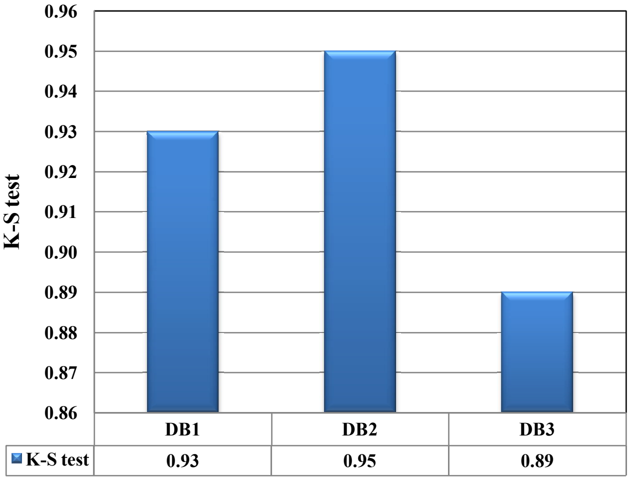


Figure 5.21: KS test values of HYBRID_MSC for databases of FVC 2002

- **Case 2:** The hill-climbing method described in [197] estimates the minutiae locations by generating synthetic templates. For HYBRID_MSC method, it is hard to estimate the minutiae locations since the bit string generated is reduced in size through Dimensionality reduction matrix and hence does not store the location of minutiae points.

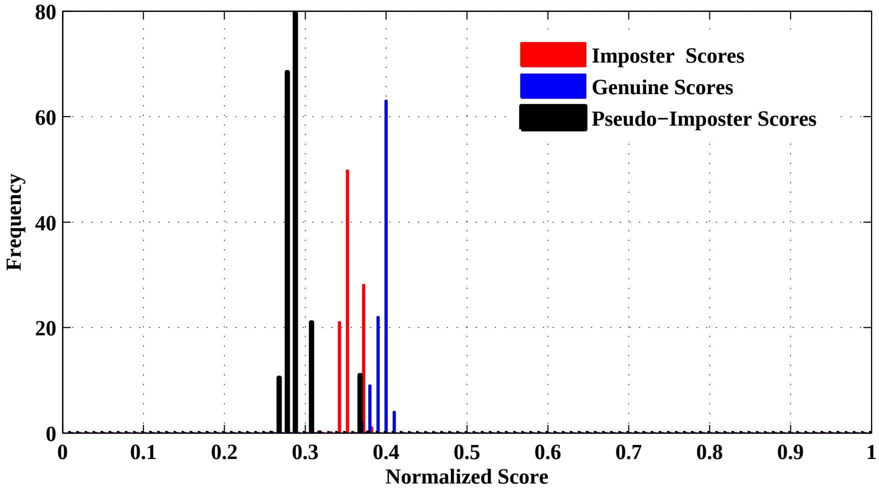


Figure 5.22: Genuine, Imposter, and Pseudo-imposter score distribution of HYBRID_MSC method on FVC 2002 DB2

5.4 Hybrid technique using Convolution Coding

We name the hybrid technique built using convolution coding for securing fingerprint templates as HYBRID_CC. The flow diagram of HYBRID_CC method is shown in Figure 5.23 containing the following steps:

(1) Compute transformed features from "Delaunay triangulation net" formed from fingerprint minutiae points.

(2) Represent the transformed features as bit string

(3) Employ "fuzzy commitment scheme" by using convolution coding.

(4) Matching.

5.4.1 Compute transformed features using Delaunay triangulation net and generate bit string

The computation of transformed features in HYBRID_CC method is similar to DTFS_INCIR algorithm discussed in chapter 3. We use the same transformed features and generate the bit string.

5.4.2 Employ fuzzy commitment scheme by using convolution coding

A convolution code is an error correcting code represented by (n,k,m), where *n* is a number of output bits, *k* is the number of input bits and *m* is the number of memory registers. Convolution code generates parity symbols by the sliding application of a boolean polynomial function to a bit stream [220]. The decoding of convolution codes can be done by sequential decoding and maximum likelihood decoding. Viterbi algorithm is the best-known implementation of maximum likelihood decoding. Figure 5.24 shows a (3,1,3) convolution encoder.

We employ the fuzzy commitment scheme [27] discussed in Section 5.1.4 using convolution encoder. A viterbi decoding algorithm is used in the decoding attempt of the codeword.

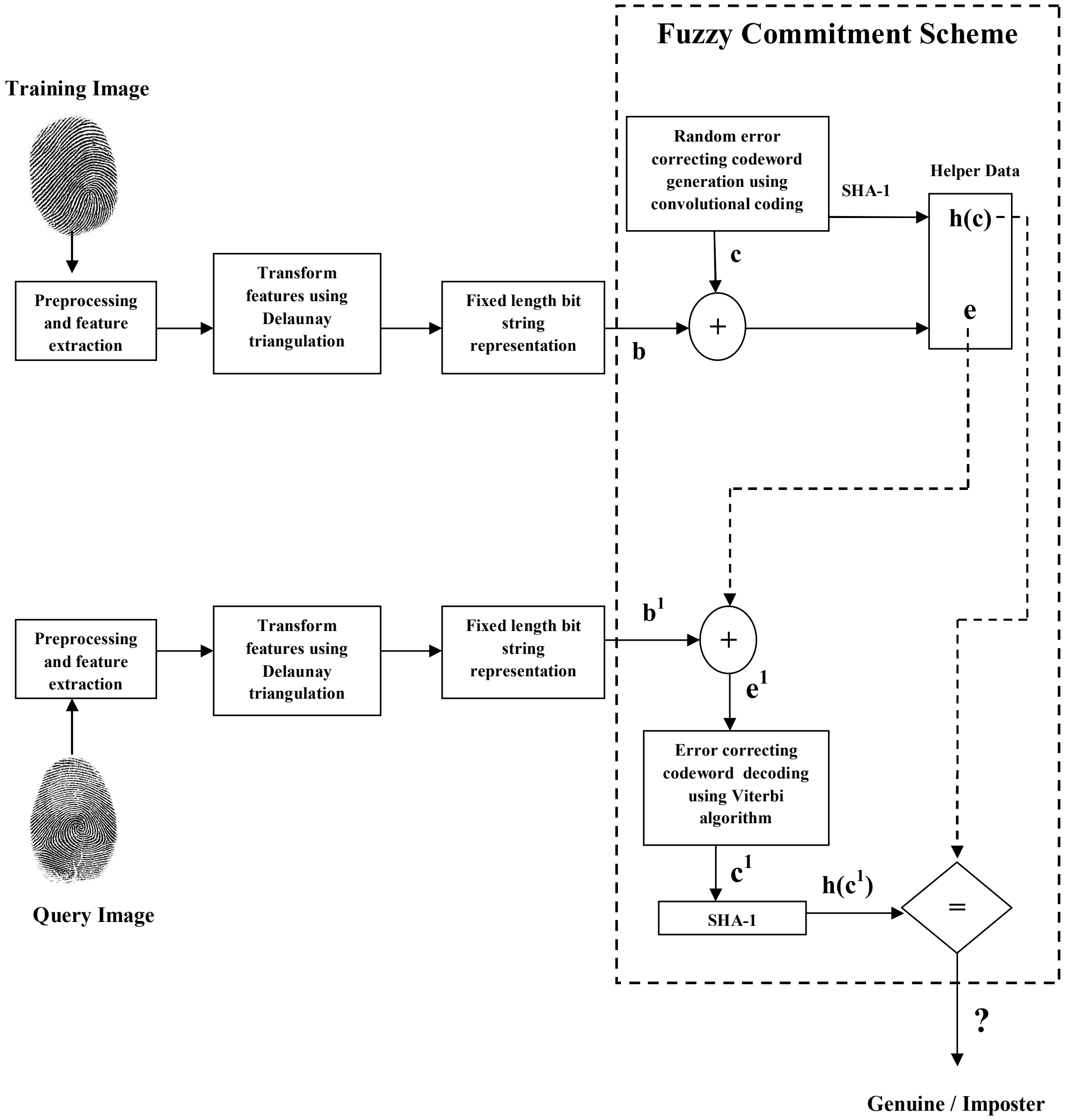

Fuzzy Commitment Scheme
Training Image
Random error correcting codeword generation using convolutional coding
SHA-1
Helper Data
h(c)
c
Preprocessing and feature extraction
Transform features using Delaunay triangulation
Fixed length bit string representation
+
e
b
b1
+
e1
Preprocessing and feature extraction
Transform features using Delaunay triangulation
Fixed length bit string representation
Error correcting codeword decoding using Viterbi algorithm
c1
h(c1)
SHA-1
=
Query Image
?
Genuine / Imposter

Figure 5.23: Overall flow diagram for HYBRID_CC method

5.4.3 Matching

The enrolled template (e) and query template(e^1) is compared by the hamming measure, *i.e.*, the percentage of bits that differ in 'e' and e^1. To compute the match score (ms) we use thresholding (th_1) of hamming measure, say HM, as follows:

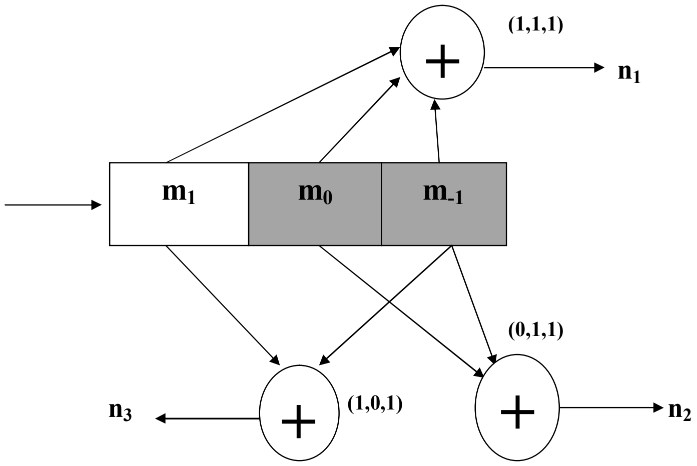


Figure 5.24: A (3,1,3)convolution encoder

```
if $HM(e,e^1)$ <$th_1$ then
    accept
    ms= $HM(e,e^1)$
else
    reject
    ms= $HM(e,e^1)$
end if
```

Algorithm 5.4 describes the HYBRID_CC method.

