IMAGE SPLICING DETECTION APPROACH BASED ON LOW DIMENSIONAL SVD FEATURES AND KERNEL PCA

ZAHRA MOGHADDASI

FACULTY OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY
UNIVERSITY OF MALAYA
KUALA LUMPUR

2017

IMAGE SPLICING DETECTION APPROACH BASED ON LOW DIMENSIONAL SVD FEATURES AND KERNEL PCA

ZAHRA MOGHADDASI

THESIS SUBMITTED IN FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

FACULTY OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY
UNIVERSITY OF MALAYA
KUALA LUMPUR

2017

UNIVERSITY OF MALAYA

ORIGINAL LITERARY WORK DECLARATION

Name of Candidate: ZAHRA MOGHADDASI

Matric No: WHA120019

Name of Degree: DOCTOR OF PHILOSOPHY

Title of Project Paper/Research Report/Dissertation/Thesis ("this Work"):

Image Splicing Detection Approach Based on Low Dimensional SVD Features and Kernel PCA

Field of Study: Information Security (Computer Science)

I do solemnly and sincerely declare that:

(1) I am the sole author/writer of this Work;
(2) This Work is original;
(3) Any use of any work in which copyright exists was done by way of fair dealing and for permitted purposes and any excerpt or extract from, or reference to or reproduction of any copyright work has been disclosed expressly and sufficiently and the title of the Work and its authorship have been acknowledged in this Work;
(4) I do not have any actual knowledge nor do I ought reasonably to know that the making of this work constitutes an infringement of any copyright work;
(5) I hereby assign all and every rights in the copyright to this Work to the University of Malaya ("UM"), who henceforth shall be owner of the copyright in this Work and that any reproduction or use in any form or by any means whatsoever is prohibited without the written consent of UM having been first had and obtained;
(6) I am fully aware that if in the course of making this Work I have infringed any copyright whether intentionally or otherwise, I may be subject to legal action or any other action as may be determined by UM.

Candidate's Signature Date:

Subscribed and solemnly declared before,

Witness's Signature Date:

Name: **Dr. Hamid A. Jalab**

Designation: Supervisor

ABSTRACT

Digital image forgery is becoming easier to perform because of the rapid development of various manipulation tools. Image splicing is one of the most common image forgery techniques. It is achieved simply by cutting a region from one or more images and pasting it, or them, into another image. This technique can cause inconsistencies in many features, such as an abnormally sharp transient at the splicing edges, and these inconsistencies are used to detect the forgery. To detect the spliced images several methods proposed utilizing the statistical features of the digital images. In this research, two efficient SVD-based feature extraction methods for image splicing detection are presented. In the first method, the natural Logarithm of inverse of each singular value is calculated. In the second method the concept of roughness measure is applied which is inversely proportional with condition number. Kernel Principal Component Analysis (PCA) is also applied as classifier feature selector to improve the classification process. And finally, support vector machine is used to distinguish between the authenticated and spliced images. The proposed methods are evaluated by applying three standard image datasets (DVMM v1, DVMM v2, and CASIA) in spatial and frequency domains. The first image dataset was the Columbia Image Splicing Detection Evaluation Dataset. This dataset contained 1845 gray-scale images (933 authentic images and 912 spliced images) in BMP format. The second image dataset is the Chinese Academy of Sciences, Institute of Automation (CASIA) with 1721 color images (800 authentic images and 921 spliced images). The third image dataset is DVMM v2, which contains 363 color images (183 authentic images and 180 spliced images). For the DVMM v1 image dataset, proposed method-1 shows an average accuracy of 98.78%. On the other hand, for CASIA image dataset, method-2 shows an average accuracy of 99.62%. Finally, with the DVMM v2 image dataset, both methods obtain an average accuracy of 100%, but in different color channels. These results outperform several current detection methods.

ABSTRAK

Pemalsuan digital imej menjadi lebih senang untuk dilaksanakan kerana terdapatnya penciptaan yang pesat bagi pelbagai jenis alatan manipulasi. Potongan imej adalah salah satu teknik pemalsuan yang biasa. Potongan imej mudah diperolehi dengan memotong sebahagian kawasan atau lebih pada gambar dan menampal padanya, atau menampal pada imej yang lain. Teknik ini boleh menyebabkan ketidakstabilan dalam pelbagai ciri, terutamanya keadaan yang tidak normal pada sisi potongan dan ketidakstabilan ini digunakan untuk mengesan pemalsuan. Bagi mengesan imej yang dipotong, beberapa kaedah dicadangkan seperti menggunakan ciri-ciri statistic pada digital imej. Di dalam kajian ini, dua kaedah pengekstrakan berasaskan SVD bagi mengesan pemotongan imej diebentangkan. Di dalam kaedah pertama, pengukuran nilai tunggal songsang dikira pada logaritma semujadi. Di dalam kaedah kedua, konsep ukuran kekasaran yang berkadar songsang dengan bilangan keadaan digunakan. Kernel Principal Component Analysis (PCA) juga digunakan sebagai ciri pengelas pra-pemprosesan untuk meningkatkan proses pengelasan. Akhir sekali, mesin sokongan vektor digunakan untuk membezakan antara imej yang sah dan imej yang terpotong. Kaedah yang dicadangkan dinilai dengan menggunakan tiga piawai set data iaitu (DVMM v1, v2 DVMM dan Casia) bagi bidang ruang dan kekerapan. Set data pertama ialah penilaian pengesanan imej Columbia. Data set ini mengandungi 1845 imej skala kelabu iaitu (933 imej sahih dan 912 imej disambungkan) dalam format BMP. Bagi set data kedua ialah dari Akademi Sains China, Institut Automation (Casia) dengan 1721 imej warna (800 imej sahih dan 921 imej disambungkan). Data set ketiga adalah adalah DVMM v2, yang mengandungi 363 imej warna (183 imej sahih dan 180 imej disambung). Bagi set data imej DVMM v1, kaedah-1 yang dicadangkan menunjukkan ketepatan purata 98,78%. Sebaliknya, untuk Casia set data imej, menunjukkan kaedah-2 mendapat ketepatan purata 99,62%. Akhir sekali, dengan set data imej DVMM v2, kedua-dua kaedah mendapat ketepatan purata 100%,

tetapi didalam saluran warna yang berbeza. Keputusan ini mengatasi beberapa kaedah pengesanan semasa.

ACKNOWLEDGEMENTS

In the name of Allah, Most Gracious, Most Merciful. I thank Allah S.W.T for granting me perseverance and strength I needed to complete this thesis.

I would like to express a great thankfulness to my supervisor, Dr. Hamid Abdullah Jalab as well as my co-supervisor, Associate Prof. Dr. Rafidah Binti Md Noor for their support, guidance, suggestions and encouragement over the past years of this research.

I would like to thank the Faculty of Computer Science and Information Technology, University of Malaya, and UMRG for providing me a great academic environment.

I would like to express a special word of thanks to my parents for their faith, support and encouragement. I also offer my regards and blessings to all of those who supported me in any respect during the completion of my research.

TABLE OF CONTENTS

Abstract iii

Abstrak iv

Acknowledgements vi

Table of Contents vii

List of Figures xiii

List of Tables xvii

List of Symbols and Abbreviations xxiii

CHAPTER 1: INTRODUCTION 1

1.1 Research Inspiration and Background 1

1.2 Problem Statement 9

1.3 Research Questions 11

1.4 Research Aim and Objectives 11

1.5 Scope of Work 12

1.6 Research Contributions 13

1.7 Thesis Outline 14

CHAPTER 2: LITERATURE REVIEW 16

2.1 Introduction 16

2.2 Digital Image Formation 16

2.2.1 Discrete Fourier Transform (DFT) 21

2.2.2 Discrete Cosine Transform (DCT) 24

2.2.2.1 Decorrelation 25

2.2.2.2 Separability 25

2.2.2.3 Symmetry 26

2.2.3 Discrete Wavelet Transform (DWT) 27
2.3 Forensics 29
2.3.1 Analog Forensics 30
2.3.2 Computer Forensics 32
2.3.3 Multimedia Forensics 35
2.4 Digital Image Forgery Creation 35
2.5 Digital Image Forgery Detection 36
2.5.1 Active Methods 38
2.5.2 Passive Methods 40
2.5.2.1 Source device authentication 41
2.5.2.2 Tamper Detection 42
2.6 Feature Extraction 47
2.6.1 Singular Value Decomposition 48
2.6.2 SVD Image Properties 49
2.6.2.1 SVD Subspaces 50
2.6.2.2 SVD architecture 50
2.6.2.3 PCA versus SVD 51
2.6.2.4 SVD Multiresolution 51
2.6.2.5 SVD Oriented Energy 52
2.6.2.6 SVD and Linear Independence 52
2.6.2.7 SVD-based orthogonal subspaces and rank approximation 53
2.6.3 Feature Extraction Based on SVD 53
2.6.4 Why SVD? 56
2.7 Feature Selection 57
2.7.1 PCA 58
2.7.2 Kernel PCA 59

2.8 Classification 61
2.9 Image Splicing detection methods 62
2.9.1 Markov-based algorithms 64
2.9.2 Moment-based algorithms 67
2.9.3 Hilbert-Huang Transfer (HHT) 71
2.9.4 Run-Length (RL) Algorithms 73
2.9.5 Co-occurrence Matrix 75
2.9.6 Phase Congruency 76
2.9.7 Three order methods 78
2.9.8 Local Binary Pattern (LBP) 80
2.9.9 Fractional Differential Texture Descriptors 82
2.10 Summary 84

CHAPTER 3: RESEARCH METHODOLOGY 89

3.1 Introduction 89
3.2 Research Phases 89
3.3 Requirement Collection and Analysis 91
3.3.1 Assessment and study of existing methods 92
3.3.2 Scope of System 92
3.3.2.1 Software 92
3.3.2.2 Hardware 93
3.3.2.3 Data 93
3.3.3 Assumptions and Hypothesis 95
3.4 Structures of the proposed methods 96
3.5 Summary 100

CHAPTER 4: RESEARCH DESIGN AND IMPLEMENTATION 101

4.1 Introduction 101

4.2 Image Preprocessing 101

4.2.1 Color Spaces 102

4.2.2 Multi-Size Blocking 105

4.2.3 Frequency Domain 106

4.3 Feature Extraction 109

4.3.1 Proposed Method 1: SVD-Log 110

4.3.2 Proposed Method 2: SVD-CN 112

4.4 Feature Selection 115

4.5 Classification 119

4.6 Computational Complexity of the Proposed Methods 123

4.7 Summary 126

CHAPTER 5: EXPERIMENTAL RESULTS AND DISCUSSION 127

5.1 Introduction 127

5.2 Classification Performance of DVMM v1.0 Image Dataset 128

5.2.1 Classification Performance of SVD-Log (Grayscale) 128

5.2.2 Classification Performance of SVD-CN (Gray-Scale) 132

5.3 Classification Performance of CASIA Image Dataset 135

5.3.1 Classification Performance of SVD-Log (Grayscale) 136

5.3.2 Classification Performance of SVD-CN (Grayscale) 138

5.3.3 Classification Performance of SVD-Log (Red Channel) 140

5.3.4 Classification Performance of SVD-CN (Red Channel) 142

5.3.5 Classification Performance of SVD-Log (Green Channel) 145

5.3.6 Classification Performance of SVD-CN (Green Channel) 147

5.3.7 Classification Performance of SVD-Log (Blue Channel) 149

5.3.8 Classification Performance of SVD-CN (Blue Channel) 150
5.3.9 Classification Performance of SVD-Log (Y Channel) 152
5.3.10 Classification Performance of SVD-CN (Y Channel) 153
5.3.11 Classification Performance of SVD-Log (Cb Channel) 155
5.3.12 Classification Performance of SVD-CN (Cb Channel) 156
5.3.13 Classification Performance of SVD-Log (Cr Channel) 158
5.3.14 Classification Performance of SVD-CN (Cr Channel) 159
5.3.15 Classification Performance of SVD-Log (RGB color model) 161
5.3.16 Classification Performance of SVD-CN (RGB Color model) 162
5.3.17 Classification Performance of SVD-Log (YCbCr Color Model) 164
5.3.18 Classification Performance of SVD-CN (YCbCr Color Model) 166
5.4 DVMM v2.0 Image Dataset 168
5.4.1 Classification Performance of SVD-Log (Grayscale) 168
5.4.2 Classification Performance of SVD-CN (Grayscale) 170
5.4.3 Classification Performance of SVD-Log (Red Channel) 172
5.4.4 Classification Performance of SVD-CN (Red Channel) 173
5.4.5 Classification Performance of SVD-Log (Green Channel) 175
5.4.6 Classification Performance of SVD-CN (Green Channel) 176
5.4.7 Classification Performance of SVD-Log (Blue Channel) 178
5.4.8 Classification Performance of SVD-CN (Blue Channel) 179
5.4.9 Classification Performance of SVD-Log (Y Channel) 181
5.4.10 Classification Performance of SVD-CN (Y Channel) 182
5.4.11 Classification Performance of SVD-Log (Cb Channel) 184
5.4.12 Classification Performance of SVD-CN (Cb Channel) 185
5.4.13 Classification Performance of SVD-Log (Cr Channel) 187
5.4.14 Classification Performance of SVD-CN (Cr Channel) 188

5.4.15 Classification Performance of SVD-Log (RGB color model) 190
5.4.16 Classification Performance of SVD-CN (RGB Color model) 191
5.4.17 Classification Performance of SVD-Log (YCbCr Color Model)..... 193
5.4.18 Classification Performance of SVD-CN (YCbCr Color Model)..... 194
5.5 Comparison between detection performance of SVD-Log and SVD-CN..... 196
5.6 Comparison with Conventional Image Splicing Methods..... 198
5.7 Summary..... 203

CHAPTER 6: CONCLUSION AND FUTURE WORK 204

6.1 Introduction..... 204
6.2 Research Findings and Achievements..... 204
6.3 Conclusion..... 207
6.4 Future Work and Directions 210
References..... 211
List of Publications and Papers Presented 223

LIST OF FIGURES

Figure 1.1: Montage (1939) of Queen Elizabeth and Canadian Prime Minister William Lyon Mackenzie King photo in Banff, Alberta, King George VI was removed from the original photograph. 1

Figure 1.2: Montage (2003) of a British soldier in Iraq trying to control a crowd of civilians in a passionate manner. Credits to Brian Walski. 2

Figure 1.3: Montage (2004) of John Kerry and Jane Fonda standing together at a podium during a 1970s anti-war rally. Credits to Ken Light (top-left), Associated Press (bottom), and Owen Franken (top-right). 3

Figure 2.1: Sample of Gray-scale image 17

Figure 2.2: Sample of Indexed image 19

Figure 2.3: Sample of true color image 20

Figure 2.4: Applying the 2D discrete Fourier transform 21

Figure 2.5: Rearranging the 2D DFT for display purposes 22

Figure 2.6: Normalized autocorrelation of an image before and after DCT 25

Figure 2.7: 2-D DCT computation using separability property 26

Figure 2.8: Two level multi-resolution wavelet decomposition of an image 27

Figure 2.9: Domains of Forensics Science 30

Figure 2.10: Classification of image forgery detection technique 38

Figure 2.11: Generic Watermark insertion 39

Figure 2.12: Watermark Extraction 39

Figure 2.13: General Identification Pattern 42

Figure 2.14: General Image Detection Schemes 43

Figure 2.15: Sample of image copy-move, (a) original image, (b) image after copy-move attack, (c) copied portions 44

Figure 2.16: Sample of Image Retouching (a) Original Image, (b) Retouched Image ... 45

Figure 2.17: Sample of Geometrical Transformation. (a) Original Image, (b) Image after geometrical transformation 46

Figure 2.18: Sample of image splicing. (a) first original image, (b) second original image, (c) spliced image 46

Figure 2.19: Moment extraction feature, (a) general block diagram, (b) prediction context, (c) prediction error 2-D array generation 68

Figure 2.20: LBP Computation Process 80

Figure 3.1: Operational Framework 91

Figure 3.2: Example of DVMM v1 Image Dataset 94

Figure 3.3: Example of CASIA Image Dataset 95

Figure 3.4: Example of DVMM v2 Image Dataset 95

Figure 3.5: General Steps of The Proposed Methods 98

Figure 3.6: First Level Design of the Proposed Methods in Spatial (a) and Frequency (b) domains 99

Figure 4.1: RGB and YCbCr color spaces 103

Figure 4.2: Image in RGB and its edges in different channels, (a) RGB Image, (b) Red channel, (c) Green channel, (d) Blue channel, (e) RGB Image, (f) edges in Red channel, (g) edges in Green channel, (h) edges in Blue channel 104

Figure 4.3: Image in YCbCr and its edges in different channels, (a) YCbCr Image, (b) Y channel, (c) Cb channel, (d) Cr channel, (e) YCbCr Image, (f) edges in Y channel, (g) edges in Cb channel, (h) edges in Cr channel 105

Figure 4.4: Mulit-Block diagram with $N \times N$ Block sizes, $N \in [3 \ldots 27]$ 106

Figure 4.5: Flowchart of image preprocessing phase 108

Figure 4.6: Pseudo-code for Image Preprocessing 109

Figure 4.7: Pseudo-code for the proposed method 1 (SVD-Log) feature extraction 111

Figure 4.8: (a) Original Image, (b) Spliced Image, (c) Graph of singular values for images (a) and (b) 113

Figure 4.9: Pseudo-code for the proposed method 2 (SVD-CN) feature extraction 114

Figure 4.10: Flowchart of feature extraction phase 115

Figure 4.11: Standard deviation of the extracted features for image dataset DVMM v1 117

Figure 4.12: Standard deviation of the extracted features for image dataset CASIA, Red channel 117

Figure 4.13: Standard deviation of the extracted features for image dataset DVMM v2, Red channel 118

Figure 4.14: Flowchart of feature selection phase 119

Figure 4.15: Pseudo-code for feature selection phase 119

Figure 4.16: Flowchart of classification phase 122

Figure 4.17: Pseudo-code for classification phase 123

Figure 5.1: Comparison of Detection Performance between features in 100-D, with PCA in 90-D, and Kernel PCA in 50-D using SVD-Log 131

Figure 5.2: Comparison of Detection Performance between features in 100-D, with PCA in 90-D, and Kernel PCA in 90-D using SVD-CN 135

Figure 5.3: Comparison of Detection Performance between features in 100-D, Kernel PCA in 90-D, and Kernel PCA in 50-D using SVD-Log & SVD-CN (Gray-scale) 140

Figure 5.4: Comparison of Detection Performance between features in 100-D, Kernel PCA in 80-D, and Kernel PCA in 50-D using SVD-Log & SVD-CN (Red Channel) . 145

Figure 5.5: Comparison of Detection Performance between features in 100-D, Kernel PCA in 30-D, and Kernel PCA in 60-D using SVD-Log & SVD-CN (Green Channel) 149

Figure 5.6: Comparison of Detection Performance between features in 100-D, Kernel PCA in 70-D, and Kernel PCA in 40-D using SVD-Log & SVD-CN (Blue Channel) 152

Figure 5.7: Comparison of Detection Performance between features in 100-D, Kernel PCA in 40-D, and Kernel PCA in 50-D using SVD-Log & SVD-CN (Y Channel)..... 155

Figure 5.8: Comparison of Detection Performance between features in 100-D, Kernel PCA in 90-D, and Kernel PCA in 80-D using SVD-Log & SVD-CN (Cb Channel)... 158

Figure 5.9: Comparison of Detection Performance between features in 100-D, Kernel PCA in 60-D, and Kernel PCA in 60-D using SVD-Log & SVD-CN (Cr Channel) ... 161

Figure 5.10: Comparison of Detection Performance between features in 100-D, Kernel PCA in 20-D, and Kernel PCA in 60-D using SVD-Log & SVD-CN (RGB Channel) 164

Figure 5.11: Comparison of Detection Performance between features in 100-D, Kernel PCA in 70-D, and Kernel PCA in 10-D using SVD-Log & SVD-CN (YCbCr Channel) 168

Figure 5.12: Comparison of Detection Performance between features in 100-D, Kernel PCA in 70-D, and Kernel PCA in 10-D using SVD-Log & SVD-CN (Gray-scale) 172

Figure 5.13: Comparison of Detection Performance between features in 100-D, Kernel PCA in 70-D, and Kernel PCA in 10-D using SVD-Log & SVD-CN (Red Channel) . 175

Figure 5.14: Comparison of Detection Performance between features in 100-D, Kernel PCA in 20-D, and Kernel PCA in 90-D using SVD-Log & SVD-CN (Green Channel) 178

Figure 5.15: Comparison of Detection Performance between features in 100-D, Kernel PCA in 20-D, and Kernel PCA in 10-D using SVD-Log & SVD-CN (Blue Channel) 181

Figure 5.16: Comparison of Detection Performance between features in 100-D, Kernel PCA in 10-D, and Kernel PCA in 10-D using SVD-Log & SVD-CN (Y Channel)..... 184

Figure 5.17: Comparison of Detection Performance between features in 100-D, Kernel PCA in 10-D, and Kernel PCA in 20-D using SVD-Log & SVD-CN (Cb Channel)... 187

Figure 5.18: Comparison of Detection Performance between features in 100-D, Kernel PCA in 10-D, and Kernel PCA in 90-D using SVD-Log & SVD-CN (Cr Channel) ... 190

Figure 5.19: Comparison of Detection Performance between features in 100-D, Kernel PCA in 30-D, and Kernel PCA in 30-D using SVD-Log & SVD-CN (RGB Color Model) 193

Figure 5.20: Comparison of Detection Performance between features in 100-D, Kernel PCA in 20-D, and Kernel PCA in 10-D using SVD-Log & SVD-CN (YCbCr Color Model) 196

Figure 5.21: Comparison between best detection accuracies of SVD-Log and SVD-CN in DVMM v1 image dataset 197

Figure 5.22: Comparison between best detection accuracies of SVD-Log and SVD-CN in CASIA image dataset 197

Figure 5.23: Comparison between best detection accuracies of SVD-Log and SVD-CN in DVMM v2 image dataset 198

LIST OF TABLES

Table 2.1: Comparison between the reviewed image splicing detection methods 85

Table 2.2: Continued, comparison between the reviewed image splicing detection methods 86

Table 2.3: Continued, comparison between the reviewed image splicing detection methods 87

Table 2.4: Continued, comparison between the reviewed image splicing detection methods 88

Table 4.1: Runtime of feature extraction applying the proposed methods 126

Table 5.1: Detection accuracy of SVD-Log with the original dimension of 100 in Spatial and Frequency Domains 128

Table 5.2: Detection accuracy of SVD-Log with PCA in different dimensions in Spatial and Frequency Domains 129

Table 5.3: Detection accuracy of SVD-Log with Kernel PCA in different dimensions in Spatial and Frequency Domains 131

Table 5.4: Detection accuracy of SVD-CN with the original dimension of 100 in Spatial and Frequency Domains 132

Table 5.5: Detection accuracy of SVD-Log with PCA in different dimensions in Spatial and Frequency Domains 133

Table 5.6: Detection accuracy of SVD-Log with Kernel PCA in different dimensions in Spatial and Frequency Domains 134

Table 5.7: Detection accuracy of SVD-Log with the original dimension of 100 in Spatial and Frequency Domains 136

Table 5.8: Detection accuracy of SVD-Log with PCA in different dimensions in Spatial and Frequency Domains 137

Table 5.9: Detection accuracy of SVD-Log with Kernel PCA in different dimensions in Spatial and Frequency Domains 137

Table 5.10: Detection accuracy of SVD-CN with the original dimension of 100 in Spatial and Frequency Domains 138

Table 5.11: Detection accuracy of SVD-CN with PCA in different dimensions in Spatial and Frequency Domains 139

Table 5.12: Detection accuracy of SVD-CN with Kernel PCA in different dimensions in Spatial and Frequency Domains.....139

Table 5.13: Detection accuracy of SVD-Log with the original dimension of 100 in Spatial and Frequency Domains.....141

Table 5.14: Detection accuracy of SVD-Log with PCA in different dimensions in Spatial and Frequency Domains.....141

Table 5.15: Detection accuracy of SVD-Log with Kernel PCA in different dimensions in Spatial and Frequency Domains.....142

Table 5.16: Detection accuracy of SVD-CN with the original dimension of 100 in Spatial and Frequency Domains.....143

Table 5.17: Detection accuracy of SVD-CN with PCA in different dimensions in Spatial and Frequency Domains.....143

Table 5.18: Detection accuracy of SVD-CN with Kernel PCA in different dimensions in Spatial and Frequency Domains.....144

Table 5.19: Detection accuracy of SVD-Log with the original dimension of 100 in Spatial and Frequency Domains.....146

Table 5.20: Detection accuracy of SVD-Log with Kernel PCA in Frequency Domain146

Table 5.21: Detection accuracy of SVD-CN with the original dimension of 100 in Spatial and Frequency Domains.....147

Table 5.22: Detection accuracy of SVD-CN with Kernel PCA in different dimensions in Spatial and Frequency Domains.....148

Table 5.23: Detection accuracy of SVD-Log with the original dimension of 100 in Spatial and Frequency Domains.....149

Table 5.24: Detection accuracy of SVD-Log with Kernel PCA in Frequency Domain150

Table 5.25: Detection accuracy of SVD-CN with the original dimension of 100 in Spatial and Frequency Domains.....151

Table 5.26: Detection accuracy of SVD-CN with Kernel PCA in different dimensions in Spatial and Frequency Domains.....151

Table 5.27: Detection accuracy of SVD-Log with the original dimension of 100 in Spatial and Frequency Domains.....153

Table 5.28: Detection accuracy of SVD-Log with Kernel PCA in Frequency Domain153

Table 5.29: Detection accuracy of SVD-CN with the original dimension of 100 in Spatial and Frequency Domains....154

Table 5.30: Detection accuracy of SVD-CN with Kernel PCA in different dimensions in Spatial and Frequency Domains....154

Table 5.31: Detection accuracy of SVD-Log with the original dimension of 100 in Spatial and Frequency Domains....156

Table 5.32: Detection accuracy of SVD-Log with Kernel PCA in Frequency Domain156

Table 5.33: Detection accuracy of SVD-CN with the original dimension of 100 in Spatial and Frequency Domains....157

Table 5.34: Detection accuracy of SVD-CN with Kernel PCA in different dimensions in Spatial and Frequency Domains....157

Table 5.35: Detection accuracy of SVD-Log with the original dimension of 100 in Spatial and Frequency Domains....159

Table 5.36: Detection accuracy of SVD-Log with Kernel PCA in Frequency Domain159

Table 5.37: Detection accuracy of SVD-CN with the original dimension of 100 in Spatial and Frequency Domains....160

Table 5.38: Detection accuracy of SVD-CN with Kernel PCA in different dimensions in Spatial and Frequency Domains....160

Table 5.39: Detection accuracy of SVD-Log with the original dimension of 100 in Spatial and Frequency Domains....162

Table 5.40: Detection accuracy of SVD-Log with Kernel PCA in Frequency Domain162

Table 5.41: Detection accuracy of SVD-CN with the original dimension of 100 in Spatial and Frequency Domains....163

Table 5.42: Detection accuracy of SVD-CN with Kernel PCA in different dimensions in Spatial and Frequency Domains....163

Table 5.43: Detection accuracy of SVD-Log with the original dimension of 100 in Spatial and Frequency Domains....165

Table 5.44: Detection accuracy of SVD-Log with Kernel PCA in Frequency Domain165

Table 5.45: Detection accuracy of SVD-CN with the original dimension of 100 in Spatial and Frequency Domains....166

Table 5.46: Detection accuracy of SVD-CN with Kernel PCA in different dimensions in Spatial and Frequency Domains.....167

Table 5.47: Detection accuracy of SVD-Log with the original dimension of 100 in Spatial and Frequency Domains.....169

Table 5.48: Detection accuracy of SVD-Log with Kernel PCA in Frequency Domain169

Table 5.49: Detection accuracy of SVD-CN with the original dimension of 100 in Spatial and Frequency Domains.....170

Table 5.50: Detection accuracy of SVD-CN with Kernel PCA in different dimensions in Spatial and Frequency Domains.....171

Table 5.51: Detection accuracy of SVD-Log with the original dimension of 100 in Spatial and Frequency Domains.....173

Table 5.52: Detection accuracy of SVD-Log with Kernel PCA in Frequency Domain173

Table 5.53: Detection accuracy of SVD-CN with the original dimension of 100 in Spatial and Frequency Domains.....174