Algorithm 5.4 HYBRID_CC ($x_i, y_i, \theta_i, A_1, c_x, c_y, c_z, n, k, \lambda$): To devise a hybrid method using convolution coding and generate secure template

```
Input:   Minutiae locations $(x_i, y_i)$
           Orientation of minutiae points $\theta_i$
           Predefined 3D Array $A_1$ divided into cells of size $C_X, C_Y, C_Z$
           Input to BCH encoder $n, k, \lambda$
Output: Normalized match score

begin
Construct Delaunay triangulation net from the fingerprint minutia;
$N \leftarrow$ Number of triangles in Delaunay triangulation net;
Initialize triangle counter $j \leftarrow 0$;
```

Algorithm 5.4 HYBRID_CC (continued)

```
while $j \leq N$do                                                  //Refer to Figure 3.8
    // Compute incircle center of each triangle
        $(x_{in}, y_{in}) \leftarrow$ incircle center of triangle $j$;
    // Compute distance from vertices of the triangle to the incircle center
        $d_a \leftarrow \sqrt{(x_{in} - x_a)^2 + (y_{in} - y_a)^2}$;
        $d_b \leftarrow \sqrt{(x_{in} - x_b)^2 + (y_{in} - y_b)^2}$;
        $d_c \leftarrow \sqrt{(x_{in} - x_c)^2 + (y_{in} - y_c)^2}$;
        $\theta_a \leftarrow$ orientation of minutia point a;
        $\theta_b \leftarrow$ orientation of minutia point b;
        $\theta_c \leftarrow$ orientation of minutia point c;
    // Compute the internal angles of each triangle
        $\alpha_a \leftarrow \cos^{-1}(\frac{d_b^2+d_c^2-d_a^2}{2*d_b*d_c})$, $\alpha_b \leftarrow \cos^{-1}(\frac{d_a^2+d_c^2-d_b^2}{2*d_a*d_c})$
        $\alpha_c \leftarrow \cos^{-1}(\frac{d_a^2+d_3^b-d_2^c}{2*d_a*d_b})$
        $DTFS(j,:) \leftarrow (d_a, \theta_a, \alpha_a, d_b, \theta_b, \alpha_b, d_c, \theta_c, \alpha_c)$;
        $j \leftarrow j + 1$;
end while
Quantize DTFS by cell sizes $c_x, c_y, c_z$, map to 3D array $A_1$
Generate bit string $B_s$;
Initialize random key R of size $B_s$;
Codeword $c \leftarrow Convolution\_Code(R)$;                        //Refer to section 5.4.2
$e \leftarrow b \oplus c$;
    //Hash value generation
$h \leftarrow$ SHA-1(c);
    // Store encrypted template e
$Encrypt\_temp(i) \leftarrow e$;
Match score generation                                              //Refer to section 5.4.3
```

Algorithm 5.4 HYBRID_CC (continued)

```
23: return match_score
24: end
```

Computational Complexity of HYBRID_CC algorithm

Delaunay triangulation formation from t minutiae points: $O(t)$

Steps 5 to 14 repeated N times: $O(N)$

Algorithm Complexity: $O(t+N)$

Cell Sizes of 3D array			FVC 2002					
			DB1		DB2		DB3	
c_x	c_y	c_z	EER	d'	EER	d'	EER	d'
10	9	9	3.44	2.83	2.88	2.92	10.85	2.34
15	**15**	**15**	**2.66**	**3.53**	**1.89**	**3.12**	**6.87**	**2.65**
20	20	20	3.43	2.83	3.34	3.06	7.55	2.71
25	25	25	3.26	3.44	1.98	3.08	7.78	2.63

Table 5.5: EER and d' values obtained for HYBRID_CC method on FVC 2002 databases

5.4.4 Experimental Results and Discussion

The HYBRID_CC method is tested by tuning the cell sizes c_x,c_y and c_z in the quantization step.

5.4.4.1 Accuracy

Table 5.5 represents the EER obtained for HYBRID_CC method. For cell sizes $c_x = 15$,$c_y = 15$, and $c_z = 15$, HYBRID_CC method show optimal result, *i.e.*, an EER of 2.66%, 1.89% and 6.87% for FVC 2002 DB1, DB2 and DB3 respectively. Also the d' values shown in the Table 5.5 are inversely proportional to EER and shows better separation between genuine and imposter scores.

Figure 5.25, 5.26, 5.27 shows the ROC curves, FAR/FRR, and score distribution for HYBRID_CC method for databases FVC 2002 respectively.

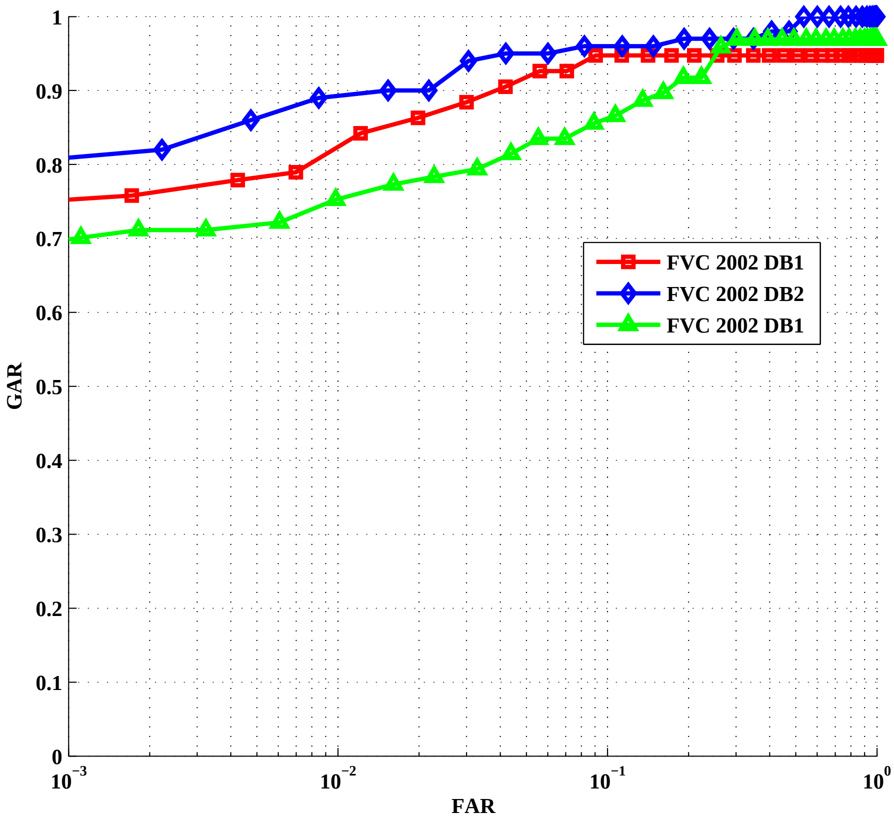


Figure 5.25: ROC curves of HYBRID_CC method for FVC 2002 databases

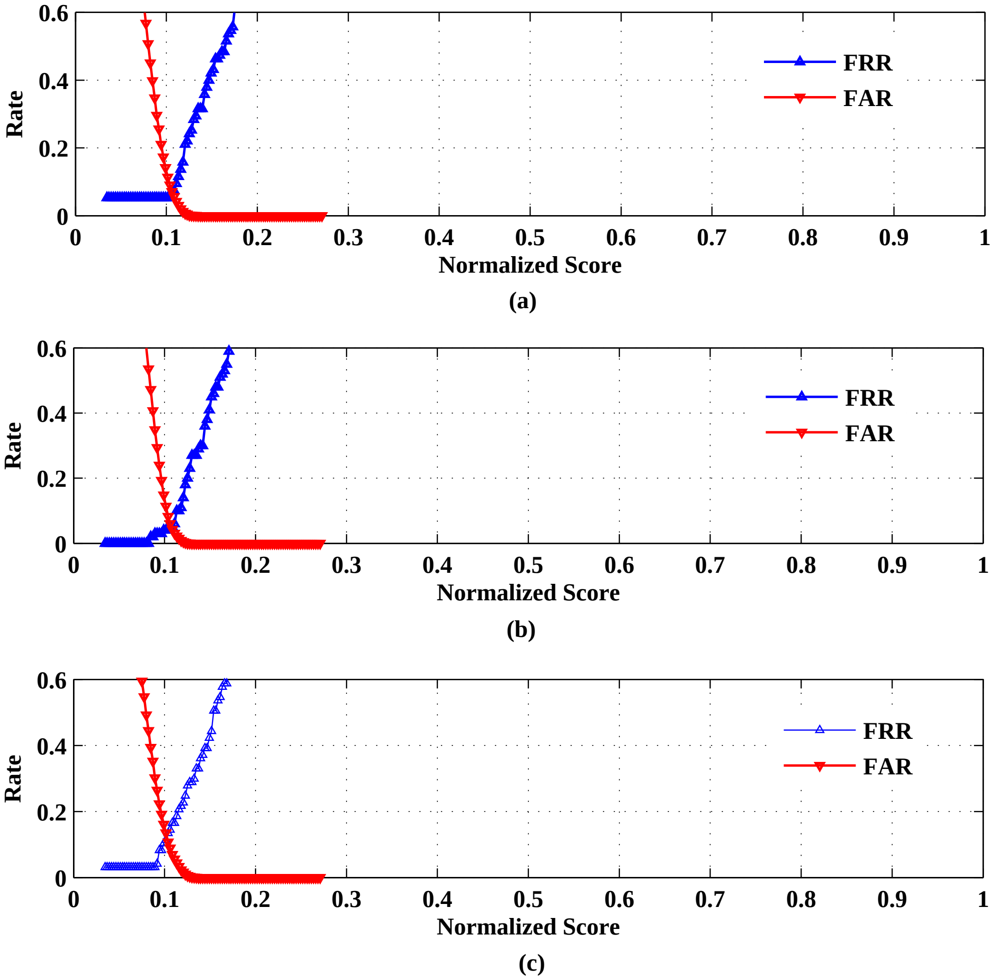
5.4.4.2 Revocability

To test revocability of HYBRID_CC method, we plot pseudo-imposter distribution as shown in Figure 5.28. From the figure, it is observed that pseudo-imposter distribution is close to imposter distribution and far from genuine distribution, which indicates that the transformed templates are not similar to accorded templates.

5.4.4.3 Security Analysis

Consider the following cases for performing security analysis on HYBRID_CC method

- **Case 1 :** Assume attacker obtained information stored in the template of a valid user, *i.e.*, the threshold value that is user-specific, the XOR'd binary vector and



(a)

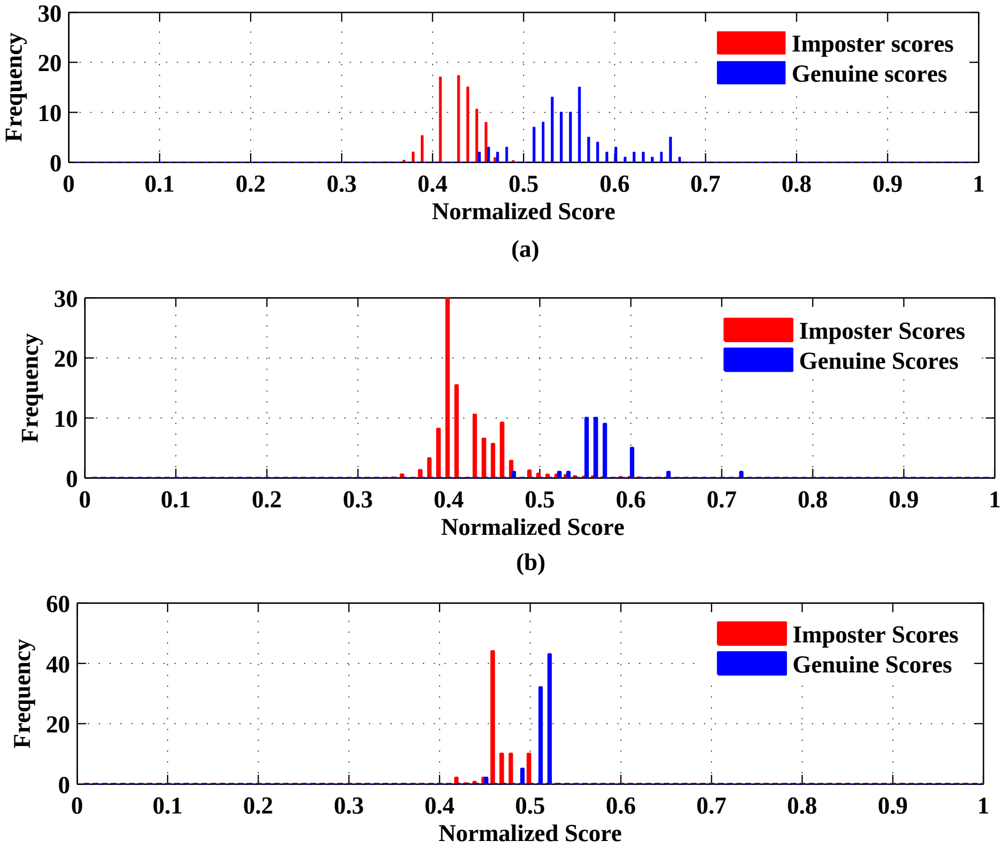
(b)

(c)

Figure 5.26: FAR/FRR of HYBRID_CC method for FVC 2002 (a)DB1 (b)DB2 (c)DB3 databases

the hash value of codeword. It is hard for the intruder to guess the original fingerprint minutiae information from the stored XOR'd template 'e' because of many-to-one mapping done to a cell in 3D array after quantization of cells. Hence, the proposed feature generation process is irreversible.

- **Case 2 :** The hill-climbing method described in [197] estimates the minutiae locations by generating synthetic templates. For HYBRID_CC method, it is hard to estimate the minutiae locations since the bit string generated is reduced in size through Dimensionality reduction matrix and hence does not store the



(a)

(b)

(c)

Figure 5.27: Genuine and Imposter score distribution of HYBRID_CC method for FVC 2002 (a)DB1 (b)DB2 (c)DB3 databases

location of minutiae points.

5.5 Performance Comparison with existing methods

We run our proposed methods BIOCRYP_DNS, HYBRID_DNS, HYBRID_MSC, and HYBRID_CC on FVC 2004 databases and the EER resulted is shown in Table 5.6. From the table, it is noticed that the proposed methods BIOCRYP_DNS, HYBRID_DNS, HYBRID_MSC, and HYBRID_CC shown less EER value compared to [18, 97, 99, 100, 105, 112, 118, 162] for FVC 2002 DB1, DB3, FVC 2004 DB1,

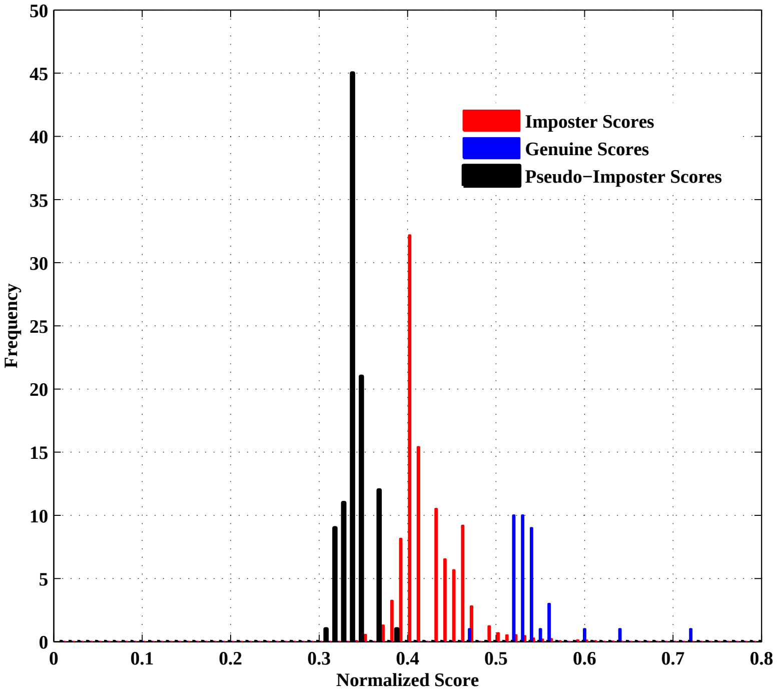


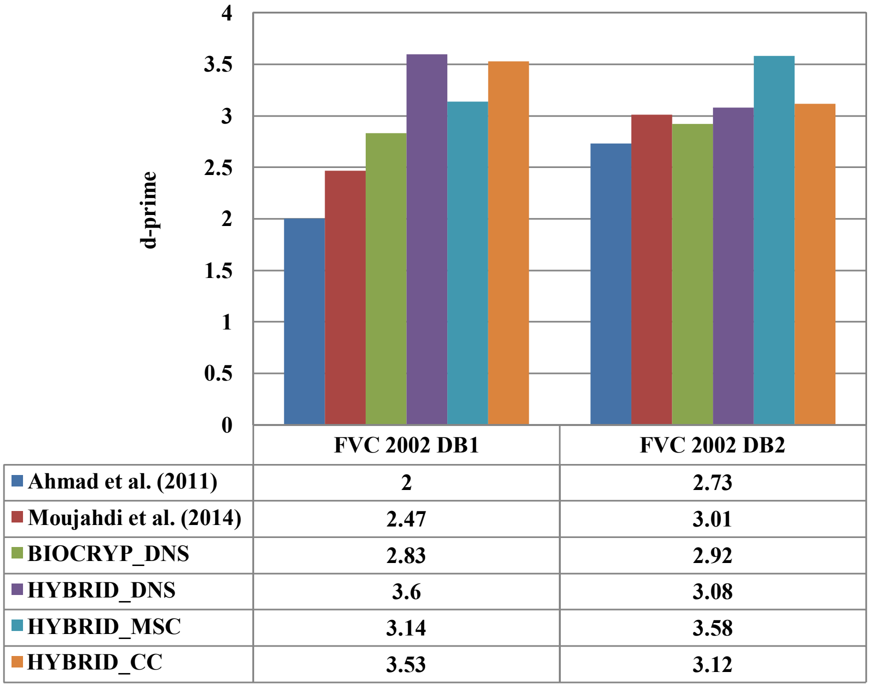
Figure 5.28: Genuine and Imposter and Psedo-imposter score distribution of HYBRID_CC method for FVC 2002 DB2

DB2, and DB3. For FVC 2002 DB2 the proposed methods shown lower EER values compared to [97, 99, 100, 105]. This indicates that the proposed methods shown better accuracy.

The comparison of d' values of proposed methods with the existing methods is made on FVC 2002 DB1 and DB2 databases. It is shown in Figure 5.29 that the separability is better, *i.e.* higher d' values are reported for the proposed methods BIOCRYP_DNS, HYBRID_DNS, HYBRID_MSC, and HYBRID_CC compared to [97, 113].

Method	**FVC 2002**			**FVC 2004**		
	DB1	DB2	DB3	DB1	DB2	DB3
Ahmad et al. (2011) [97]	9	6	27	-	-	-
Das et al. (2012) [100]	2.27	3.79	_	-	-	-
Wang and Hu (2012) [99]	3.5	4	7.5	-	-	-
Jin et al. (2012) [118]	5.19	-	-	15.76	11.64	-
Yang et al. (2013) [105]	5.93	4	-	-	-	-
Yang et al(2014) [162]	3.38	0.59	9.8	16.52	14.88	-
Jin et al. (2014) [18]	4.36	1.77	-	24.71	21.82	-
Prasad et al.(2014) [112]	1.8	1.2	-			
Proposed BIOCRYP_DNS	1.43	1.79	5.89	15.61	14.49	23.21
Proposed HYBRID_DNS	1.15	1.49	5.81	10.76	11.54	15.21
Proposed HYBRID_MSC	1.52	2.35	10.43	15.43	14.97	25.21
Proposed HYBRID_CC	2.66	1.81	6.87	14.72	13.54	20.37

Table 5.6: Comparison of EER for BIOCRYP_DNS, HYBRID_DNS, HYBRID_MSC, HYBRID_CC methods with existing methods



	FVC 2002 DB1	FVC 2002 DB2
🟦 Ahmad et al. (2011)	2	2.73
🟥 Moujahdi et al. (2014)	2.47	3.01
🟩 BIOCRYP_DNS	2.83	2.92
🟪 HYBRID_DNS	3.6	3.08
🟦 HYBRID_MSC	3.14	3.58
🟧 HYBRID_CC	3.53	3.12

Figure 5.29: Comparison of d' values for BIOCRYP_DNS, HYBRID_DNS, HYBRID_MSC, HYBRID_CC methods with existing methods

5.6 Summary

In this chapter, we presented bio-cryptosystem and hybrid methods developed for securing fingerprint templates. A bio-cryptosystem for fingerprints is devised by computing a new structure, called Delaunay Neighbor Structures (DNS) for the minutiae points. The use BCH error correcting code improved the performance of system thereby reducing EER. Also, the proposed BIOCRYP_DNS method satisfy the requirements of template protection schemes

Hybrid methods for securing fingerprint templates were developed by merging cancelable biometrics with bio-cryptosystem. The hybrid methods namely HYBRID_DNS, HYBRID_MSC, and HYBRID_CC are developed by combining cancelability and fuzzy commitment scheme. The hybrid methods improved the efficacy of system and enhance security. The experimental analysis proves the tenability of the proposed methods.

Chapter 6

Conclusions and Future Directions

6.1 Conclusions

In our present work, we presented three algorithms namely kNNS, DTFS_INCIR, and DTFS_AVGLO for generation of cancelable fingerprint templates. The kNNS algorithm is robust with regard to the Cartesian space, for instance, minutiae points need not be linearly separable. Moreover, to construct the kNNS for each minutia point there are only a few parameters to tune: distance metric and neighborhood parameter (k). The triangles in Delaunay triangulation net are structurally more stable under distortion. The usefulness of Delaunay triangulation opposed to other triangulation methods is the fact that triangles are well-shaped. Furthermore, the minutiae insertions and deletions affect the Delaunay triangulation structure locally. The proposed computation of feature sets kNNS, DTFS_INCIR and DTFS_AVGLO includes all minutiae points in the fingerprint image. This led to maintain a good balance between security and performance of the biometric system. Accuracy, revocability, diversity and irreversibility are clearly analyzed for the proposed algorithms. Further, the experimental evaluation of proposed algorithms is done not only in terms of EER but also in terms of d' and KS test values. Experimental results prove the validity and significance of the proposed algorithms.