Table 5.54: Detection accuracy of SVD-CN with Kernel PCA in different dimensions in Spatial and Frequency Domains.....174

Table 5.55: Detection accuracy of SVD-Log with the original dimension of 100 in Spatial and Frequency Domains.....175

Table 5.56: Detection accuracy of SVD-Log with Kernel PCA in Frequency Domain176

Table 5.57: Detection accuracy of SVD-CN with the original dimension of 100 in Spatial and Frequency Domains.....177

Table 5.58: Detection accuracy of SVD-CN with Kernel PCA in different dimensions in Spatial and Frequency Domains.....177

Table 5.59: Detection accuracy of SVD-Log with the original dimension of 100 in Spatial and Frequency Domains.....178

Table 5.60: Detection accuracy of SVD-Log with Kernel PCA in Frequency Domain179

Table 5.61: Detection accuracy of SVD-CN with the original dimension of 100 in Spatial and Frequency Domains.....180

Table 5.62: Detection accuracy of SVD-CN with Kernel PCA in different dimensions in Spatial and Frequency Domains.....180

Table 5.63: Detection accuracy of SVD-Log with the original dimension of 100 in Spatial and Frequency Domains.....181

Table 5.64: Detection accuracy of SVD-Log with Kernel PCA in Frequency Domain182

Table 5.65: Detection accuracy of SVD-CN with the original dimension of 100 in Spatial and Frequency Domains.....183

Table 5.66: Detection accuracy of SVD-CN with Kernel PCA in different dimensions in Spatial and Frequency Domains.....183

Table 5.67: Detection accuracy of SVD-Log with the original dimension of 100 in Spatial and Frequency Domains.....185

Table 5.68: Detection accuracy of SVD-Log with Kernel PCA in Frequency Domain185

Table 5.69: Detection accuracy of SVD-CN with the original dimension of 100 in Spatial and Frequency Domains.....186

Table 5.70: Detection accuracy of SVD-CN with Kernel PCA in different dimensions in Spatial and Frequency Domains.....186

Table 5.71: Detection accuracy of SVD-Log with the original dimension of 100 in Spatial and Frequency Domains.....187

Table 5.72: Detection accuracy of SVD-Log with Kernel PCA in Frequency Domain188

Table 5.73: Detection accuracy of SVD-CN with the original dimension of 100 in Spatial and Frequency Domains.....189

Table 5.74: Detection accuracy of SVD-CN with Kernel PCA in different dimensions in Spatial and Frequency Domains.....189

Table 5.75: Detection accuracy of SVD-Log with the original dimension of 100 in Spatial and Frequency Domains.....190

Table 5.76: Detection accuracy of SVD-Log with Kernel PCA in Frequency Domain191

Table 5.77: Detection accuracy of SVD-CN with the original dimension of 100 in Spatial and Frequency Domains.....192

Table 5.78: Detection accuracy of SVD-CN with Kernel PCA in different dimensions in Spatial and Frequency Domains.....192

Table 5.79: Detection accuracy of SVD-Log with the original dimension of 100 in Spatial and Frequency Domains.....193

Table 5.80: Detection accuracy of SVD-Log with Kernel PCA in Frequency Domain194

Table 5.81: Detection accuracy of SVD-CN with the original dimension of 100 in Spatial and Frequency Domains 195

Table 5.82: Detection accuracy of SVD-CN with Kernel PCA in different dimensions in Spatial and Frequency Domains 195

Table 5.83: Comparison of the proposed methods with the existing image splicing detection methods based on DVMM v1 200

Table 5.84: Continued, comparison of the proposed methods with the existing image splicing detection methods based on CASIA 201

Table 5.85: Continued, comparison of the proposed methods with the existing image splicing detection methods based on DVMM v2 202

LIST OF SYMBOLS AND ABBREVIATIONS

AUC	:	Area Under Curve
CASIA	:	Chinese Academy of Sciences, Institute of Automation
CN	:	Condition Number
DCT	:	Discrete Cosine Transform
DFT	:	Discrete Fourier Transform
DIP	:	Digital Image Processing
DVMM	:	Digital Video MultiMedia
DWT	:	Discrete Wavelet Transform
FN	:	False Negative
FP	:	False Positive
KPCA	:	Kernel Principal Component Analysis
LBP	:	Local Binary Pattern
Log	:	Logarithm
PCA	:	Principal Component Analysis
RGB	:	Red, Green, and Blue Color Space
SVD	:	Singular Value Decomposition
SVM	:	Support Vector Machine
TN	:	True Negative
TP	:	True Positive
TNR	:	True Negative Rate
TPR	:	True Positive Rate
YCbCr	:	Luma (Y) and two chrominance (CbCr) components Color Space

CHAPTER 1: INTRODUCTION

1.1 Research Inspiration and Background

For many decades, photography has served a vital function in people's lives, and is considered as being one of the most important revolutions in terms of recording moments. However, people have begun to lose trust in photographs, because of the increasing use of manipulation tools, which have made photographs less trustworthy. The history of image tampering (i.e. creating photographs that never happened in real life) is as old as the art of photography itself. Images have been manipulated for malicious purposes in many instances.

In 1939, a photo (Figure 1.1) of Queen Elizabeth, Canadian Prime Minister William Lyon Mackenzie and King George VI was taken in Banff, Alberta. However, on an election poster for the Prime Minister which made use of the photo, King George VI was removed. It has been assumed that the Prime Minister had the photo modified, since a photo of him alone with the Queen could portray him in a stronger position (Farid, 2008).

Figure 1.1: Montage (1939) of Queen Elizabeth and Canadian Prime Minister William Lyon Mackenzie King photo in Banff, Alberta, King George VI was removed from the original photograph**.**

In 2003, the *Los Angeles Times* printed an image from photojournalist Brian Walski on its front cover, which showed a British soldier in Iraq trying to control a crowd of civilians in a compassionate manner. However, the moment as depicted never happened. The photograph was a composite of two different photographs, which had been merged to create an appealing image (Figure 1.2). The image tampering was discovered, and Walski was fired (Rocha et al., 2011).

Figure 1.2: Montage (2003) of a British soldier in Iraq trying to control a crowd of civilians in a passionate manner. Credits to Brian Walski.

In the 2004 US presidential campaign, many allies of John Kerry were surprised to see a photo-montage which was published in many newspapers, showing Jane Fonda and Kerry together at the podium of an anti-Vietnam War rally back in 1970. In fact, all of the photos were fake. The photo of Kerry has been taken at an anti-war rally in 1971, in Mineola, New York, by a photographer named Ken Light. The picture of Fonda came from a speech in 1972, in Miami Beach, Florida, and was taken by different photographer named Owen Franken (Figure 1.3) (Rocha et al., 2011).

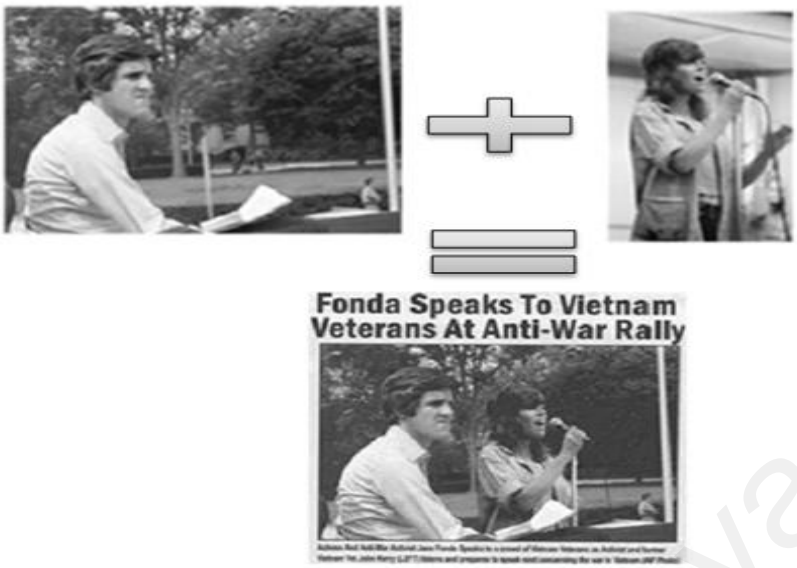


Figure 1.3: Montage (2004) of John Kerry and Jane Fonda standing together at a podium during a 1970s anti-war rally. Credits to Ken Light (top-left), Associated Press (bottom), and Owen Franken (top-right).

In 2009, Brazilian newspaper *Folha de Sao Paulo* published an article, including one spliced image generated from various greyscale images, to illustrate how the Brazilian Chief of Staff had actively contributed in the resistance against the military regime (Rocha et al., 2011).

Such episodes helped to generate many questions regarding the use of digital photos as evidence. According to the *Wall Street Journal*, 10% of all color photos appearing in the U.S had in fact been retouched or otherwise modified (Amsberry, 1989). Additionally, the scientific community is an interesting field in terms of forgeries (Farid, 2006b) (Pearson, 2005). A confirmation of an image's authenticity is often required, before the contents of the image can be relied upon. Due to this, images have popularly been utilized as evidence and also as historical records. The number and range of photography's applications in forensic investigation, criminal investigation, journalism, law

enforcement, medical imaging and insurance claims has expanded considerably over time (Birajdar & Mankar, 2013).

An innovative research area called image forgery detection emerged from the necessity to regain the trustworthiness of photographs. This field aims to verify the authenticity of a digital image. Image forgery detection is generally categorized into two basic groups: active methods and passive methods. The methods of active forgery detection, for example digital signatures and digital watermarking, employ a known code of authentication inserted into the photo content before the photos are sent through unreliable public channels. By confirming the existence of such authentication codes, the authentication can be verified by making comparisons with the code which was inserted originally. However, this approach needs specific software or hardware to insert authentication code within the photo before it is published.

The approach of blind or passive forgery detection employs the received photo only for evaluating its integrity or authenticity, with no watermark or signature of the original photo from the senders. It works according to the consideration that, even though digital forgeries might not display any visual signs of being tampered with, the image consistency or underlying statistical properties of a natural photo may have been manipulated, presenting new artifacts leading to various types of inconsistencies. Such inconsistencies could therefore be used to search for forgery. This method is in common use, since it does not require any prior image information. The existing methods in this field identify different tampering traces and detect them separately, along with the tampered region's location (Birajdar & Mankar, 2013).

Basically, these passive methods respond to two main questions regarding the image history:

a. Was the photo captured by the device which it was claimed to be captured with?
b. Does the photo still depict all of its original content?

The first question is one of the most significant points of interest when the image source is itself a piece of evidence – for example when the ownership of the camera is compromising or whether the content has been uniquely recorded by a certain device, for example in video surveillance. The other question is more interesting in general and could be applied directly to cases of image forgery. Providing answers to such questions is easiest when the original photo is readily available. Regarding a potentially fake image, the availability of the original image is enough to demonstrate forgery. In practice, however, there is often little or no information about the original image. Thus, investigators are usually forced to authenticate the history of images with a blind approach (Redi et al., 2011).

The goal of digital image forensics (DIF) is to provide supporting tools for blind study. Such new disciplines come from current security-related multimedia study domains, such as Steganography and Watermarking, and also exploit photo-processing and analysis devices to recover information regarding the image history.

Two main study approaches are utilized in Digital Image Forensics. The first one contains the approaches which try to answer the question (a) above. Through applying certain analysis, the aim is to recognize the specific image-capturing device or to determine which device(s) could not have captured it.

These approaches are considered to be the image-source tool identification methods. The other group of techniques attempts instead to show traces of semantic modification (Forgeries) through reviewing the inconsistencies in the natural statistics of the image.

Such approaches are known as tampering detection methods. Both the DIF domain and these fields are generally attracting a rising interest from scientific researchers (Redi et al., 2011).

The content of the image can be modified in two main operations: first, by omitting information, or by adding it. Usually in order to remove information, forgers will not need to use another image. In contrast, a usual method to add more information to an image is by utilizing the extracted material from one or more separate images. However, it should be noted that such operations are not always essential, since simple image processing can provide relevant modification of both the pragmatics and semantics of a photo.

Tampering approaches and their intentions, therefore, can be categorized into two groups: those that create forgery by working on one particular image, such as copy-move; and those which access the content of further images, such as splicing. In fact, image splicing is considered one of the most popular and simple schemes for image tampering. It includes the replacement of photo fragments from one or more various images into other images. The detection of image splicing is also very important in image forgery detection (Redi et al., 2011).

Studies were conducted on this topic, and several image splicing detection methods have been proposed and developed. They usually follow a general framework that includes the following main phases:

1. **Image preprocessing:** before feature extraction, different operations will be performed over all of the images, for example RGB image transformation into grayscale, cropping, or DWT or DCT transformation to improve the classification performance.

2. **Feature Extraction:** a group of features are extracted for each single class, which helps to differentiate it from other existing classes. In particular, informative features of the images are extracted and selected which are highly sensitive to image manipulation. One of the preferred attributes of the chosen features and the constructed vector is to be low-dimensional, which can minimize the complexity level of training computation, as well as classification.

3. **Feature selection and classifier selection:** according to the extracted group of features, an appropriate classifier should be selected or designed. A broad category of images should be selected in order to train the classifiers and achieve some critical classifier parameters that can be used for classification. Also, feature selection is utilized to minimize feature dimensionality without reducing the based classification performance of machine learning in parallel with computational complexity reduction (Sutthiwan et al., 2009).

4. **Classification:** the goal of classification is to distinguish the images and then categorize them as two different groups: authentic and forged images (Birajdar & Mankar, 2013).

Consequently, an image splicing detection model that can better reflect image splicing artifacts with lower dimensionality and higher detection accuracy in different image datasets will be more efficient. Existing methods mainly concentrate on the feature extraction step and develop different high-dimensional image splicing detection techniques. However, handling the extracted high-dimensional and redundant features

can be a difficult and time-consuming process. The detection accuracy obtained is also very low in some of the algorithms or is high in only one image dataset or in specific color channels.

Ng et al. (Tian-Tsong Ng et al., 2004a) developed an image-splicing detection model based on the use of bicoherence magnitude and phase features, with the assumption that the image splicing procedure is nonlinear and that the image involved is non-stationary. Their expanded method had 768 dimensions and achieved an unsatisfactory detection performance of 72%, since more recent methods can detect spliced images with higher detection accuracy.