We also presented feature level and score level fusion methods for generating cancelable fingerprint templates namely FEATURELEVEL_LSDS, SCORELEVEL_WS, and SCORELEVEL_TOP. FEATURELEVEL_LSDS fusion method is built using fused structures at the feature level. The fused structure maintains a good balance between security and performance. Experimental results prove the tenability of proposed feature level fusion method. Score level fusion methods are built by fusing kNNS and DTFS_AVGLO algorithms discussed in chapter 3. Since the algorithm kNNS uses local and global matching between enrolled and query templates, it results in better matching accuracy. By using DTFS_AVGLO, the minutia insertions and deletions affect the feature set computation locally which leads to better matching accuracy. Further, these two algorithms are combined by performing score-level fusion using weighted sum rule and T-operators. Experimental results clearly state that the proposed fusion methods when compared to each of the individual algorithms used for fusing shown better performance. From the experimental analysis, the potential of using T-operators in the generation of cancelable fingerprint templates is extinguished.

We also presented bio-cryptosystem and hybrid methods developed for securing fingerprint templates. A bio-cryptosystem for fingerprints is developed by computing a new structure called “Delaunay Neighbor Structures (DNS)” for the minutiae points. The use BCH error correcting code improved the performance of system thereby reducing the “Equal Error Rate(EER)”. Also, the proposed system (BIOCRYP_DNS) satisfies the requirements of template protection schemes and is clearly analyzed in the experimental evaluation. Hybrid methods for securing fingerprint templates were developed by merging cancelable biometrics with bio-cryptosystem. The hybrid methods namely HYBRID_DNS, HYBRID_MSC, and HYBRID_CC are developed by combining of cancelability and “fuzzy commitment scheme” improved the efficacy of system and enhances security. Also, the proposed hybrid methods satisfy the “requirements of template protection schemes”. The experimental analysis proves the tenability of the proposed methods

Alignment-free methods for generation of cancelable fingerprint templates that include all minutiae points for transformation process provides better security without degrading the performance of the system. Hybrid methods that combine cancelable biometrics and bio-cryptosystems shown better performance compared to alignment-free methods for protecting fingerprint templates. The use of Intra-modal fusion, *i.e.*, using same biometric modality (fingerprint) for fusion (viz. score level) for achieving fingerprint template security shown better performance compared to hybrid methods.

6.2 Future research directions

The following research directions are suggested for future:

(1) Most of the biometric template protection techniques designed so far are analyzed on small and mid-size databases. However, these schemes have to be evaluated on large scale databases.

(2) The hybrid template protection approaches, which have been paid little attention so far need to be developed to make use of advantages of multiple schemes. For *e.g.*, a scheme that secures a cancelable template using a biometric cryptosystem may have the advantages of both cancelable biometrics (which provides high diversity and revocability) and biometric cryptosystem (which provides high security) approaches.

(3) Multi-modal biometric template protection schemes are to be developed to make use of advantages of multi-biometrics.

(4) There is a rampant growth in the use of mobile devices in the past decade. The development of template protected active authentication systems make mobile devices secure from impersonation.

Author’s Publications

Journal Papers:

[1] Mulagala Sandhya, Munaga V.N.K Prasad, Raghavendra Rao Chillarige, “Generating cancellable fingerprint templates based on Delaunay triangle feature set construction”, *IET Biometrics*, Volume 5, Issue 2, pp. 131-139, 2016.

[2] Mulagala Sandhya, Munaga V.N.K Prasad, “Multi-algorithmic cancelable fingerprint template generation based on weighted-sum rule and T-operators”, *Pattern Analysis and Applications*, 2016, DOI: 10.1007/s10044-016-0584-5.

[3] Mulagala Sandhya, Munaga V.N.K Prasad, “Cancelable fingerprint cryptosystem using multiple spiral curves and fuzzy commitment scheme”, *International Journal of Pattern Recognition and Artificial Intelligence (IJPRAI)*, 2016, DOI: 10.1142/S0218001417560043.

[4] Mulagala Sandhya, Munaga V.N.K Prasad, “A Hybrid Approach For Securing Fingerprint Templates using Delaunay Neighbor Structures”, *International Journal of Trust Management in Computing and Communications (IJTMCC)* (In Press).

Papers in Conference Proceedings:

[5] Mulagala Sandhya, Munaga V.N.K Prasad, “k-Nearest Neighborhood Structure (k-NNS) based Alignment-free method for Fingerprint Template Protection”, *In: Proceedings of* 8^{th} *IAPR International Conference on Biometrics (ICB)*, pp. 386-393, 2015.

[6] Mulagala Sandhya, Munaga V.N.K Prasad, "A Bio-cryptosystem for fingerprints using Delaunay Neighbor Structures (DNS) and Fuzzy Commitment Scheme", *In: Proceedings of Second International Symposium on Signal Processing and Intelligent Recognition Systems (SIRS)*, pp. 145-157, 2016.

[7] Mulagala Sandhya, Munaga V.N.K Prasad, "Cancelable fingerprint cryptosystem based on convolution coding", *In: Proceedings of Second International Symposium on Signal Processing and Intelligent Recognition Systems (SIRS)*, pp. 159-171, 2016.

Book Chapter:

[8] Mulagala Sandhya, Munaga V.N.K Prasad, "Biometric Template Protection: A Systematic Literature Review of Approaches, Modalities and Fusion", *Biometric Security and Privacy – Opportunities & Challenges in The Big Data Era*, Edited by: Somaya Al-Maadeed, Richard Jiang, Danny Crookes, Azeddine Beghdadi, Ahmed Bouridane, Springer, 2016 (In Press).

Revision Submitted Journal Paper:

[9] Mulagala Sandhya, Munaga V.N.K Prasad, "Securing fingerprint templates using fused structures", *IET Biometrics*.

References

[1] Nalini K. Ratha, Jonathan H. Connell, and Ruud M. Bolle, "An Analysis of Minutiae Matching Strength", In *Proceedings of International Conference on Audio- and Video-Based Biometric Person Authentication (AVBPA)*, pp. 223–228, 2001.

[2] Barbara Kitchenham and Stuart Charters, Guidelines for performing Systematic Literature Reviews in Software Engineering, Technical Report EBSE 2007-001, Keele University and Durham University Joint Report, 2007.

[3] Barbara Kitchenham, "Procedures for performing systematic reviews", *Keele, UK, Keele University*, 33, pp. 1–26, 2004.

[4] Anil K. Jain, Karthik Nandakumar, and Arun Ross, "50 years of biometric research: Accomplishments, challenges, and opportunities", *Pattern Recognition Letters*, 79, pp. 80 – 105, 2016.

[5] Davide Maltoni, Dario Maio, Anil Jain, and Salil Prabhakar, *Handbook of fingerprint recognition*, Springer Science & Business Media, 2009.

[6] Anil K. Jain, P. Flynn, and Arun A. Ross, *Handbook of Biometrics*, Springer, 2007.

[7] AK Jain, A Ross, and S Prabhakar, "An introduction to biometric recognition", *IEEE Transactions on Circuits and Systems for Video Technology*, 14, pp. 4–20, 2004.

[8] Anil Jain, Arun A Ross, and Karthik Nandakumar, *Introduction to biometrics*, Springer Science & Business Media, 2011.

[9] A. K. Jain and K. Nandakumar, “Biometric Authentication: System Security and User Privacy”, *Computer*, 45(11), pp. 87–92, 2012.

[10] A. Jain, B. Klare, and A. Ross, “Guidelines for best practices in biometrics research”, In *Proceedings of International Conference on Biometrics (ICB)*, pp. 541–545, 2015.

[11] Arun A. Ross, Karthik Nandakumar, and Anil K. Jain, *Handbook of Multibiometrics*, Springer-Verlag, 2006.

[12] Anil K. Jain, Karthik Nandakumar, and Abhishek Nagar, “Biometric Template Security”, *EURASIP Journal of Advanced Signal Processing*, 2008, doi: 10.1155/2008/579416.

[13] Andy Adler and Raffaele Cappelli, “Template Security”, In *Encyclopedia of Biometrics*, pp. 1322–1327, Springer, 2009.

[14] Patrizio Campisi, “Security and Privacy in Biometrics: Towards a Holistic Approach”, In *Security and Privacy in Biometrics*, pp. 1–23, Springer, 2013.

[15] S. Hidano, T. Ohki, and K. Takahashi, “Evaluation of security for biometric guessing attacks in biometric cryptosystem using fuzzy commitment scheme”, In *Proceedings of the International Conference of the Biometrics Special Interest Group (BIOSIG)*, pp. 1–6, 2012.

[16] U. Uludag, S. Pankanti, S. Prabhakar, and A.K. Jain, “Biometric cryptosystems: Issues and Challenges”, *Proceedings of the IEEE*, 92(6), pp. 948–960, 2004.

[17] David Chek Ling Ngo, Andrew Beng Jin Teoh, and Jiankun Hu, *Biometric Security*, Cambridge Scholars Publishing, United Kingdom, 2015.

[18] Zhe Jin, Meng-Hui Lim, Andrew Beng Jin Teoh, and Bok-Min Goi, “A non-invertible Randomized Graph-based Hamming Embedding for generating cancelable fingerprint template”, *Pattern Recognition Letters*, 42, pp. 137–147, 2014.

[19] Andy Adler, “Can images be regenerated from biometric templates?”, In *Proceedings of Biometrics Consortium Conference*, 2003.

[20] Arun Ross, Jidnya Shah, and Anil K Jain, "From template to image: Reconstructing fingerprints from minutiae points", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(4), pp. 544–560, 2007.

[21] Wei Jing Wong, Andrew B.J. Teoh, Yau Hee Kho, and M.L. Dennis Wong, "Kernel PCA enabled bit-string representation for minutiae-based cancellable fingerprint template", *Pattern Recognition*, 51, pp. 197 – 208, 2016.

[22] K. Nandakumar and A.K. Jain, "Biometric Template Protection: Bridging the performance gap between theory and practice", *IEEE Signal Processing Magazine*, 32(5), pp. 88–100, 2015.

[23] ISO/IEC JTC1 SC27 24745, Biometric information protection, 2011.

[24] K. Simoens, Bian Yang, Xuebing Zhou, F. Beato, C. Busch, E.M. Newton, and B. Preneel, "Criteria towards metrics for benchmarking template protection algorithms", In *Proceedings of International Conference on Biometrics (ICB)*, pp. 498–505, 2012.

[25] Gregory J. Schmidt George J. Tomko, Colin Soutar, Fingerprint controlled public key cryptographic system, 1996, URL `http://www.google.co.in/patents/US5832091`, US Patent 5832091 A.

[26] Colin Soutar, Danny Roberge, Alex Stoianov, Rene Gilroy, and Bhagavatula Vijaya Kumar, "Biometric Encryption: enrollment and verification procedures", In *Proceedings of SPIE: Optical Pattern Recognition*, pp. 24–35, 1998.

[27] Ari Juels and Martin Wattenberg, "A Fuzzy Commitment Scheme", In *Proceedings of the ACM Conference on Computer and Communications Security*, CCS, pp. 28–36, 1999.

[28] Nalini K. Ratha, Jonathan H. Connell, and Ruud M. Bolle, "Enhancing security and privacy in biometrics-based authentication systems", *IBM systems Journal*, 40(3), pp. 614–634, 2001.

[29] A. Juels and M. Sudan, "A fuzzy vault scheme", In *Proceedings of International Symposium on Information Theory*, p. 408, 2002.

[30] Hao Feng and Chan Choong Wah, "Private key generation from online handwritten signatures", *Information Management & Computer Security*, 10(4), pp. 159–164, 2002.

[31] Y. Dodis, L. Reyzin, and A. Smith, "Fuzzy Extractors: How to Generate Strong Keys from Biometrics and Other Noisy Data", In *Proceedings of International Conference on the Theory and Applications of Cryptographic Techniques - EUROCRYPT*, pp. 523–540, 2004.

[32] Y. Sutcu, Qiming Li, and N. Memon, "Protecting Biometric Templates With Sketch: Theory and Practice", *IEEE Transactions on Information Forensics and Security*, 2(3), pp. 503–512, 2007.

[33] T.E. Boult, "Robust distance measures for face-recognition supporting revocable biometric tokens", In *Proceedings of International Conference on Automatic Face and Gesture Recognition (FGR)*, pp. 560–566, 2006.

[34] T.E. Boult, W.J. Scheirer, and R. Woodworth, "Revocable fingerprint biotokens: accuracy and security analysis", In *Proceedings of International Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–8, 2007.

[35] Shuiming Ye, Ying Luo, Jian Zhao, and Sen-Ching S. Cheung, "Anonymous Biometric Access Control", *EURASIP Journal of Information Security*, 2009.

[36] Tee Connie, A.B.J. Teoh, Michael Goh, and David Ngo, "PalmHashing: a novel approach for cancelable biometrics", *Information Processing Letters*, 93(1), pp. 1 – 5, 2005.

[37] Andrew B.J. Teoh, Yip Wai Kuan, and Sangyoun Lee, "Cancellable biometrics and annotations on BioHash", *Pattern Recognition*, 41(6), pp. 2034 – 2044, 2008.

[38] Ann Cavoukian and Alex Stoianov, "Biometric Encryption", In *Encyclopedia of Biometrics*, pp. 260–269, Springer, 2009.

[39] Umut Uludag and Anil K Jain, "Fuzzy fingerprint vault", In *Proceedings of Workshop: Biometrics: Challenges Arising from Theory to Practice*, pp. 13–16, 2004.

[40] C. Rathgeb, A. Uhl, and P. Wild, "Reliability-balanced feature level fusion for fuzzy commitment scheme", In *Proceedings of International Joint Conference on Biometrics (IJCB)*, pp. 1–7, 2011.

[41] Feng Hao, R. Anderson, and J. Daugman, "Combining Cryptography with Biometrics Effectively", *IEEE Transactions on Computers*, 55(9), pp. 1081–1088, 2006.

[42] Anil K. Jain, Karthik Nandakumar, and Abhishek Nagar, "Fingerprint Template Protection: From Theory to Practice", In *Security and Privacy in Biometrics*, pp. 187–214, 2013.

[43] Gang Zheng, Wanqing Li, and Ce Zhan, "Cryptographic Key Generation from Biometric Data Using Lattice Mapping", In *Proceedings of International Conference on Pattern Recognition (ICPR)*, pp. 513–516, 2006.

[44] Christian Rathgeb and Andreas Uhl, "A survey on biometric cryptosystems and cancelable biometrics", *EURASIP Journal on Information Security*, 2011, doi: 10.1186/1687-417X-2011-3.

[45] Umut Uludag, Sharath Pankanti, and Anil K. Jain, "Fuzzy Vault for Fingerprints", In *Proceedings of International Conference on Audio- and Video-Based Biometric Person Authentication (AVBPA)*, pp. 310–319, 2005.

[46] U. Uludag and A. Jain, "Securing Fingerprint Template: Fuzzy Vault with Helper Data", In *Proceedings of Conference on Computer Vision and Pattern Recognition Workshop (CVPRW)*, pp. 163–163, 2006.

[47] Sanaul Hoque, Michael Fairhurst, and Gareth Howells, "Evaluating biometric encryption key generation using handwritten signatures", In *Proceedings of Symposium on Bio-inspired Learning and Intelligent Systems for Security (BLISS)*, pp. 17–22, 2008.

[48] Christian Rathgeb and Andreas Uhl, "An iris-based interval-mapping scheme for biometric key generation", In *Proceedings of International Symposium on Image and Signal Processing and Analysis*, pp. 511–516, 2009.

[49] V.M. Patel, N.K. Ratha, and R. Chellappa, "Cancelable Biometrics: A review", *IEEE Signal Processing Magazine*, 32(5), pp. 54–65, 2015.

[50] Abhishek Nagar, Karthik Nandakumar, and Anil K. Jain, "A hybrid biometric cryptosystem for securing fingerprint minutiae templates", *Pattern Recognition Letters*, 31(8), pp. 733 – 741, 2010.

[51] Karthik Nandakumar, Abhishek Nagar, and Anil K Jain, "Hardening fingerprint fuzzy vault using password", In *Proceedings of International Conference on Biometrics (ICB)*, pp. 927–937, 2007.

[52] Zekeriya Erkin, Martin Franz, Jorge Guajardo, Stefan Katzenbeisser, Inald Lagendijk, and Tomas Toft, "Privacy-preserving face recognition", In *Proceedings of International Symposium on Privacy Enhancing Technologies*, pp. 235–253, 2009.

[53] Ahmad-Reza Sadeghi, Thomas Schneider, and Immo Wehrenberg, "Efficient privacy-preserving face recognition", In *Proceedings of International Conference on Information, Security and Cryptology*, pp. 229–244, 2009.

[54] Arun Ross and Anil Jain, "Information fusion in biometrics", *Pattern Recognition Letters*, 24(13), pp. 2115 – 2125, 2003.

[55] Ning Wang, Qiong Li, Ahmed A Abd El-Latif, Jialiang Peng, Xuehu Yan, and Xiamu Niu, "A novel template protection scheme for multibiometrics based on fuzzy commitment and chaotic system", *Signal, Image and Video Processing*, 9, pp. 1–11, 2015.

[56] Ruud M Bolle, Jonathan H Connell, and Nalini K Ratha, "Biometric perils and patches", *Pattern Recognition*, 35(12), pp. 2727–2738, 2002.

[57] R. Cappelli, D. Maio, A. Lumini, and D. Maltoni, “Fingerprint Image Reconstruction from Standard Templates”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(9), pp. 1489–1503, 2007.

[58] J. Feng and A. K. Jain, “Fingerprint Reconstruction: From Minutiae to Phase”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(2), pp. 209–223, 2011.

[59] S. Li and A. C. Kot, “An Improved Scheme for Full Fingerprint Reconstruction”, *IEEE Transactions on Information Forensics and Security*, 7(6), pp. 1906–1912, 2012.

[60] K. Cao and A. K. Jain, “Learning Fingerprint Reconstruction: From Minutiae to Image”, *IEEE Transactions on Information Forensics and Security*, 10(1), pp. 104–117, 2015.

[61] MinYi Jeong, Chulhan Lee, Jongsun Kim, Jeung-Yoon Choi, Kar-Ann Toh, and Jaihie Kim, “Changeable biometrics for appearance based face recognition”, In *Proceedings of Biometrics Symposium: Special Session on Research at the Biometric Consortium Conference*, pp. 1–5, 2006.

[62] A.B.J. Teoh and D.C.L. Ngo, “Biophasor: Token Supplemented Cancellable Biometrics”, In *Proceedings of International Conference on Control, Automation, Robotics and Vision (ICARCV)*, pp. 1–5, 2006.

[63] A. Teoh and Chong Tze Yuang, “Cancelable Biometrics Realization With Multispace Random Projections”, *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 37(5), pp. 1096–1106, 2007.

[64] Andrew Beng Jin Teoh and Lee-Ying Chong, “Secure speech template protection in speaker verification system”, *Speech Communication*, 52(2), pp. 150 – 163, 2010.

[65] W. K. Yip, A. B. J. Teoh, and D. C. L. Ngo, “Replaceable and securely hashed keys from online signatures”, *IEICE Electronics Express*, 3(18), pp. 410–416, 2006.

[66] L. Leng, J. S. Zhang, M. K. Khan, X. Bi, and M. Ji, "Cancelable PalmCode generated from randomized Gabor Filters for palmprint protection", In *Proceedings of International Conference of Image and Vision Computing New Zealand (IVCNZ)*, pp. 1–6, 2010.

[67] Lu Leng and Jiashu Zhang, "PalmHash Code vs. PalmPhasor Code", *Neurocomputing*, 108, pp. 1 – 12, 2013.

[68] Lu Leng, Andrew Beng Jin Teoh, Ming Li, and Muhammad Khurram Khan, "Analysis of correlation of 2DPalmHash Code and orientation range suitable for transposition", *Neurocomputing*, 131, pp. 377 – 387, 2014.

[69] A.B.J. Teoh, A. Goh, and D.C.L. Ngo, "Random Multispace Quantization as an Analytic Mechanism for BioHashing of Biometric and Random Identity Inputs", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(12), pp. 1892–1901, 2006.

[70] Adams Kong, King-Hong Cheung, David Zhang, Mohamed Kamel, and Jane You, "An analysis of BioHashing and its variants", *Pattern Recognition*, 39(7), pp. 1359 – 1368, 2006.

[71] Chong Siew Chin, Andrew Teoh Beng Jin, and David Ngo Chek Ling, "High security iris verification system based on random secret integration", *Computer Vision and Image Understanding*, 102(2), pp. 169–177, 2006.

[72] Youngsung Kim and Kar-Ann Toh, "A Method to Enhance Face Biometric Security", In *Proceedings of International Conference on Biometrics: Theory, Applications, and Systems (BTAS)*, pp. 1–6, 2007.

[73] Yongjin Wang and KN Plataniotis, "Face based biometric authentication with changeable and privacy preservable templates", In *Proceedings of Biometrics Symposium*, pp. 1–6, 2007.

[74] Jinyu Zuo, N.K. Ratha, and J.H. Connell, "Cancelable iris biometric", In *Proceedings of International Conference on Pattern Recognition (ICPR)*, pp. 1–4, 2008.

[75] Ouda Osama, Norimichi Tsumura, and Toshiya Nakaguchi, "Bioencoding: A reliable tokenless cancelable biometrics scheme for protecting iris codes", *IEICE Transactions on Information and Systems*, 93(7), pp. 1878–1888, 2010.

[76] Ouda Osama, Norimichi Tsumura, and Takao Nakaguchi, "Tokenless cancelable biometrics scheme for protecting iris codes", In *Proceedings of International Conference on Pattern Recognition (ICPR)*, pp. 882–885, 2010.

[77] Ouda Osama, Norimichi Tsumura, and Toshiya Nakaguchi, "On the Security of BioEncoding Based Cancelable Biometrics", *IEICE Transactions on Information and Systems*, 94(9), pp. 1768–1777, 2011.

[78] Yip Wai Kuan, Andrew BJ Teoh, and David CL Ngo, "Secure hashing of dynamic hand signatures using wavelet-Fourier compression with biophasor mixing and 2^N discretization", *EURASIP Journal of Advances in Signal Processing*, 2006.

[79] Shinji Hirata and Kenta Takahashi, "Cancelable biometrics with perfect secrecy for correlation-based matching", In *Proceedings of International Conference on Biometrics (ICB)*, pp. 868–878, 2009.