To improve detection rate, Fu et al. (Fu et al., 2006) proposed an image splicing detection approach using the Hilbert–Huang transform (HHT). They considered the high nonlinearity and non-stationary nature of the image splicing operation and merged this technique with a statistical natural image model on the basis of moments of characteristic functions with wavelet decomposition. Their method, with only 110 dimensions obtained an accuracy rate of 80.15%.

Due to unsatisfactory detection rates obtained from proposed splicing detection methods, in 2007, Shi et al. (Shi et al., 2007b) proposed a blind, passive, and natural image splicing detection model based on statistical feature extraction methods. Their model includes Markov transition probabilities and moments of characteristic functions of wavelet sub-bands applied to different 2D arrays and to the 2D arrays of multi-size block discrete cosine transform. Their model achieved a detection accuracy rate of 91.8% with 266 dimensions, which was a promising improvement in image splicing detection.

He et al. (He et al., 2012) expanded the natural Markov-based model applied in (Shi et al., 2007b) by capturing the inter-block and intra-block correlation between the block

DCT coefficients proposed in (C. Chen & Shi, 2008) and merging it with the features generated from the DWT domain. Their model has a very high-dimensionality of 6966. They reduced their features by applying a feature selection method called support vector machine recursive feature elimination (SVM-RFE), which achieved a detection rate of 93.55% over a digital video multimedia (DVMM) image dataset (Tian-Tsong Ng et al., 2004c).

Previous image splicing detection methods were mostly developed for greyscale images, though in 2011 Zhao et al. (Zhao et al., 2011) proposed applying the run-length run-number (RLRN) vectors in four directions of the chroma spaces, since detecting spliced images in one color space is difficult. Their results showed that the features extracted from the chroma channels (94.7%) were more accurate than those extracted from the R, G, B, and Y channels. Their feature extraction model with only 60-D was very simple, but their model is only applicable for colored images; the detection accuracy in grayscale images is very low.

It has been declared by established practices that an efficient splicing detection system should satisfy some specific criteria. The aim of this research is to come up with new splicing detection methods which enhance the detection accuracy of the spliced images with a low-dimensional feature extraction method.

1.2 Problem Statement

Image splicing is a common process that is used to produce digital image forgery (T-T Ng & Chang, 2004); it is achieved simply by cutting a region from one or more images and pasting it, or them, into another image (Dong et al., 2008). This technique can cause inconsistencies in many features, such as an abnormally sharp transient boundary at the

splicing edges (Fang et al., 2010), and these inconsistencies are used to detect the forgery. In recent years, several splicing detection methods have been proposed that are discussed in next chapter. However, these detection methods struggle with the following problems:

- **High-dimensionality**

  The existing methods concentrate only on the actual image splicing detection techniques, and handling the extracted high-dimensional and redundant features can be a difficult and time-consuming process. A large number of features exponentially increases the computational time. Humans and machine learning methods find it difficult to interpret high-dimensional data. Therefore, there is a need for low-dimensional image splicing detection approaches.

- **Low detection rate**

  Detection rate is one of the most important factors for splicing detection methods. The accuracy rate should be high enough to detect a spliced image accurately. In fact, if the detection rate is low, then determining the authenticity of the image will be problematic issue because original images will be detected as spliced by mistake, and vice versa. Thus, the accuracy rate of the detection method is significant.

- **High computation time**

  The feature extraction computation time of the authentic or spliced image should be reasonable for a system. In some cases, the features of more than 1000 images need to be extracted and classified. Accordingly, detection algorithms with a high time consumption are a fundamental systemic drawback.

- **Feature selection**

  Most of the features extracted from the images have redundancy. In the classification process, the redundant features reduce the accuracy of the classifier and thus the detection rate of the approach is decreased. Therefore, feature selection should be carried out to remove redundant information from the extracted features and to obtain the most discriminative information with less dimensionality. Consequently, the method applied as feature selector should be selected appropriately to provide a set of important features with a smaller size.

1.3 Research Questions

In order to overcome the challenges identified, there are some research questions that need to be considered:

1. How to develop new image splicing detection methods to detect spliced images more accurately?
2. How to develop new methods with low dimensionality?
3. How to apply SVD as a feature extraction method in image splicing detection?
4. How to preprocess features to improve the classification performance?
5. How to test and evaluate the performance of the splicing detection approach?

1.4 Research Aim and Objectives

This research aims to come out with two low-dimensional SVD-based image splicing detection methods which enhance accuracy rate within an acceptable computational time.

To meet the above aim, the following objectives should be achieved:

1. To investigate different image splicing detection methods.
2. To select an appropriate image preprocessing methods to prepare the image for feature extraction.
3. To propose two image splicing detection methods (SVD-Log and SVD-CN) based on Singular Value Decomposition (SVD) with low dimensionality.
4. To select appropriate feature selector to improve the classification performance.
5. To test and evaluate the two proposed methods (SVD-Log and SVD-CN) by measuring the detection rate using three standard image datasets DVMM v1 & v2, and CASIA, and evaluate them using the true positive (TP), true negative (TN), and accuracy rates which represent the average detection rate.

1.5 Scope of Work

This research consists of a number of stages:

i. **Research investigation:** Based on the results of the literature review, the limitations of image splicing detectors are identified. Information is collected from various publications including journals and conference papers. Finally, a decision is made as to propose two image splicing detection approaches with low dimensionality, and high detection rate in every available image dataset.

ii. **Methodology:** The second phase of the research consists of the design and implementation of the two splicing detection schemes. Its structure consists of two methods, which are SVD-based splicing detection methods.

iii. **Data collection:** The data for this research, namely three standard image datasets (DVMM v1 & 2, and CASIA), is prepared based on image splicing

operation. The selected image datasets are in standard use for evaluating image splicing detection techniques.

iv. **Experimental Results:** experimental results on the selected datasets are designed and achieved. The performance of the proposed methods is evaluated and compared with existing methods in terms of true negative, true positive, and average detection rates.

1.6 Research Contributions

By concentrating on the existing methods for image splicing detection techniques some issues were identified as follows, low detection rate, high dimensionality, and high computation time. However, by considering the concept of SVD and its applications in image processing, using SVD in image splicing detection technique will improve the mentioned issues promisingly.

This thesis presents the major research contributions. Several concepts are applied for image splicing detection throughout this research while most of them are general and can be applied in different fields, such as forgery detection and object recognition. The basic contributions of this research are briefly as follows:

- **Proposing two low dimensional methods for splicing detection:**

  The major contribution of this research is proposing two efficient SVD-based splicing detection techniques in which the second proposed method (SVD-CN) has less computation complexity than the first proposed method (SVD-Log). These techniques are able to distinguish authentic images from spliced ones with a high enough accuracy rate within a reasonable time. The research conducts comprehensive comparisons with the existing algorithms to ensure the high

accuracy of the results. The results demonstrate the proposed schemes have high enough accuracy rate in DVMM v1 & v2, and CASIA (98.78, 100, and 99.62 respectively) compared to other techniques.

- **Proposing two efficient SVD-based feature extraction techniques to extract useful features of the image:**

  Two new SVD-based feature extraction techniques (SVD-Log and SVD-CN) are proposed as image splicing detection approaches. SVD is a mathematical concept applied widely in image processing applications. This research presents how this concept can be utilized efficiently in feature extraction for detecting spliced images. Therefore, the main contribution of this research is how to define feature extraction methods based on SVD to detect spliced images accurately.

- **Selecting an appropriate feature selector to improve classification performance:**

  The extracted features might have redundancy of a linear or nonlinear nature which decreases classification performance. Thus, the features should be preprocessed before the classification phase. Selecting an appropriate feature selector that results in the best set of features with maximum classification performance is another major contribution of this research.

1.7 Thesis Outline

This research contains seven chapters. All of the chapters are developed based on suggested splicing detection schemes. The foundation of each chapter is explained as follows:

Chapter Two starts by defining state-of-the-art passive detection methods for image splicing. In this chapter, a comprehensive review of current schemes is provided. In addition, various aspects of the most recently developed algorithms for splicing detection, for example low dimensionality and detection rate, are examined.

Chapter Three then covers the methodology of the suggested image splicing detection schemes in each stage of the research. In this chapter the methodologies of various stages of the two suggested methods (SVD-Based Image Splicing Detection) are explained.

Chapter Four describes the detailed design of the proposed image splicing detection techniques. The implementation of each technique will also be elaborated in detail.

Chapter Five reveals the results of image splicing detection, highlighting the outcomes of the proposed algorithms. Also, this chapter analyzes and evaluates the results achieved by the methods in comparison to other splicing detection methods.

Finally, Chapter Six presents a summary of the thesis and the two proposed image splicing detection schemes. It explains the conclusion, as well as containing many technical suggestions for future study directions.

CHAPTER 2: LITERATURE REVIEW

2.1 Introduction

The current chapter explains the state-of-the-art techniques for splicing forgery detection according to statistical feature extraction approaches. During the past few decades, the ideas of digital image manipulation and forgery detection both attracted many companies which have attempted to appropriately counter and solve the problems mentioned in the previous chapter. The critical role of keeping the integrity and privacy of digital data has also had a large influence on different investigations.

In this chapter, many concepts regarding image forgery detection and current methods of splicing forgery detection will be examined. Various aspects of current algorithms for forgery detection including detection rate, complexity and dimensionality are studied in the current chapter, which contains the following parts: Section 2.2 explains digital image formation and its different domains (spatial and frequency). In section 2.3, different digital image concepts, digital forensics, and the detection and creation of digital image forgery, which are utilized for the aim of splicing forgery detection, are all reviewed. In addition, a number of recently proposed methods for splicing forgery detection are presented in Tables 2.1-2.4.

2.2 Digital Image Formation

Generally, an image can be explained as a function $g(x, y)$ with two dimensions i.e. x and y, in which x and y are both spatial coordinates and the value of $g(x, y)$ in any x and y location indicates the grey level intensity of that image in that point. For instance, a greyscale image could be defined as (Cao, 2006):

$$g_{ij}\ where\ g_{ij} = g(x_i, y_i) \quad \textbf{(2.1)}$$

When *x* and *y* and *g(x, y)* are discrete and finite quantities, this image is known as "a digital image". In addition, the finite set regarding the digital values are known as pixels or picture elements

(Cao, 2006).

Every pixel value in the digital image is related to the brightness of that point in that image: usually its value is obtained from analog to digital converter output. The image, as a matrix of pixels, (if it is square) is defined as N * N m-bit pixels, in which N is the amount of points and the axes, as well as m, which controls the brightness values. Utilizing the m bits provides a range of 2m values between 0 and 2m – 1 (Nixon, 2008).

Here, if m equals 8, it provides levels of brightness from 0 to 255, which are mainly shown as black and white respectively, along with shades of grey between them, since they are relevant to the greyscale image presented in Figure 2.1 (Thompson & Shure, 1995).

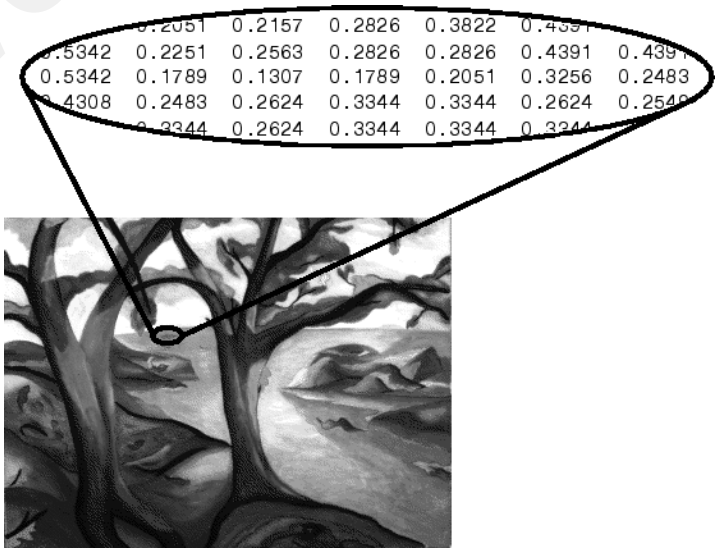


Figure 2.1: Sample of Gray-scale image

In general, all of the pixels are maintained in the memory of a computer as an array with two dimensions or a real number matrix. A group of single 2-D images shapes the colored images. Various techniques of image processing regarding the monochrome images could be developed as color images (3-D) by processing the individual three component images (Gonzalez, 2010).

In the case of color images, rather than utilizing only one single image plane, the color images are shown through three key intensity components. In general, such components are red, green and blue (RGB Model) even though there are other color models. For instance, the CMYK color model is described with components including black, yellow, cyan and magenta. In the case of different color models, the color of the pixels could be defined by two key methods.

Firstly, there is ability to relate each pixel with an integer value, which could be applied as the index for a table which stores the intensity of different color components. The index is utilized in order to repair the actual color from this table while pixels are being processed or displayed. In this regard, the table is considered as the palette for the image, and such display can be done by color mapping. The key reason for utilizing such color representation is to minimize memory requirements. This means that one image plane, as well as a palette, are stored. This uses less memory than storing blue, green and red components specifically, and reduces the cost of the hardware, as well as having other benefits, such as in image transmission. The key disadvantage is the fact that the image quality will be minimized due to a narrowed set of available colors. Figure 2.2 shows a sample of an indexed digital image (Thompson & Shure, 1995).