[80] J.K. Pillai, V.M. Patel, R. Chellappa, and N.K. Ratha, "Sectored Random Projections for Cancelable Iris Biometrics", In *Proceedings of International Conference on Acoustics Speech and Signal Processing (ICASSP)*, pp. 1838–1841, 2010.

[81] Nalini K Ratha, Sharat Chikkerur, Jonathan H Connell, and Ruud M Bolle, "Generating cancelable fingerprint templates", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(4), pp. 561–572, 2007.

[82] Nalini Ratha, Jonathan Connell, Ruud M Bolle, and Sharat Chikkerur, "Cancelable biometrics: A case study in fingerprints", In *Proceedings of International Conference on Pattern Recognition (ICPR)*, pp. 370–373, 2006.

[83] Sharat Chikkerur, Nalini K Ratha, Jonathan H Connell, and Ruud M Bolle, "Generating registration-free cancelable fingerprint templates", In *Proceedings*

of International Conference on Biometrics: Theory, Applications and Systems (BTAS), pp. 1–6, 2008.

[84] Jutta Hämmerle-Uhl, Elias Pschernig, and Andreas Uhl, "Cancelable iris biometrics using block re-mapping and image warping", In *Proceedings of International Conference on Information Security*, pp. 135–142, 2009.

[85] Peter Farberbock, Jutta Hämmerle-Uhl, Dominik Kaaser, Elias Pschernig, and Andreas Uhl, "Transforming Rectangular and Polar Iris Images to Enable Cancelable Biometrics", In *Proceedings of International Conference Image Analysis and Recognition (ICIAR)*, pp. 276–286, 2010.

[86] Christian Rathgeb and Andreas Uhl, "Secure iris recognition based on local intensity variations", In *Proceedings of International Conference on Image Analysis and Recognition (ICIAR)*, pp. 266–275, 2010.

[87] E Maiorana, M Martinez-Diaz, P Campisi, J Ortega-Garcia, and A Neri, "Template protection for HMM-based on-line signature authentication", In *Proceedings of Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 1–6, 2008.

[88] Emanuele Maiorana, Patrizio Campisi, Javier Ortega-Garcia, and Alessandro Neri, "Cancelable biometrics for hmm-based signature recognition", In *Proceedings of International Conference on Biometrics: Theory, Applications and Systems (BTAS)*, pp. 1–6, 2008.

[89] Emanuele Maiorana, Patrizio Campisi, Julian Fierrez, Javier Ortega-Garcia, and Alessandro Neri, "Cancelable templates for sequence-based biometrics with application to on-line signature recognition", *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, 40(3), pp. 525–538, 2010.

[90] Wenhua Xu, Qianhua He, Yanxiong Li, and Tao Li, "Cancelable Voiceprint Templates Based on Knowledge Signatures", In *Proceedings of International Symposium on Electronic Commerce and Security*, pp. 412–415, 2008.

[91] Jan Camenisch and Markus Stadler, “Efficient group signature schemes for large groups”, In *Proceedings of International Conference on Advances in Cryptology*, pp. 410–424, 1997.

[92] Huijuan Yang, Xudong Jiang, and A.C. Kot, “Generating secure cancelable fingerprint templates using local and global features”, In *Proceedings of International Conference on Computer Science and Information Technology (ICCSIT)*, pp. 645–649, 2009.

[93] J.K. Pillai, V.M. Patel, R. Chellappa, and N.K. Ratha, “Secure and Robust Iris Recognition Using Random Projections and Sparse Representations”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(9), pp. 1877–1893, 2011.

[94] Chulhan Lee and Jaihie Kim, “Cancelable fingerprint templates using minutiae-based bit-strings”, *Journal of Network and Computer Applications*, 33(3), pp. 236 – 246, 2010.

[95] Loris Nanni, Emanuele Maiorana, Alessandra Lumini, and Patrizio Campisi, “Combining local, regional and global matchers for a template protected on-line signature verification system”, *Expert Systems with Applications*, 37(5), pp. 3676 – 3684, 2010.

[96] Jin Zhe and A.T.B. Jin, “Fingerprint template protection with Minutia Vicinity Decomposition”, In *Proceedings of International Joint Conference on Biometrics (IJCB)*, pp. 1–7, 2011.

[97] Tohari Ahmad, Jiankun Hu, and Song Wang, “Pair-polar coordinate-based cancelable fingerprint templates”, *Pattern Recognition*, 44(10–11), pp. 2555 – 2564, 2011.

[98] Kenta Takahashi and Shinji Hirata, “Cancelable biometrics with provable security and its application to fingerprint verification”, *IEICE Transactions on fundamentals of electronics, communications and computer sciences*, 94(1), pp. 233–244, 2011.

[99] Song Wang and Jiankun Hu, "Alignment-free cancelable fingerprint template design: A densely infinite-to-one mapping (DITOM) approach", *Pattern Recognition*, 45(12), pp. 4129 – 4137, 2012.

[100] Priyanka Das, Kannan Karthik, and Boul Chandra Garai, "A robust alignment-free fingerprint hashing algorithm based on minimum distance graphs", *Pattern Recognition*, 45(9), pp. 3373 – 3388, 2012.

[101] Chouaib Moujahdi, Sanaa Ghouzali, Mounia Mikram, Mohammed Rziza, and George Bebis, "Spiral Cube for Biometric Template Protection", In *Proceedings of International Conference on Image and Signal Processing (ICISP)*, pp. 235–244, 2012.

[102] M. Ferrara, D. Maltoni, and R. Cappelli, "Noninvertible Minutia Cylinder-Code Representation", *IEEE Transactions on Information Forensics and Security*, 7 (6), pp. 1727–1737, 2012.

[103] Wei-jing Wong, Mou-ling Dennis Wong, and Yau-hee Kho, "Multi-line code: A low complexity revocable fingerprint template for cancelable biometrics", *Journal of Central South University*, 20(5), pp. 1292–1297, 2013.

[104] Wei Jing Wong, Andrew B.J. Teoh, M.L. Dennis Wong, and Yau Hee Kho, "Enhanced multi-line code for minutiae-based fingerprint template protection", *Pattern Recognition Letters*, 34(11), pp. 1221 – 1229, 2013.

[105] Wencheng Yang, Jiankun Hu, Song Wang, and Jucheng Yang, "Cancelable Fingerprint Templates with Delaunay Triangle-Based Local Structures", In *Proceedings of International Symposium Cyberspace Safety and Security (CSS)*, pp. 81–91, 2013.

[106] C. Rathgeb, F. Breitinger, and C. Busch, "Alignment-free cancelable iris biometric templates based on adaptive bloom filters", In *Proceedings of International Conference on Biometrics (ICB)*, pp. 1–8, 2013.

[107] C. Rathgeb, F. Breitinger, C. Busch, and H. Baier, "On application of bloom filters to iris biometrics", *IET Biometrics*, 3(4), pp. 207–218, 2014.

[108] M. Ferrara, D. Maltoni, and R. Cappelli, “A two-factor protection scheme for MCC fingerprint templates”, In *Proceedings of International Conference of the Biometrics Special Interest Group (BIOSIG)*, pp. 1–8, 2014.

[109] C. Rathgeb and C. Busch, “Cancelable multi-biometrics: Mixing iris-codes based on adaptive bloom filters”, *Computers and Security*, 42, pp. 1 – 12, 2014.

[110] Munaga VNK Prasad and C Santhosh Kumar, “Fingerprint template protection using multiline neighboring relation”, *Expert Systems with Applications*, 41(14), pp. 6114–6122, 2014.

[111] Zhe Jin, Bok-Min Goi, Andrew Teoh, and Yong Haur Tay, “A two-dimensional random projected minutiae vicinity decomposition-based cancellable fingerprint template”, *Security and Communication Networks*, 7(11), pp. 1691–1701, 2014.

[112] Munaga V.N.K Prasad, P Swathi, CR Rao, and BL Deekshatulu, “Minimum Spanning Tree (MST) based techniques for generation of cancelable fingerprint templates”, *International Journal of Pattern Recognition and Artificial Intelligence*, 28(06), pp. 1456013, 2014.

[113] Chouaib Moujahdi, George Bebis, Sanaa Ghouzali, and Mohammed Rziza, “Fingerprint shell: Secure representation of fingerprint template”, *Pattern Recognition Letters*, 45, pp. 189 – 196, 2014.

[114] Lu Leng, Andrew Beng Jin Teoh, Ming Li, and Muhammad Khurram Khan, “A remote cancelable palmprint authentication protocol based on multi-directional two-dimensional PalmPhasor-fusion”, *Security and Communication Networks*, 7(11), pp. 1860–1871, 2014.

[115] M.A. Murillo-Escobar, C. Cruz-Hernández, F. Abundiz-Pérez, and R.M. López-Gutiérrez, “A robust embedded biometric authentication system based on fingerprint and chaotic encryption”, *Expert Systems with Applications*, 42(21), pp. 8198 – 8211, 2015.

[116] Song Wang and Jiankun Hu, “A Blind System Identification Approach to Cancelable Fingerprint Templates”, *Pattern Recognition*, 54, pp. 14–22, 2016.

[117] Song Wang, Guang Deng, and Jiankun Hu, "A partial Hadamard transform approach to the design of cancelable fingerprint templates containing binary biometric representations", *Pattern Recognition*, 61, pp. 447 – 458, 2017.

[118] Zhe Jin, Andrew Beng Jin Teoh, Thian Song Ong, and Connie Tee, "Fingerprint template protection with minutiae-based bit-string for security and privacy preserving", *Expert Systems with Applications*, 39(6), pp. 6157 – 6167, 2012.

[119] R. Cappelli, M. Ferrara, and D. Maltoni, "Minutia Cylinder-Code: A New Representation and Matching Technique for Fingerprint Recognition", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(12), pp. 2128–2141, 2010.

[120] J. Bringer, C. Morel, and C. Rathgeb, "Security analysis of Bloom filter-based iris biometric template protection", In *Proceedings of International Conference on Biometrics (ICB)*, pp. 527–534, 2015.

[121] J. Bringer, H. Chabanne, G. Cohen, B. Kindarji, and G. Zemor, "Optimal Iris Fuzzy Sketches", In *Proceedings of International Conference on Biometrics: Theory, Applications, and Systems (BTAS)*, pp. 1–6, 2007.

[122] Andrew Beng Jin Teoh and Jaihie Kim, "Secure biometric template protection in fuzzy commitment scheme", *IEICE Electronics Express*, 4(23), pp. 724–730, 2007.

[123] Emanuele Maiorana, Patrizio Campisi, and Alessandro Neri, "User adaptive fuzzy commitment for signature template protection and renewability", *Journal of Electronic Imaging*, 17(1), 2008.

[124] J. Bringer, H. Chabanne, G. Cohen, B. Kindarji, and G. Zemor, "Theoretical and Practical Boundaries of Binary Secure Sketches", *IEEE Transactions on Information Forensics and Security*, 3(4), pp. 673–683, 2008.

[125] Meng Ao and StanZ. Li, "Near Infrared Face Based Biometric Key Binding", In *Proceedings of International Conference on Biometrics (ICB)*, pp. 376–385, 2009.

[126] Christian Rathgeb and Andreas Uhl, "Systematic construction of iris-based fuzzy commitment schemes", In *Proceedings of International Conference on Biometrics (ICB)*, pp. 940–949, 2009.

[127] Haiping Lu, Karl Martin, Francis Bui, K. N. Plataniotis, and Dimitris Hatzinakos, "Face Recognition with Biometric Encryption for Privacy-enhancing Self-exclusion", In *Proceedings of the International Conference on Digital Signal Processing (DSP)*, pp. 625–632, 2009.

[128] E. Maiorana and P. Campisi, "Fuzzy Commitment for Function Based Signature Template Protection", *IEEE Signal Processing Letters*, 17(3), pp. 249–252, 2010.

[129] K. Nandakumar, "A fingerprint cryptosystem based on minutiae phase spectrum", In *Proceedings of International Workshop on Information Forensics and Security (WIFS)*, pp. 1–6, 2010.

[130] C. Rathgeb and A. Uhl, "Adaptive fuzzy commitment scheme based on iris-code error analysis", In *Proceedings of European Workshop on Visual Information Processing (EUVIP)*, pp. 41–44, 2010.

[131] Peng Li, Xin Yang, Hua Qiao, Kai Cao, Eryun Liu, and Jie Tian, "An effective biometric cryptosystem combining fingerprints with error correction codes", *Expert Systems with Applications*, 39(7), pp. 6562 – 6574, 2012.

[132] Yadigar Imamverdiyev, Andrew Beng Jin Teoh, and Jaihie Kim, "Biometric cryptosystem based on discretized fingerprint texture descriptors", *Expert Systems with Applications*, 40(5), pp. 1888 – 1901, 2013.

[133] Li Yuan, "Multimodal Cryptosystem Based on Fuzzy Commitment", In *Proceedings of International Conference on Computational Science and Engineering (CSE)*, pp. 1545–1549, 2014.

[134] S. Billeb, C. Rathgeb, H. Reininger, K. Kasper, and C. Busch, "Biometric template protection for speaker recognition based on universal background models", *IET Biometrics*, 4(2), pp. 116–126, 2015.

[135] T. Ignatenko and F.M.J. Willems, “Information Leakage in Fuzzy Commitment Schemes”, *IEEE Transactions on Information Forensics and Security*, 5(2), pp. 337–348, 2010.

[136] C. Rathgeb and A. Uhl, “Statistical attack against iris-biometric fuzzy commitment schemes”, In *Proceedings of Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 23–30, 2011.

[137] C. Rathgeb and A. Uhl, “Statistical attack against fuzzy commitment scheme”, *IET Biometrics*, 1(2), pp. 94–104, 2012.

[138] Christian Rathgeb and Andreas Uhl, “Iris-biometric fuzzy commitment schemes under signal degradation”, In *Proceedings of International Conference on Image and Signal Processing (ICISP)*, pp. 217–225, 2012.

[139] Xuebing Zhou, A. Kuijper, R. Veldhuis, and C. Busch, “Quantifying privacy and security of biometric fuzzy commitment”, In *Proceedings of International Joint Conference on Biometrics (IJCB)*, pp. 1–8, 2011.

[140] Xuebing Zhou, A. Kuijper, and C. Busch, “Retrieving secrets from iris fuzzy commitment”, In *Proceedings of International Conference on Biometrics (ICB)*, pp. 238–244, 2012.

[141] Xuebing Zhou and C. Busch, “Measuring privacy and security of iris fuzzy commitment”, In *Proceedings of International Carnahan Conference on Security Technology (ICCST)*, pp. 168–173, 2012.

[142] E.J.C. Kelkboom, J. Breebaart, I. Buhan, and R. Veldhuis, “Maximum Key Size and Classification Performance of Fuzzy Commitment for Gaussian Modeled Biometric Sources”, *IEEE Transactions on Information Forensics and Security*, 7(4), pp. 1225–1241, 2012.

[143] K. Nandakumar, A.K. Jain, and S. Pankanti, “Fingerprint-Based Fuzzy Vault: Implementation and Performance”, *IEEE Transactions on Information Forensics and Security*, 2(4), pp. 744–757, 2007.

[144] Abhishek Nagar, Karthik Nandakumar, and Anil K Jain, “Securing fingerprint template: Fuzzy vault with minutiae descriptors”, In *Proceedings of International Conference on Pattern Recognition (ICPR)*, pp. 1–4, 2008.

[145] DaeHun Nyang and KyungHee Lee, “Fuzzy face vault: how to implement fuzzy vault with weighted features”, In *Proceedings of International Conference on Universal Access in Human-Computer Interaction (UAHCI)*, pp. 491–496, 2007.

[146] Youn Joo Lee, Kang Ryoung Park, Sung Joo Lee, Kwanghyuk Bae, and Jaihie Kim, “A New Method for Generating an Invariant Iris Private Key Based on the Fuzzy Vault System”, *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 38(5), pp. 1302–1313, 2008.

[147] E.J.C. Kelkboom, X. Zhou, J. Breebaart, R.N.J. Veldhuis, and C. Busch, “Multi-algorithm fusion with template protection”, In *Proceedings of International Conference on Biometrics: Theory, Applications, and Systems (BTAS)*, pp. 1–8, 2009.

[148] Xuebing Zhou, A. Opel, J. Merkle, U. Korte, and C. Busch, “Enhanced template protection with passwords for fingerprint recognition”, In *Proceedings of International Workshop on Security and Communication Networks (IWSCN)*, pp. 67–74, 2011.

[149] A. Nagar, K. Nandakumar, and A.K. Jain, “Multibiometric Cryptosystems Based on Feature-Level Fusion”, *IEEE Transactions on Information Forensics and Security*, 7(1), pp. 255–268, 2012.

[150] Thi Hanh Nguyen, Yi Wang, Yajun Ha, and Renfa Li, “Improved chaff point generation for vault scheme in bio-cryptosystems”, *IET Biometrics*, 2(2), pp. 48–55, 2013.

[151] G.S. Eskander, R. Sabourin, and E. Granger, “Improving Signature-Based Biometric Cryptosystems Using Cascaded Signature Verification-Fuzzy Vault (SV-FV) Approach”, In *Proceedings of International Conference on Frontiers in Handwriting Recognition (ICFHR)*, pp. 187–192, 2014.

[152] George S. Eskander, Robert Sabourin, and Eric Granger, “A bio-cryptographic system based on offline signature images”, *Information Sciences*, 259, pp. 170 – 191, 2014.

[153] G.S. Eskander, R. Sabourin, and E. Granger, “Offline signature-based fuzzy vault: A review and new results”, In *Proceedings of Symposium on Computational Intelligence in Biometrics and Identity Management (CIBIM)*, pp. 45–52, 2014.

[154] A.K. Bhateja, S. Chaudhury, and P.K. Saxena, “A Robust Online Signature Based Cryptosystem”, In *Proceedings of International Conference on Frontiers in Handwriting Recognition (ICFHR)*, pp. 79–84, 2014.

[155] Liu Hong-wei and Wang Yao, “A new fuzzy fingerprint vault using multivariable linear function based on Lorenz Chaotic System”, In *Proceedings of International Conference on Computer Science and Automation Engineering (CSAE)*, volume 1, pp. 531–534, 2012.

[156] Hailun Liu, Dongmei Sun, Ke Xiong, and Zhengding Qiu, “Palmprint based multidimensional fuzzy vault scheme”, *The Scientific World Journal*, 2014.

[157] C. Rathgeb, J. Wagner, B. Tams, and C. Busch, “Preventing the cross-matching attack in bloom filter-based cancelable biometrics”, In *Proceedings of International Workshop on Biometrics and Forensics (IWBF)*, pp. 1–6, 2015.

[158] Thi Hanh Nguyen, Yi Wang, Yajun Ha, and Renfa Li, “Performance and security-enhanced fuzzy vault scheme based on ridge features for distorted fingerprints”, *IET Biometrics*, 4(1), pp. 29–39, 2015.

[159] B. Tams, J. Merkle, C. Rathgeb, J. Wagner, U. Korte, and C. Busch, “Improved Fuzzy Vault Scheme for Alignment-Free Fingerprint Features”, In *Proceedings of International Conference of theBiometrics Special Interest Group (BIOSIG)*, pp. 1–12, 2015.

[160] D. Bansal, S. Sofat, and M. Kaur, “Fingerprint fuzzy vault using hadamard transformation”, In *Proceedings of International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pp. 1830–1834, 2015.

[161] Cai Li and Jiankun Hu, "A Security-Enhanced Alignment-Free Fuzzy Vault-Based Fingerprint Cryptosystem Using Pair-Polar Minutiae Structures", *IEEE Transactions on Information Forensics and Security*, 11(3), pp. 543–555, 2016.

[162] Wencheng Yang, Jiankun Hu, Song Wang, and Milos Stojmenovic, "An alignment-free fingerprint bio-cryptosystem based on modified Voronoi neighbor structures", *Pattern Recognition*, 47(3), pp. 1309 – 1320, 2014.

[163] Yevgeniy Dodis, Leonid Reyzin, and Adam Smith, "Fuzzy Extractors: How to Generate Strong Keys from Biometrics and Other Noisy Data", In *Proceedings of Advances in Cryptology*, pp. 523–540, 2004.

[164] B. Tams, P. Mihailescu, and A. Munk, "Security Considerations in Minutiae-Based Fuzzy Vaults", *IEEE Transactions on Information Forensics and Security*, 10(5), pp. 985–998, 2015.

[165] M. Lafkih, P. Lacharme, C. Rosenberger, M. Mikram, S. Ghouzali, M. El Haziti, and D. Aboutajdine, "Vulnerabilities of fuzzy vault schemes using biometric data with traces", In *Proceedings of International Wireless Communications and Mobile Computing Conference (IWCMC)*, pp. 822–827, 2015.

[166] Fabian Monrose, Michael K Reiter, Qi Li, and Susanne Wetzel, "Cryptographic key generation from voice", In *Proceedings of Symposium on Security and Privacy, S&P*, pp. 202–213, 2001.

[167] Fabian Monrose, Michael K Reiter, Qi Li, and Susanne Wetzel, "Using voice to generate cryptographic keys", In *Proceedings of A Speaker Odyssey-The Speaker Recognition Workshop*, pp. 237–242, 2001.

[168] B. Carrara and C. Adams, "You are the key: Generating cryptographic keys from voice biometrics", In *Proceedings of International Conference on Privacy Security and Trust (PST)*, pp. 213–222, 2010.

[169] Lifang Wu, Xingsheng Liu, Songlong Yuan, and Peng Xiao, "A novel key generation cryptosystem based on face features", In *Proceedings of International Conference on Signal Processing (ICSP)*, pp. 1675–1678, 2010.

[170] Eryun Liu, Jimin Liang, Liaojun Pang, Min Xie, and Jie Tian, "Minutiae and modified Biocode fusion for fingerprint-based key generation", *Journal of Network and Computer Applications*, 33(3), pp. 221 – 235, 2010.