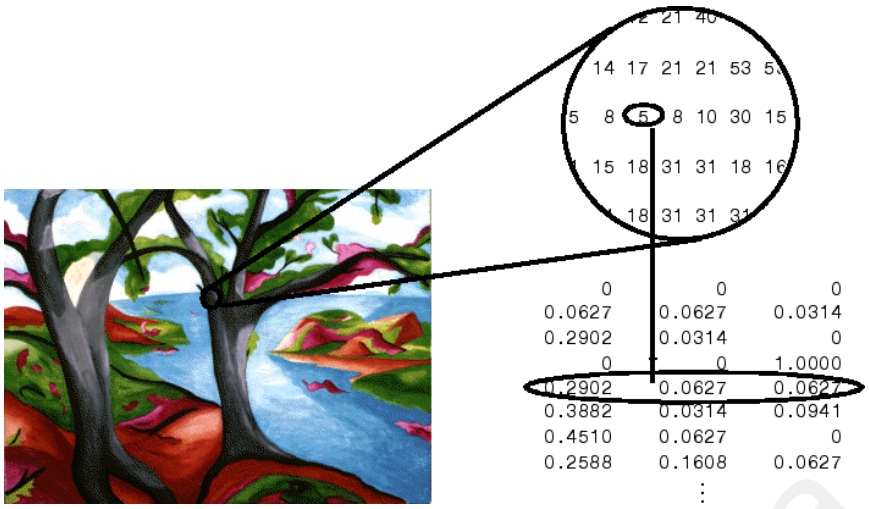


Figure 2.2: Sample of Indexed image

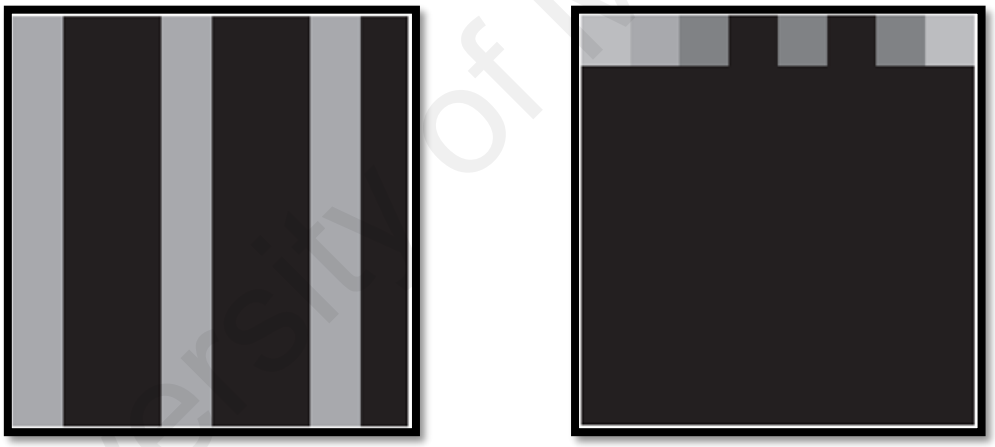
Another way to show the color is to utilize many image planes to keep the pixel color components. Such a scheme is considered as true color, and it shows the image more appropriately, specifically by using more colors. One of the most popular formats employs 8 bits for each of these three components of RGB. Such images are called 24-bit true color, and they can include 16777216 various colors at the same time. Regardless of the specific need for more memory, the image quality also the continued reduction in memory cost makes the format a good choice, even for selected still frames from a video recording. In practice, a suitable compression algorithm is often useful in such situations, specifically when images need to be transmitted within a network (Nixon, 2008). Figure 2.3 presents an example of a true color image (Thompson & Shure, 1995).

Figure 2.3: Sample of true color image

Images can also be considered in a representation other than the spatial domain, such as the frequency domain. According to its name, images are assumed to be frequency collection constructs. This can be performed in cases of the frequency domain, and additionally assumes various processes of transformation. It will provide various perspectives toward both image processing and the image itself which will be utilized in various kinds of applications, not just as a tool to extend methods but also to allow quick computer processing. Different transformations, including Discrete Wavelet Transform (DWT), Discrete Cosine Transform (DCT) and Discrete Fourier Transform (DFT), are broadly used in image processing applications. A summarized description of these transformation approaches is provided here.

2.2.1 Discrete Fourier Transform (DFT)

The basic and first transform would be DFT – Discrete Fourier Transform. The images' Fourier transform can also be achieved through optically transmitting a laser via photographic forming and the slides of an image by means of a lens (Nixon, 2008). However, it is limited to transmissive systems while the reflective formation can potentially broaden its application remarkably (due to the fact that optical calculation is only a little quicker than its digital counterparts). The 2-D DFT magnitude to a specific image for vertical bars is demonstrated in Fig 2.4 (b) (Nixon, 2008). It suggests that there are only spatial horizontal frequencies; the image will be consistent within the vertical axis, so there will be no spatial vertical frequencies.



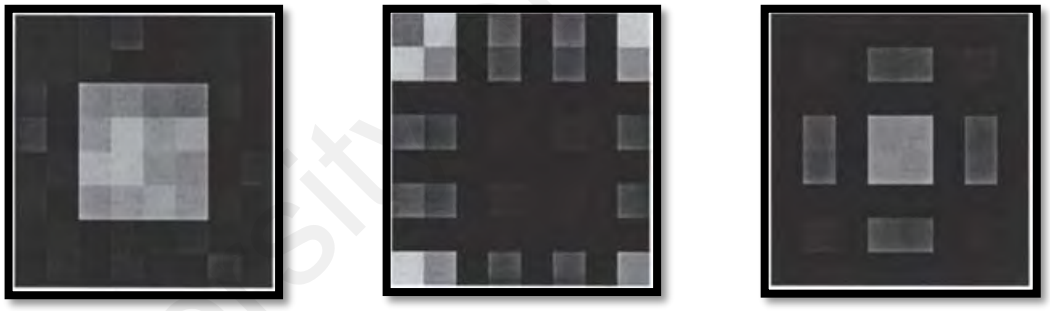
(a) Image of vertical bars (b) Fourier transform of bars

Figure 2.4: Applying the 2D discrete Fourier transform

The image's Fourier transform provides frequency constructs. Each single component's position demonstrates its own frequency: components with low frequency are close to the origin, while components with high frequency are further away. The component with the lowest frequency - zero – is a D.C. component which shows the sample average value.

The 2-D Fourier transform arrangement puts components with low frequency at the corner side of the transforms. The square image in Fig 2.5(a) demonstrates this within its transform in Fig 2.5(b) (Nixon, 2008). In fact, a spatial transform is more convenient to be visualized with D.C or zero frequency components in the center, along with increasing frequency on the image edge. This can be adjusted either through rotation of each of the four existing quadrants in Fourier transform by 180 degrees.

The other option is to reorder the original image to provide a transform that moves the transform toward the center. These operations both lead to an image as presented in Figure 2.5(c), in which the transformation is easier than what it seems. Notice that it is supposed to improve visualization, and will not modify any information in the frequency domain, just the way it is demonstrated.



(a) Image of square (b) Original DFT (c) Reordered DFT

Figure 2.5: Rearranging the 2D DFT for display purposes

In order to rearrange this image and put the D.C component in the center, the frequency components must be recorded. These can be obtained by multiplying each single image point's $g(x, y)$ by $-1(x + y)$.

Another DFT property is called a shift invariant. Decomposition as spatial frequency is not dependent on the features of the positions inside an image. If all of the features shift by a certain value, or obtain the image from another position, the magnitude of the image's Fourier transform will not change. Such a property is considered as shift invariance. However, although the Fourier transform magnitude will remain fixed, the phase will not remain constant.

The differed phase shows that in case of application, the Fourier transform magnitude of a face could be similar regardless of the face position in images (subject or camera could move down or up), considering that the face is bigger than the version in the image. It demonstrates that while Fourier transform is utilized to examine the human face image, to define it through its spatial frequency, we should not control the camera position or the face exactly.

The image's Fourier transform will rotate while image source is rotating. This can be considered as due to decomposition as spatial frequency demonstrates the feature orientation inside an image. Thus, an orientation dependency is constructed towards the process of Fourier transform. It reveals that if the properties of frequency domain should be utilized in image analysis, through the Fourier transform, the original image's orientation should be fixed or known. Usually it is possible to fix the orientation as well as estimating its value, while the orientation of a feature cannot be fixed. In addition, there are many methods to bring invariance in rotation, for example translation to some polar representation, so the process can be complicated.

The superposition principle is highly critical in system analysis. Remarkably, there is a property which means that a system is linear if its reaction to two joined signals is the total of responses to individual signals. In practice, it shows that the images can be divided by considering their components in the frequency domain. For example, in an image of

bloody fingerprints on cloth, it is not easy to separate the cloth from the fingerprints through analyzing the image. However, through transmitting it into the frequency domain, its Fourier transform demonstrates powerful components because of their texture and shows the inverse Fourier transform, meaning that the cloth can be excluded from the original picture. The fingerprint now can be observed in the final image.

Since DFT is widely applied in facial detection methods (Lai et al., 2001; J. Li et al., 2004), it is also applied in this research to investigate its effects in image splicing detection applications.

2.2.2 Discrete Cosine Transform (DCT)

DCT is the most popular block-based transform. It decorrelates the image data and then each transformed coefficient can be freely encoded without sacrificing compression efficiency. The two-dimensional DCT that is used in image processing is as follows (Khayam, 2003):

$$S(u,v) = \frac{2}{N} C(u)C(v) \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} s(x,y) \cos\left(\frac{\pi u(2x+1)}{2N}\right) \cos\left(\frac{\pi v(2y+1)}{2N}\right) \quad \textbf{(2.2)}$$

$$f(x,y) = \frac{2}{N} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} C(u)C(v)S(u,v) \cos\left(\frac{\pi u(2x+1)}{2N}\right) \cos\left(\frac{\pi v(2y+1)}{2N}\right) \quad \textbf{(2.3)}$$

$$C(k) = \begin{cases} \frac{1}{\sqrt{2}} & for\ k = 0 \\ 1 & otherwise \end{cases}$$

Where *u, v, x, y = 0, 1, 2, ..., N-1*. Here are some of the most important properties of DCT:

2.2.2.1 Decorrelation

The main objective of DCT is that it eliminates the correlation between adjacent pixels. In fact, DCT is a kind of transformation that maps the correlated data called spatial data into uncorrelated coefficients referred as transformation. This makes the resulting coefficients unrelated, which can be encoded separately. Figure 2.6 shows that the amplitude of the autocorrelation for the transformed image is very small, which exposes the decorrelation characteristics (Khayam, 2003).

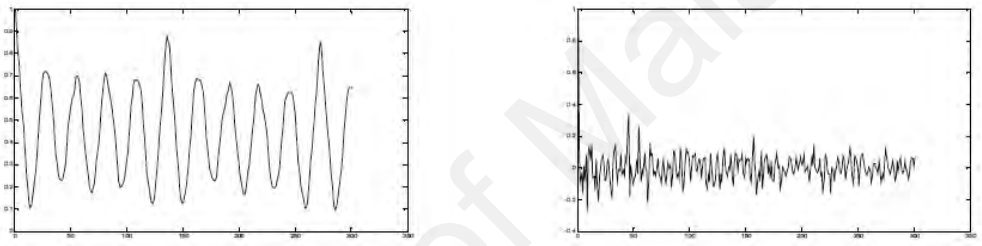


Figure 2.6: Normalized autocorrelation of an image before and after DCT

2.2.2.2 Separability

The DCT Equation (2.3) can be also expressed as follows:

$$S(u,v) = \frac{2}{N} C(u)C(v) \sum_{u=0}^{N-1} \cos\left(\frac{\pi u(2x+1)}{2N}\right) \sum_{v=0}^{N-1} s(x,y) \cos\left(\frac{\pi v(2y+1)}{2N}\right) \quad \textbf{(2.4)}$$

This gives an advantage whereby *S(u, v)* can be calculated in two steps by using 1-D operation on the rows and columns of an image. This property also exists in inverse DCT. Figure 2.7 shows this property (Khayam, 2003):

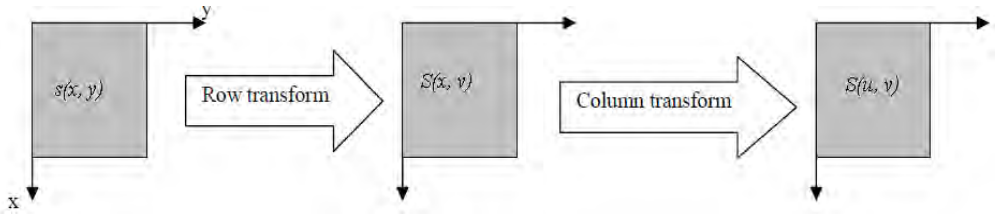


Figure 2.7: 2-D DCT computation using separability property

2.2.2.3 Symmetry

Symmetry is another property of DCT which is expressed as (Khayam, 2003):

$$T = M\ s\ M \tag{2.5}$$

Where M is a N × N matrix and *m(i, j)* is computed as (Khayam, 2003):

$$m(i,j) = \alpha(j)\sum_{j=0}^{N-1}\cos\left(\frac{\pi i(2j+1)}{2N}\right) \tag{2.6}$$

and *s* is the N × N image matrix.

This is a practical property due to inferring that the transformation matrix can be pre-calculated offline and then applied to the image to provide enhancements in computation efficiency in terms of orders of magnitude (Khayam, 2003). DCT has been widely applied in different forensics techniques including steganalysis, copy-move, splicing, etc. (He et al., 2012; Shi et al., 2007b; J. Zhang et al., 2009). It is also applied in this work to evaluate the performance of SVD-based features on DCT transformed images.

It is also applied in this work to evaluate the performance of SVD-based features on DCT transformed images.

2.2.3 Discrete Wavelet Transform (DWT)

Discrete Wavelet Transform is another method which decomposes the input signal into four non-overlapping multi-resolution sub-bands LL1, LH1, HL1 and HH1 (Al-haj & Abu-errub, 2008). LL1 is the low frequency sub-band which its statistical characteristic is similar to the original image and HL1, LH1, and HH1 are the high frequency sub-bands. LL1 can be more further decomposed into another four sub-bands, because it has the most similarity with the original image (Al-haj & Abu-errub, 2008; Mitra & Acharya, 2005).

The total information of the image (energy of the image) is presented in the low frequency sub-band. The high frequency sub-bands consists of the edges and texture of the image, meaning that the human eye remains insensitive to the changes in these sub-bands. Figure 2.8 shows the two-level decomposition of image according to discrete wavelet transform (Al-haj & Abu-errub, 2008).

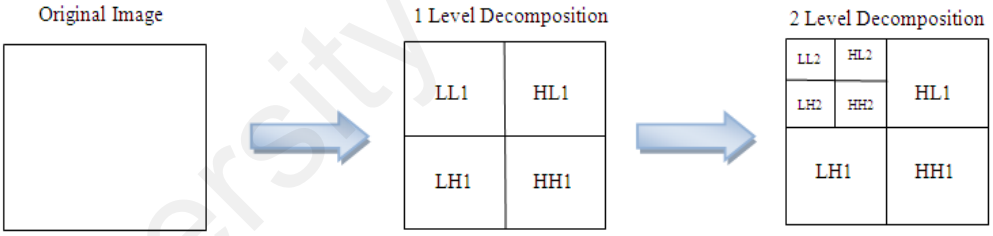


Figure 2.8: Two level multi-resolution wavelet decomposition of an image

Broadly, DWT is utilized in field of image processing due to its great benefits in space-frequency and multi-resolution analysis, and its ability to show localized and transient modifications in frequency or spatial domains. Thus DWT is a proper method to be used in different fields of image forensics.

From Gabor wavelets applications, measurement of the iris texture can be realized to provide a high strength security system (Daugman, 1993), the extraction of face feature regarding automatic face recognition (Lades et al., 1993) and also the recognition of

image forgery. The wavelets are developing promisingly (Daubechies, 1990) and also have applications in analysis, coding, image restoration and image texture (Laine & Fan, 1993) (Silva & Ghanbari, 1996) (Banham & Katsaggelos, 1996).

In general, DIP, or digital image processing, means to process a single digital image through using a digital computer using algorithms for specific transformations. Due to digital image data, which is in the form of a matrix, DIP can use many mathematical methods. The critical areas include integral transforms, computational linear algebra, statistics, as well as other numerical analysis methods. A lot of DIP algorithms could be written as matrix equations, thus, computational techniques in linear algebra can be a critical dimension of this subject (Jähne, 2005).

The process of digital image includes a broad range of applications, for example image compression and operation, image analysis and computer vision. In fact, there is an assumption of three kinds of digital processing: first, low-level processing, which is detectable by the fact that both its outputs and inputs are images. Second is the mid-level processing, which is determined by the idea that the inputs are images, but its outputs are extracted attributes from such images; and finally higher-level processes, including "to make sense" of the ensemble of identified objects as in the case of image analysis, as well as conducting the cognitive function which is related to human vision (Jähne, 2005).

In fact, the processing of digital images is a practical technology in several fields such as: Classification, Image compression, Pattern recognition, Feature extraction, Multi-scale signal analysis and projection (Cao, 2006).

In this research we apply the digital images in grey and true color to perform feature extract and image analysis based on singular value decomposition concepts.

2.3 Forensics

The technology of communication and information has developed a digital revolution that is fundamentally altering the world. The digital information stored in computing systems describes tangible assets of our lives in detail, and continues to become an even bigger part of reality. In addition, existing counterparts within the field of communications, since they are computer-mediated, are changing a lot of real-world or social interactions. As a result, the processes of law must develop and evolve in the digital sphere, for example in terms of prosecution and enforcement of crimes.

This will increase the necessity to reconstruct, in a reliable and scientific way, sequences of actions conducted within the digital sphere to identify (or at least approach), facts regarding causal relationships. It should be known that perpetrators can be potentially held accountable for actions, in order to deter any kind of imitators. Those efforts to utilize scientific approaches to achieve probative facts in criminal studies are known as forensics. This concept is rooted in the Latin word "Forum" which means "the main square" or a place in which public court hearings occurred during ancient times.

The term "computer forensics" has been used to define the same sort of efforts when computers are involved in criminal activities (Kruse II & Heiser, 2001). Somehow, the definition of computer forensics is blurred somehow, since computers are able to represent manifold relations with regard to crimes: they can be used as devices to commit crimes within the real world, or tools that generate a digital sphere through which crimes are committed. In both situations, forensic scholars can strive to reveal probative facts from the computers involved.

Such situations can be even more sophisticated when sensors are provided to different scenarios. The sensors are able to capture parts of reality and transform them into digital

representations, for example, audio files or images that are later stored and/or processed in many computers. These kinds of digital representations regarding parts of the real world could be interesting for forensic scholars; however, they can only be used as probative facts when they are both authentic and reliable. Understanding such objectives explains the scope of multimedia forensics. According to Figure 2.9, it can be observed how all of the forensics sciences can be subdivided via their evidence domain. By this domain, we can find the facts: analog or classical forensics will identify physical evidence traces while digital forensics are limited to showing digital evidence. In this part, all of these concepts are defined briefly (Böhme et al., 2009).

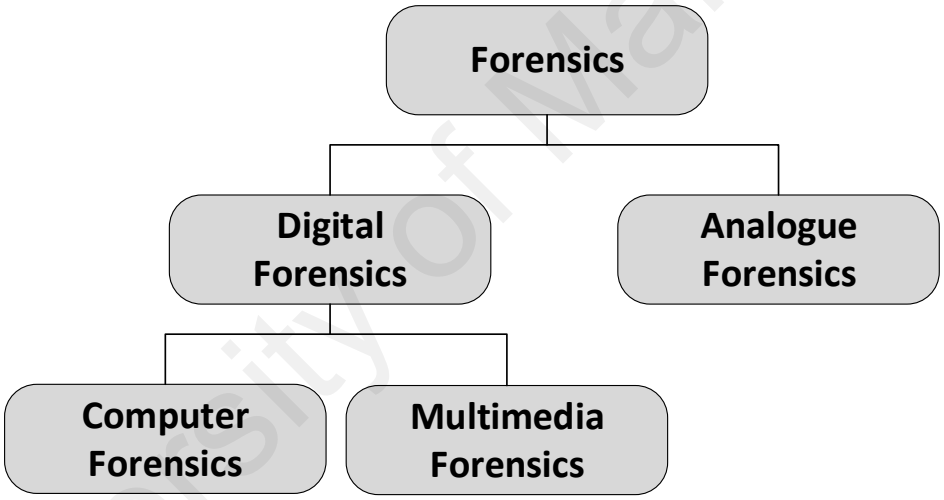


Figure 2.9: Domains of Forensics Science

2.3.1 Analog Forensics

Classical forensics means the effort to demonstrate probative facts from physical proof in real context. It was discussed that the discipline is based on two key principles: 1) divisibility and 2) transfer (Inman & Rudin, 2002). The first one refers to the fact that matter can be divided into smaller parts when the proper force is used. Such smaller parts

will maintain the features of original matter, and also obtain the features provided by this separation.

The other principle, which is also called the exchange principle, mentions that while two subjects are interacting in reality, each of them will pick up some physical matter from the other one (Inman & Rudin, 2000; Saferstein, 2001). This kind of exchange might include footprints and fingerprints, clothes fibers, hair, wounds, scratches and oil stains. This example reveals that the transfer should not just be minimized to a transfer on some microscopic scale. According to Inman and Rudin (Inman & Rudin, 2002), the transfer even covers the exchange of patterns, such as footprints. Thus, transfer is not just transferring to matter but transferring its traits as well. If a person accepts the given principles, so it is, according to Kirk's statement (Kirk, 1974):

"In fact, physical evidence could not be wrong, it could not be absent wholly or it could not perjure. Its value can only be diminished by human failure in order to realize it, study and recognize it."

This means that a scholar of unconstrained forensics can freely investigate reality from many other infinite perspectives (though not simultaneously). Hence, they will have no possible chance to even identify the subtlest trace. However, in case of modern epistemology, it is acceptable to agree that human understanding of reality is limited in many different ways. In addition, the organs of human sense and perception provide an inaccurate image of reality. At the time that it might be assumed as a filter that could be compensated for at some point through technical tools (a microscope improves the human visual system resolution), the uncertainty principle presented by Heisenberg provides more barriers: an observer is usually a part of similar reality, and the fact that they are interacting with it will alter the object which should be observed.

Also, it is consistent with reflections of Inman and Rudin on the transfer and division of physical matter (Inman & Rudin, 2000) which do not differentiate between forensic scholars and perpetrators as the subjects who are participating in the exchange process. The important fact about such an insight is that it is reliable for probative facts initiating from physical evidence. In addition, if it is difficult for a perpetrator to understand all of these traces, is there a possibility for them to forge other traces which can result in false accusations? It will correspond to their efforts to alter reality in order to generate a very different image of their actions.

Due to both perpetrators and forensic investigators being involved in a similar reality and thus subject to the same cognitive and physical barriers, even the most highly sophisticated perpetrator could not make sure that their "alteration of reality" is totally consistent with reality. Thus, in reality committing an "excellent crime" and creating a consistent image of reality, which can hide all of the traces, is a challenging issue. Hence, the exact study of physical proofs can be used to find either reliable probative evidence or zero (Böhme et al., 2009).

2.3.2 Computer Forensics

Also computers are more than physical tools, they also shape part of the real world. Thus, at first glance, if a person accepts the transfer and divisibility principles, they need to hold equally for computer forensics. However, while individuals talk about computer forensics, they usually have the implicit consideration that such forensic analysis is only limited to digital evidence maintained in finite automata represented by each computer. This suggests an observer model, together with a dramatically minimized view on reality: only bits are the theoretical constructs, which include no information regarding their history. For instance, the usual practice in order to copy digital evidence available in a

computer and to base more studies on this copy exclusively implements such a model for the observer (Casey et al., 2014; Group, 2006). This observer model additionally shows that, in case of computer forensics, the question of divisibility is not important. The traits' transfer will be the possible basis for computer forensics theory.

This comes together with the consequences of the reliability of facts initiating from digital evidence. Due to the fact that the amount of states within a closed system is limited, there is often a non-negligible possibility that a complex perpetrator puts a computer in a state that suitably removes all of the traces. Considering that the whole states for a computer are recorded on hard disk, these states could be obtained, for instance by utilizing the computer after being booted from live-CD and so not modifying the contents of a disk.

It will contribute to develop a plausible and valid state with logical effort and time, as just a minor fraction of all probable states is in fact relevant for identifying a certain system state. But this provides an observer model which only considered parts of the whole state space; so the observer will ignore additional used technology to generate a certain state. Practically, it is usually not easy for researchers to show the systems' borders to be examined, specifically if it utilizes network wireless links.

Also with an observer model, which shows the whole system, it follows from the analysis limitations of digital evidence, which can never neglect the probability that a perpetrator has dealt with all of the digital traces appropriately. While these kinds of sophisticated perpetrators might be rare, some skepticism is due in case of the residual probability of error while digital proof is utilized in court in order to judge things like capital crimes.

So, is the transfer principle suitable for computer forensics? A lot of practitioners these days do not agree, since based on their knowledge, each perpetrator might make mistakes and then will leave traces of criminal activity on the evidence. Nevertheless, the nature of evidence (i.e. digital) will make it possible to include traces appropriately. In addition, unlike the practical barriers of the observer in analog forensics, the perpetrator is aware of the "blind spot" of the researcher beforehand, so can modify his actions and create misleading or false facts. There are benefits of cheap and easy computer forensics due to automation, meaning many studies can be carried out from offices of forensic scholars, which leads to lower costs of probative force. Since social relationships are shifting toward the digital sphere, the offices of state-funded studies should make precise decisions on resource allocation among the exploitation of digital and physical evidence.

A very different condition comes if computer forensics is realized in a wider sense which compromises digital and physical evidence. This kind of physical evidence, even though it is often expensive to achieve, could be highly indicative of side-information. Some attributes, for example wear and tear, electromagnetic emanations recording (Kuhn, 2002), temperatures (Zander & Murdoch, 2008), and also all types of analog directions on storage tools (Wright & Kleiman, 2008) may provide some information regarding past states of the target computer, and therefore of effort made to conceal traces. Also, the digitized or digital evidence maintained in other tools could shape this additional information when there is a secured integrity, such as through secure logging (Schneier & Kelsey, 1998). For instance, United States agent Oliver L. North was convicted in 1986 over the Iran-Contra affair because he failed to deal with evidence kept on backup tapes (Böhme et al., 2009).

2.3.3 Multimedia Forensics

A critical group of digital data that is usually identified and studied on storage devices is called digital multimedia data, including image, audio and video. At the same time that digitalized and digital media currently influence and empower our daily lives in remarkable ways, some critics have revealed concerns that it is not easy to alter media data. Thus, the questions on the authenticity of media are of high relevance and of specific interest in court that consequential decisions can be made based on digital media evidence.

During recent years, the scope of multimedia forensics has grown significantly and currently includes scholars from various communities (Böhme et al., 2009; Popejoy, 2015).

Multimedia forensics could be explained as science which attempts, through studying digital assets, to conduct an evaluation of these contents and to also obtain information which can be helpful in supporting and mentioning the material connected to a scene provided by a certain digital document. In fact, multimedia forensics should provide appropriate tools to cope with disparate digital tools which will provide images and also with various processing devices, which lets the unskilled user modify the digital products (Caldelli et al., 2009). The methods of multimedia forensics, nowadays, highly emphasize the study of digital images.

2.4 Digital Image Forgery Creation

Within decades of photography's emergence, different approaches had been suggested to alter the images. For example, combination print was among the first methods of image forgery, in which darkroom skills were utilized to print many image fragments on a

unique photograph paper. One of the first famous combination prints was named Oscar G., as well as *The Two Ways of Life* (1857) by Reijlander, which employed almost 30 images. Then, during the 20th Century, photomontage which is a cut and paste composite of image fragments, became popular in political satire, surreal art and other forms. Photomontage and combination print both are time-consuming and technically demanding, and their usage is usually detectable.

Due to high availability of strong editing tools, for example Adobe Photoshop, several image modification functions can be done in the digital domain with a more convenient process, leading to higher verisimilitude. Generally, the process of image forgery creation includes transformation, selection and composition, as well as final image retouching. The creators of the forgery can thus fuse the transformed fragment of image or generated image portion from transformed 3D model toward other image utilizing methods, for example matting for coherent-looking composition. Lastly, a composite image is retouched to omit the remaining artefact. This phase might include the removal of specific items from an image, which is sometimes considered as reverse cropping (Tian-Tsong Ng et al., 2006).

2.5 Digital Image Forgery Detection

Traditionally, the photographer shows the truth. However, the same faith in digital photos has faded, because of their ease of manipulation. Images, as compared to texts, offer natural and influential communication media for people, since often people require no specific training to create an image. Hence, the ability to confirm the digital images' credibility and conduct image forgery detection could support digital image trustworthiness. Currently, digital images already have been employed for reporting the news, insurance claims, criminal or forensic studies, national intelligence investigations

and legal proceedings. Thus, this kind of image forgery detection can highly influence the application domain mentioned above.