[171] Y. C. Feng, P. C. Yuen, and A. K. Jain, "A Hybrid Approach for Generating Secure and Discriminating Face Template", *IEEE Transactions on Information Forensics and Security*, 5(1), pp. 103–117, 2010.

[172] Hailun Liu, Dongmei Sun, Ke Xiong, and Zhengding Qiu, "A hybrid approach to protect palmprint templates", *The Scientific World Journal*, 2014.

[173] Y.J. Chin, T.S. Ong, A.B.J. Teoh, and K.O.M. Goh, "Integrated biometrics template protection technique based on fingerprint and palmprint feature-level fusion", *Information Fusion*, 18, pp. 161 – 174, 2014.

[174] A. Kumar and A. Kumar, "A Cell-Array Based Multibiometric Cryptosystem", *IEEE Access*, 4, pp. 15–25, 2016.

[175] Zhe Jin, Andrew Beng Jin Teoh, Bok-Min Goi, and Yong-Haur Tay, "Biometric cryptosystems: A new biometric key binding and its implementation for fingerprint minutiae-based representation", *Pattern Recognition*, 56, pp. 50 – 62, 2016.

[176] Lu Leng and Andrew Beng Jin Teoh, "Alignment-free row-co-occurrence cancelable palmprint Fuzzy Vault", *Pattern Recognition*, 48(7), pp. 2290 – 2303, 2015.

[177] Shantanu D Rane, Wei Sun, and Anthony Vetro, "Secure distortion computation among untrusting parties using homomorphic encryption", In *Proceedings of International Conference on Image Processing (ICIP)*, pp. 1485–1488, 2009.

[178] M. Barni, T. Bianchi, D. Catalano, M. Di Raimondo, R. D. Labati, P. Failla, D. Fiore, R. Lazzeretti, V. Piuri, A. Piva, and F. Scotti, "A privacy-compliant fingerprint recognition system based on homomorphic encryption and Fingercode templates", In *Proceedings of International Conference on Biometrics: Theory Applications and Systems (BTAS)*, pp. 1–7, 2010.

[179] M. Barni, T. Bianchi, D. Catalano, M. Di Raimondo, R. Donida Labati, P. Failla, D. Fiore, R. Lazzeretti, V. Piuri, F. Scotti, et al., “Privacy-preserving fingercode authentication”, In *Proceedings of the ACM workshop on Multimedia and security*, pp. 231–240, 2010.

[180] Margarita Osadchy, Benny Pinkas, Ayman Jarrous, and Boaz Moskovich, “Scifi-a system for secure face identification”, In *Proceedings of Symposium on Security and Privacy (SP)*, pp. 239–254, 2010.

[181] Rohan Kulkarni and Anoop Namboodiri, “Secure hamming distance based biometric authentication”, In *Proceedings of International Conference on Biometrics (ICB)*, pp. 1–6, 2013.

[182] Cagatay Karabat, Mehmet Sabir Kiraz, Hakan Erdogan, and Erkay Savas, “THRIVE: threshold homomorphic encryption based secure and privacy preserving biometric verification system”, *EURASIP Journal on Advances in Signal Processing*, 2015, doi: 10.1186/s13634-015-0255-5.

[183] M. Gomez-Barrero, J. Fierrez, and J. Galbally, “Variable-length template protection based on homomorphic encryption with application to signature biometrics”, In *Proceedings of International Conference on Biometrics and Forensics (IWBF)*, pp. 1–6, 2016.

[184] A. Nagar and A.K. Jain, “On the security of non-invertible fingerprint template transforms”, In *Proceedings of International Workshop on Information Forensics and Security (WIFS)*, pp. 81–85, 2009.

[185] Abhishek Nagar, Karthik Nandakumar, and Anil K. Jain, “Biometric template transformation: a security analysis”, In *Proceedings of SPIE: Electronic Imaging, Media Forensics and Security*, volume 7541, 2010.

[186] Kenta Takahashi, Takahiro Matsuda, Takao Murakami, Goichiro Hanaoka, and Masakatsu Nishigaki, “A Signature Scheme with a Fuzzy Private Key”, In *International Conference on Applied Cryptography and Network Security (ACNS)*, pp. 105–126, 2015.

[187] Kenta Takahashi and Takao Murakami, “Optimal Sequential fusion for Multibiometric Cryptosystems”, *Information Fusion*, 32, pp. 93–108, 2016.

[188] Cai Li, Jiankun Hu, J. Pieprzyk, and W. Susilo, “A New Biocryptosystem-Oriented Security Analysis Framework and Implementation of Multibiometric Cryptosystems Based on Decision Level Fusion”, *IEEE Transactions on Information Forensics and Security*, 10(6), pp. 1193–1206, 2015.

[189] M. Savvides, B.V.K.V. Kumar, and P.K. Khosla, “Cancelable biometric filters for face recognition”, In *Proceedings of International Conference on Pattern Recognition (ICPR)*, pp. 922–925 Vol.3, 2004.

[190] J. Ortega-Garcia, J. Fierrez-Aguilar, D. Simon, J. Gonzalez, M. Faundez-Zanuy, V. Espinosa, A. Satue, I. Hernaez, J.-J. Igarza, C. Vivaracho, D. Escudero, and Q.-I. Moro, “MCYT baseline corpus: a bimodal biometric database”, *IEE Proceedings Vision, Image and Signal Processing*, 150(6), 2003.

[191] Steven Fortune, *Voronoi Diagrams and Delaunay Triangulations*, CRC Press, Inc., 1997.

[192] Manuel Abellanas, Ferran Hurtado, and Pedro A. Ramos, “Structural tolerance and Delaunay triangulation”, *Information Processing Letters*, 71(5–6), pp. 221 – 227, 1999.

[193] J. Wayman, Anil K. Jain, D. Maltoni, and D. Maio, *Biometric Systems: Technology, Design and Performance Evaluation*, Springer Verlag, 2005.

[194] Fingerprint Verification Competetion, `http://bias.csr.unibo.it/fvc2002`,.

[195] Fingerprint Verification Competetion, `http://bias.csr.unibo.it/fvc2004`,.

[196] Neurotechnology VeriFinger SDK, `http://www.neurotechnology.com`.

[197] Umut Uludag and Anil K. Jain, "Attacks on biometric systems: a case study in fingerprints", In *Proceedings of SPIE: Security, Steganography, and Watermarking of Multimedia Contents*, volume 5306, pp. 622–633, 2004.

[198] Joachim Gudmundsson, Mikael Hammar, and Marc van Kreveld, "Higher order Delaunay triangulations", *Computational Geometry*, 23(1), pp. 85 – 98, 2002.

[199] Joachim Gudmundsson, Herman J. Haverkort, and Marc van Kreveld, "Constrained higher order Delaunay triangulations", *Computational Geometry*, 30 (3), pp. 271 – 277, 2005.

[200] Norman Poh and Samy Bengio, "Database, protocols and tools for evaluating score-level fusion algorithms in biometric authentication", *Pattern Recognition*, 39(2), pp. 223 – 233, 2006.

[201] Qian Tao and Raymond Veldhuis, "Threshold-optimized decision-level fusion and its application to biometrics", *Pattern Recognition*, 42(5), pp. 823 – 836, 2009.

[202] Mingxing He, Shi-Jinn Horng, Pingzhi Fan, Ray-Shine Run, Rong-Jian Chen, Jui-Lin Lai, and Muhammad Khurram Khanand Kevin Octavius Sentosai, "Performance evaluation of score level fusion in multimodal biometric systems", *Pattern Recognition Letters*, 43(5), pp. 1789–1800, 2010.

[203] P. P. Paul and M. Gavrilova, "Rank level fusion of multimodal cancelable biometrics", In *Proceedings of International Conference on Cognitive Informatics Cognitive Computing*, pp. 80–87, 2014.

[204] Jialiang Peng, Ahmed A. Abd El-Latif, Qiong Li, and Xiamu Niu, "Multimodal biometric authentication based on score level fusion of finger biometrics", *Optik*, 125(23), pp. 6891–6897, 2014.

[205] Zahra S. Shariatmadar and Karim Faez, "Finger-Knuckle-Print recognition performance improvement via multi-instance fusion at the score level", *Optik*, 125 (3), pp. 908–910, 2014.

[206] Hiew Moi Sim, Hishammuddin Asmuni, Rohayanti Hassan, and Razib M. Othman, "Multimodal biometrics: Weighted score level fusion based on non-ideal iris and face imagesl", *Expert Systems With Applications*, 41(11), pp. 5390–5404, 2014.

[207] Mamta and Madasu Hanmandlu, "Multimodal biometric system built on the new entropy function for feature extraction and the Refined Scores as a classifier", *Expert Systems With Applications*, 42(7), pp. 3702–3723, 2015.

[208] M.M. Gupta and J. Qi, "Theory of T-norms and fuzzy inference methods", *Fuzzy Sets and Systems*, 40(3), pp. 431 – 450, 1991.

[209] Madasu Hanmandlu, Jyotsna Grover, Ankit Gureja, and H.M.Guptai, "Score level fusion of multimodel biometrics using triangular norms", *Pattern Recognition Letters*, 32, pp. 1843–1850, 2011.

[210] Lotfi A. Zadeh, "Outline of a New Approach to the Analysis of Complex Systems and Decision Processes", *IEEE Transactions on Systems, Man and Cybernetics*, 3(1), pp. 28–44, 1973.

[211] Wyllis Bandler and Ladislav Kohout, "Fuzzy power sets and fuzzy implication operators", *Fuzzy Sets and Systems*, 4(1), pp. 13 – 30, 1980.

[212] Siegfried Weber, "A general concept of fuzzy connectives, negations and implications based on T-norms and T-conorms", *Fuzzy Sets and Systems*, 11(1–3), pp. 103 – 113, 1983.

[213] Ronald R. Yager, "On a general class of fuzzy connectives", *Fuzzy Sets and Systems*, 4(3), pp. 235 – 242, 1980.

[214] Schweizer and Sklar, "Probabilistic Metric Spaces", *North-Holland, New York*, 1983.

[215] M.J. Frank, "On the simultaneous associativity of F(x, y) and x+yF(x, y)", *Aequationes Mathematicae*, 19(1), pp. 194–226, 1979.

[216] J. Dombi, “A general class of fuzzy operators, the demorgan class of fuzzy operators and fuzziness measures induced by fuzzy operators”, *Fuzzy Sets and Systems*, 8(2), pp. 149 – 163, 1982.

[217] Didier Dubois and Henri Prade, “New Results about Properties and Semantics of Fuzzy Set-Theoretic Operators”, In *Fuzzy Sets: Theory and Applications to Policy Analysis and Information Systems*, pp. 59–75, 1980.

[218] Yu Yandong, “Triangular norms and TNF-sigma-algebras”, *Fuzzy Sets and Systems*, 16(3), pp. 251 – 264, 1985.

[219] J. Aczél and C. Alsina, “Characterizations of some classes of quasilinear functions with applications to triangular norms and to synthesizing judgements”, *Aequationes Mathematicae*, 25(1), pp. 313–315, 1982.

[220] S. Lin and D. J. Costello, Error control coding, 1982, Prentice Hall.