The key performance of image forgery detection is to evaluate both the origin and authenticity of an image. Thus, the trustworthiness of the digital images is a core concern for the process of image forgery detection. In 1993, the camera concept of authenticity (Friedman, 1993) was suggested in order to increase accountability of digital images. In fact, a trustworthy camera puts a digital watermark on images when it is acquired, and later image tampering can be traced according to modifications on the digital watermark. However, the identification of a trustworthy camera needs the camera producers to follow a popular standard protocol, when the customers should accept reduced quality of images because of that embedded digital watermark.

The most common concern about digital watermark security was shown in SDMI, Secure Digital Music Initiative fiasco (Craver et al., 2001), in which the suggested audio system of watermarking was hacked swiftly by scholars of watermarking and cryptography of Princeton University, Rice University and Xerox PARC. In order to detect forgery images, a lot of approaches were introduced. These techniques are widely classified as two groups: Passive and Active techniques. This categorization is according to the idea that is the original image existed or not. Figure 2.10 shows classification of image forgery detection techniques (Tian-Tsong Ng et al., 2006).

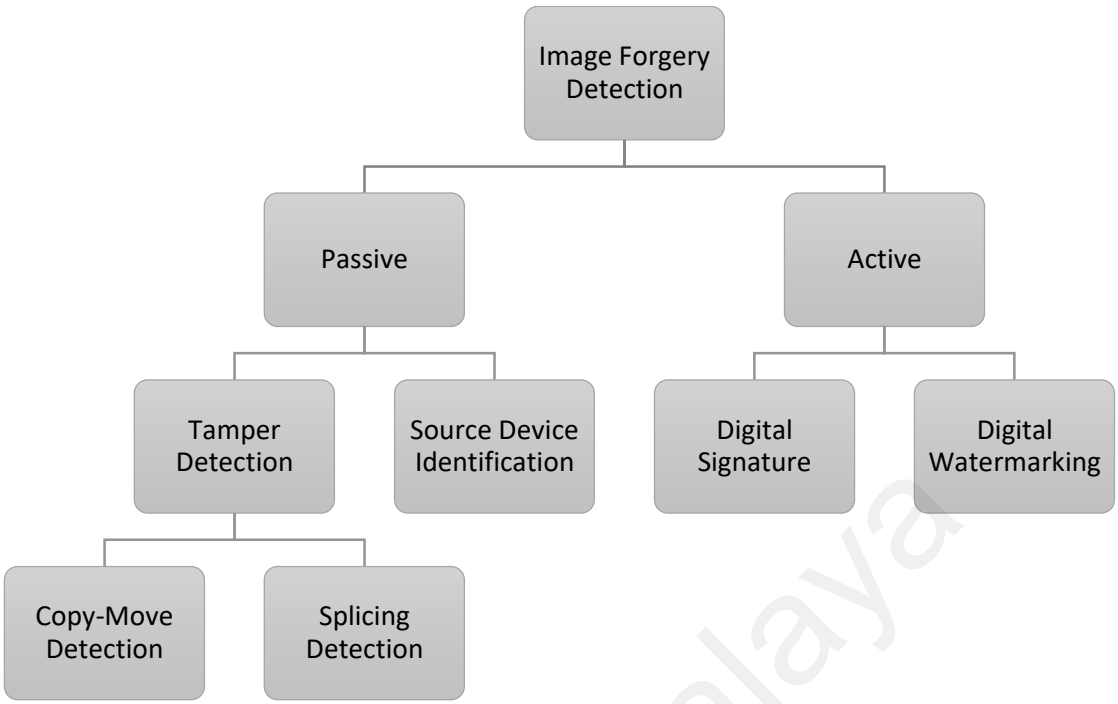


Figure 2.10: Classification of image forgery detection technique

2.5.1 Active Methods

In methods of active authentication, prior knowledge of an image is in fact essential to the authentication process. It is related to data, in which many codes exist in an image at the generation time. Such code verification will authenticate the images' originality. The techniques of active authentication are also categorized into two kinds: digital signatures and digital watermarking (Cox et al., 2002; Katzenbeisser & Petitcolas, 2000; Z. Zhang, Ren, et al., 2008). The digital watermarks are embedded in images at the time of image acquisition, or in the processing phase (Mushtaq & Mir, 2014).

There are two key sides for designing a typical system of watermarking (Hartung & Kutter, 1999). The source side is to prepare the watermark data *D* to be embedded into the original image *M*, in order to achieve the watermarked image *N*. Another side is watermark *D* extraction and provision of the confidence assessment for the image. Here,

Figure 2.11 demonstrates the generic watermark system at the source side. There is a watermarked image $N = f_1(M, D, K)$, in which K will denote the key (Luo et al., 2007).

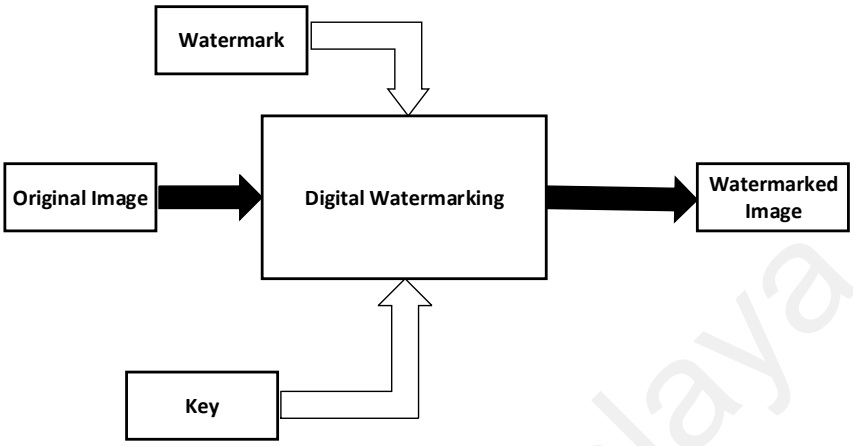


Figure 2.11: Generic Watermark insertion

Figure 2.12 demonstrates the extraction of watermark at the receiver side. The extracted watermark could be denoted as $D' = f_2(N, K)$, in which N is known as the authentic image (Luo et al., 2007).

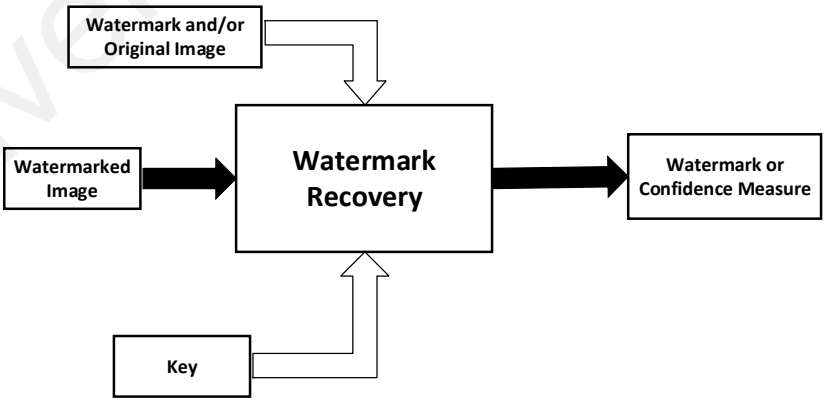


Figure 2.12: Watermark Extraction

As noted before, the basic watermark-based technique for authentication of images is adding a watermark to original images in the source side as well as recovering partly or fully the watermark at the receiving side in order to understand if the image was modified. Thus, any kind of manipulation before embedding the watermark could not be detected by means of such a technique (Birajdar & Mankar, 2013).

The method of digital signature has the same features and characteristics, in that it includes secondary information, mostly drawn from the image at the acquisition step, into the image. Many studies have been conducted in digital signatures (Lin & Chang, 1998; C.-S. Lu & Liao, 2003; Sengupta & Mandal, 2013; X. Wang et al., 2012; H.-B. Zhang et al., 2004) as well as digital watermarking (Chamlawi et al., 2010; Y.-S. Chen & Wang, 2011; Rosales-Roldan et al., 2013; Shieh et al., 2006; Spagnolo & De Santis, 2011). The key drawback of such methods asserts that they should embed into images at the recording time utilizing specific equipment, so prior knowledge regarding the image remains indispensable (Mushtaq & Mir, 2014).

2.5.2 Passive Methods

Also the passive authentication known as image forensics is the authentication process of images with no need for prior information from the image (Tian-Tsong Ng et al., 2006; Zhou & Zhang, 2010). The passive methods are based on the idea that, although tampering might not have visual traces, it will possibly modify underlying statistics (Mushtaq & Mir, 2014). It is in fact such inconsistencies which are employed to detect tampering images. Moreover, exhaustive studies have been done in the passive image forensics field (Farid, 2009; Luo et al., 2007; Tian-Tsong Ng et al., 2006).

There are two questions in case of passive authentication (Caldelli et al., 2009):

a. What is the digital content source? (Authentication of source device)

b. Is this kind of digital content authentic? (Tamper detection)

2.5.2.1 Source device authentication

A digital image might be captured from different devices such as scanners, cameras, technology, and computer graphics. The concern from this issue is toward source identification (Birajdar & Mankar, 2013).

In a law court, the origin of a specific image could demonstrate key evidence; the evidence's validity may be based on logical doubt that this image was captured from the device that it was supposed/claimed to be captured with, such as covert videos or video-surveillance materials (Redi et al., 2011).

Thus, the process within the devices is almost known. However, various imaging devices have various features due to their physical apparatus, various image processing techniques, as well as various parameters used in those devices. This will result in various patterns from the output images. Such patterns can be employed as "fingerprints" of those devices to trace their image sources.

Figure 2.13 demonstrates the general identification pattern of image-source (Luo et al., 2007). Consider that *M* is the image to be detected. It might be captured from many candidate imaging devices. The process of identification includes: first, the *M* features and schemes from image devices are mainly being extracted by applying knowledge of the model for image acquisition. Later, the existed similarities among such patterns and the features are evaluated. Finally, a confidence measure for the image devices will be given in order to recognize the source of image *M* (Luo et al., 2007).

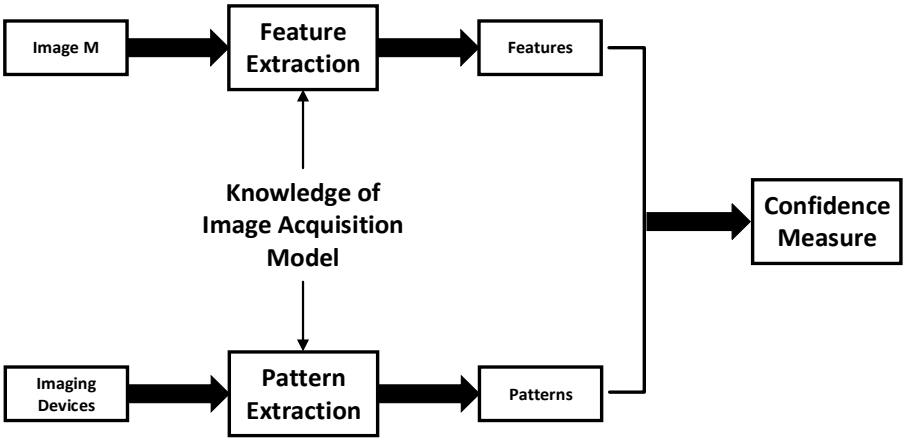


Figure 2.13: General Identification Pattern

2.5.2.2 Tamper Detection

Referring to the *Oxford Dictionary* (Dictionary, 2004), literally, the verb tamper means "to interfere with something for causing damage or do unauthorized modifications". In case of digital image processing, the tampering shows the intentional modification of images because of malicious objectives: since images are natural information carriers, image modification is known as tampering while it specifically attempts to alter semantic meaning of those visual messages (Redi et al., 2011).

A digital image might be modified by image processing after it is captured by a certain imaging device. The subject considers the issue if it is possible to define an image that is modified by a specific operation or not. In fact, original images usually include many consistent features, for example light condition, consistent noise distribution, etc. However, these features have to be modified by applying some post-processing operations on the image. In addition, some characteristics of the modified images will become more or less inconsistent at some point. Realizing the differences before and after the modification operations is a critical factor to distinguish the authentic images. It is

mentioned that many operations include malicious tampering while others do not alter the image contents, such as contrast or color adjustment.

Figure 2.14 demonstrates a general detection scheme of image modification (Luo et al., 2007). The image *M* is tested to see if it is original or has been modified through specific operations. According to Figure 2.14, first the features are extracted from image *M* and achieve original or modified patterns, significantly utilizing the information of the model of image manipulation or, on some occasions, joining with statistical features in natural scenes and the model for image acquisition. Later, patterns and features are compared to decide if image *M* was modified or not.

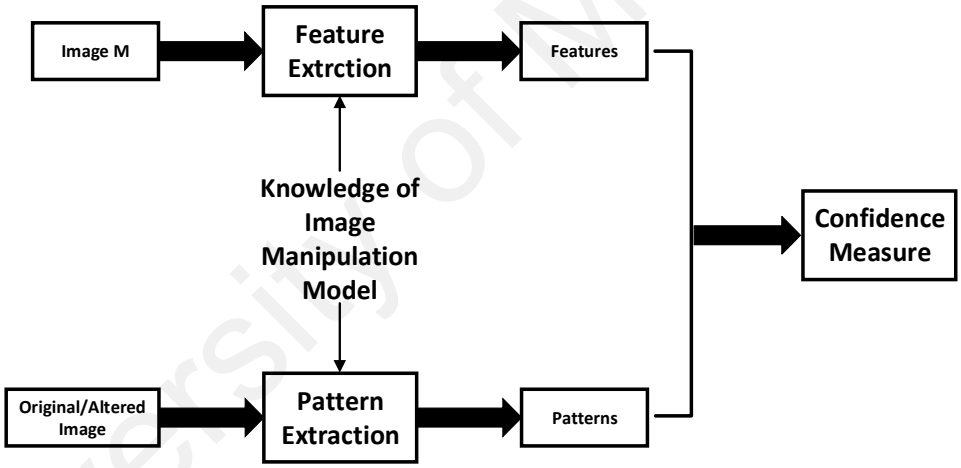
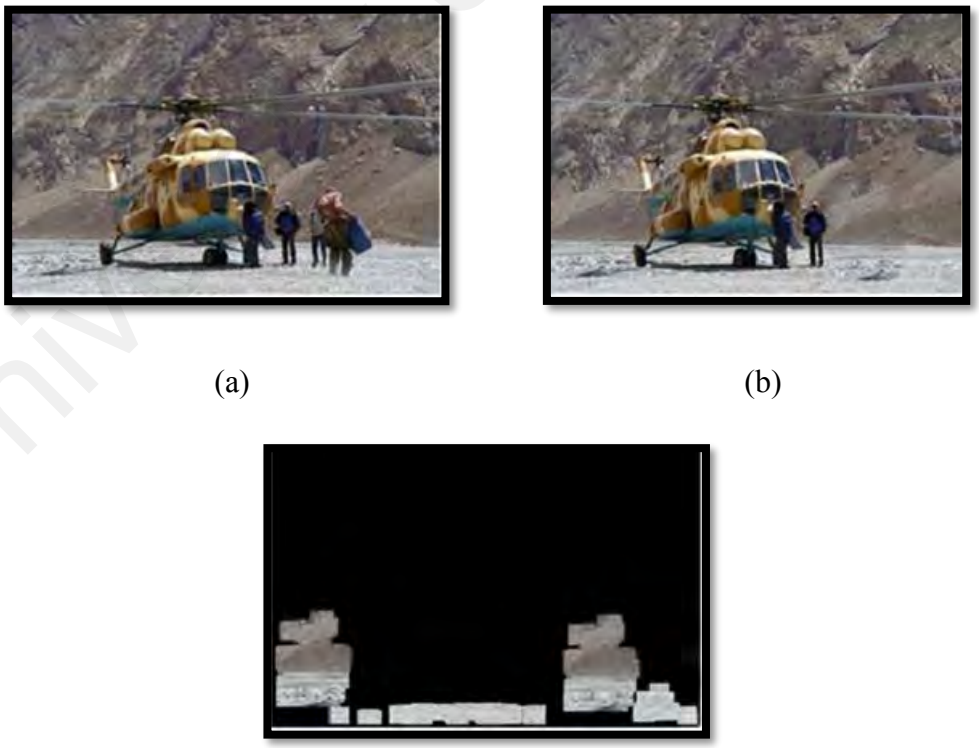


Figure 2.14: General Image Detection Schemes

On most occasions, passive forensics could be converted as a pattern identification problem. An appropriate solution to this problem is realizing various patterns based on knowledge from different imaging devices, or alteration, or natural scene barriers. The patterns chosen with distinguishing capability are critical for such newly developed technology (Luo et al., 2007).

There are different operations to modify the original image. Generally, the semantic content of an image can be modified in two main directions: through omitting information or through adding it. In order to exclude information, often forgers do not require access to other image content according to the top row in Figure 2.15(a). On the other hand, one common way to add some extraneous data to image is using material obtained from some specific images. The techniques of tampering, as well as their malicious intentions, can be categorized into two groups: those that create forgery in just one image and those which access the contents of multiple images (composite) (Redi et al., 2011).

- **Image copy-move:** Usually the attack named copy-move is utilized to conceal different parts of images or omit unwanted portions of it. So a portion from image is copied and then pasted over those unwanted portions in that image. Figure 2.15 provides an example for such copy-move forgery (Granty et al., 2010).



(a) (b)

(c)

Figure 2.15: Sample of image copy-move, (a) original image, (b) image after copy-move attack, (c) copied portions

- **Image retouching**: Image retouching is performed in a lot of magazine covers in order to give poor quality images an improved appeal through altering its background or via making modifications in the hue of that image so it will have a better feeling. In Figure 2.16 a retouched image is presented (Granty et al., 2010).

(a) (b)

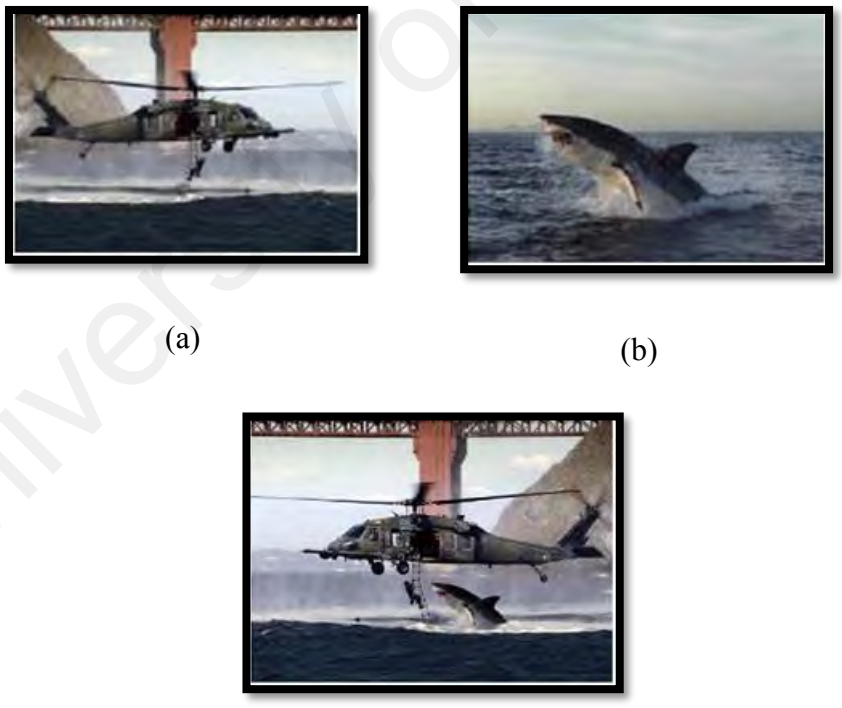
Figure 2.16: Sample of Image Retouching (a) Original Image, (b) Retouched Image

- **Geometrical transformation**: Some pictures have a part of the image modified by many popular geometric transformations for example rotation, scaling and translation. The forgers will copy a portion of the image and then make modifications to it through changing that specific portion geometrically. This operation is presented in Figure 2.17 (Granty et al., 2010). Image (a) shows only one cartridge while the second image (image (b)) demonstrates the image with two cartridges in which the second cartridge has been made by scaling and transformation operations.

(a) (b)

Figure 2.17: Sample of Geometrical Transformation. (a) Original Image, (b) Image after geometrical transformation

- **Image splicing**: This is a technique of tampering with photos through combining two different sources in order to generate a new picture that has most portions of one picture in details. Figure 2.18 shows an example of a spliced picture (Farid, 2006a).



(a) (b)

(c)

Figure 2.18: Sample of image splicing. (a) first original image, (b) second original image, (c) spliced image

2.6 Feature Extraction

Image splicing operations modify the smoothness and continuity of the image content. Thus image splicing detection methods needs some features to find the modification done during the splicing process. In section 2.9, different statistical feature extraction methods are presented and discussed. Features of the image contain local and global assets of a picture, for example average grey levels, intensity histogram circles, shapes, texels as well as the counter shapes are captured from the image to show the digital images within abstract or real mathematical presentations (Jin et al., 2009).

The image features are employed as critical assets which can be employed in the image-matching process; such image features should include proper assets in order to make them useful for identifying discontinuities within image contents.

The feature extraction could be possible using various algorithms. The most appropriate algorithms for feature extracting should bring exact feature recognition, which has low computational extraction costs. One asset of a useful algorithm is the extraction's low cost. Generally, there should be a balance between detection accuracy, dimensionality, and computation time.

Actually, image feature extraction is carried out to understand premise features. Currently, major types to extract features of an image are: 1) intuitive features including contour, edge of image, texture etc.; 2) grey demographic features, for example histograms; 3) transforming features domain for example Haar transform, Walsh transform and Fourier transform. Moreover, algebra features are usually employed to solve the image recognition issue. Also, SVD algebra is an efficient method for feature extraction which was broadly used, due to having the best properties for signal processing,

data compression and pattern recognition (Jin et al., 2009). Since SVD is applied in this research, its concepts and properties are discussed in this section.

2.6.1 Singular Value Decomposition

Regarding the linear algebra, SVD is in fact the factorization of the rectangular complex or real matrix analogous toward symmetric diagonalization or Hermitian square matrices utilizing the eigenvectors basis. SVD is an effective and consistent approach to splitting a system into a group of linearly independent aspects, each of them providing their own contribution to energy (Andrews & Patterson, 1976; Moonen et al., 1992). For one digital Image M with size equal to $m \times n$ together with $m \geq n$, a decomposition of this matrix could be as below (Sadek, 2012):

$$M = USV^T \quad \textbf{(2.7)}$$

$$Sv = \text{diagonal of } (S) \quad \textbf{(2.8)}$$

where U and V (V^T is transpose of V) are two square orthogonal matrices in sizes of $m \times m$ and $n \times n$ respectively, and S is a rectangular matrix of $m \times n$ in which its diagonal factors demonstrate the singular values, Sv_i for M. The orthogonal matrix U columns are known as left singular vectors while orthogonal matrix V columns are known as right singular vectors. The LSCs, left singular vectors for M are MM^T eigenvectors and RSCs, right singular vectors, for M are M^TM eigenvectors. Any singular value (SV) demonstrates the image luminance layer and the singular vectors' corresponding pair (SCs) shows the image geometry (Ganic et al., 2003). In addition, V and U are orthogonal unitary matrices (total of squares for each single column is unity and all of the columns are not correlated) so S will be a diagonal matrix (only a leading diagonal includes non-zero values) with the

singular values sorted in descending order. The singular value for each specific eigen image is simply 2-norm (Sadek, 2012). As SVD increases the biggest singular values, the first eigen image will be a pattern which covers the biggest portion of covariance-variance structure (Andrews & Patterson, 1976; Kamm, 1998) .

2.6.2 SVD Image Properties

SVD would be an optimal decomposition of a matrix within a least square sense, whereby only a few coefficients contain the highest possible signal energy (Konda & Nakamura, 2009; Moonen et al., 1992). The singular value decomposition, SVD, would therefore be the effective and fixed approach to split a system into two groups of linearly independent factors, each of them having their own proportion of energy.

The singular value decomposition is considered as a numerical method utilized to identify numerical analysis matrices (Andrews & Patterson, 1976; Kamm, 1998). This would be an interesting algebraic transform regarding image processing, due to its numerous benefits, for example maximum packing of energy that is usually employed in compression (Xu et al., 2004; Yang & Lu, 1995), its capability to handle the image based on two separate subspaces: noise and data subspaces (Konstantinides et al., 1996; Sverdlov et al., 2006; Xu et al., 2004), which is mainly employed in case of noise filtering. Moreover, it has been employed in watermarking processes (Gorodetski et al., 2001; Xu et al., 2004). All of the applications mentioned can exploit the main SVD properties.

In addition, it is often employed to solve least squares issues, multivariate analysis and matrix computing pseudo-inverse. The SVD is both a reliable and robust orthogonal matrix decomposition approach, which, because of its stability and conceptual properties, is becoming highly common in the field of signal processing (Andrews & Patterson, 1976; Kamm, 1998). The SVD can adapt to differences in domestic statistics of a digital photo

(Yang & Lu, 1995). A lot of SVD properties are interesting but are not yet wholly employed.

This part investigates key SVD properties which might be used in image processing. Even though many of these properties are totally used in image processing, other ones still require further study. Different SVD properties are remarkably beneficial for images, for example its full energy packing, calculating pseudo-inverse of the matrix, solving least squares issues, as well as multivariate analysis (Konda & Nakamura, 2009; Moonen et al., 1992). One of the main SVD properties is its relationship to matrix rank and its capability to approximate matrices of a certain rank. Often the digital images are introduced by low rank matrices and so can be defined in total by a small group of eigenimages. This structure can increase the alteration of signal into two specific subspaces (Andrews & Patterson, 1976; Kamm, 1998).

2.6.2.1 SVD Subspaces

The SVD consists of two subspaces (orthogonal subdominant and dominant). It shows the partition of vector space for M-dimensional toward both subdominant and dominant spaces (Moonen et al., 1992; Sverdlov et al., 2006). This useful attribute of SVD is used in both watermarking and noise filtering (Gorodetski et al., 2001; Konstantinides et al., 1996).

2.6.2.2 SVD architecture

In case of SVD image decomposition, the singular values will define the luminance of the image layer and the corresponding pair singular vectors show the image layer geometry. The biggest object feature in an image is identified by means of SVD in general corresponding to associated eigenimage with the biggest singular value, and the image

noise is related to an eigenimage that is related to SVs (Andrews & Patterson, 1976; Kamm, 1998).

2.6.2.3 PCA versus SVD

The PCA, principle component analysis, is also known as Karhunen-Loéve transform (KLT) or the Hotelling transform. The PCA is utilized to calculate the dominant vectors shown as a specific data set and brings the appropriate basis for reconstruction of the lowest mean squared for a set of data. The computational foundation of the PCA is the computation of SVD, or similarly the eigenvalues decomposition for SVD data covariance matrix is relevant to the spectral decomposition of the square matrix M or standard eigenvector-eigenvalues toward VLV^T in which V is in fact orthogonal and L is also diagonal. In addition, U and V from SVD represent respectively the eigenvectors for MM^T and M^TM. If M is symmetrical, the singular values of M are known as the eigenvalues for absolute value of M (Andrews & Patterson, 1976; Kamm, 1998).

2.6.2.4 SVD Multiresolution

SVD has the most efficient energy packing compared to other types of transform. In various applications, it would be helpful to achieve the statistical characterization of a certain image at many different resolutions. The SVD is capable to decompose a matrix as its orthogonal components by which proper sub-rank approximations might be achieved. Through the multiresolution SVD, the below critical features of an image could be evaluated, at each single level of many resolution levels: principle components spercity, isotropy, mean squared error as some meaningful components and self-similarity under scaling (Kakarala & Ogunbona, 2001; Yang & Lu, 1995).

2.6.2.5 SVD Oriented Energy

In the case of SVD oriented energy analysis, both rank of signal space and problem orientation can be defined, so SVD can be considered a proper and efficient technique to split a single system into a group of linearly independent features, with each one of them having its own contribution to energy (Sadek, 2012). The SVD is shown as the linear combination of its key components, and a few dominant features including a rank of observation system, could be minimized dramatically. The oriented energy notion is an appropriate method to divide signals from various sources or choosing signal subspaces of the maximal signal integrity and activity (Moonen et al., 1992; Sadek, 2008). It should be noted that singular values show the energy square root in the relevant principal direction. The dominant direction can also be equal to the first singular vector *V1* in SVD. The accuracy of the dominance estimate can be evaluated by calculating the normalized difference or simply the difference among first two SVs (Bigun et al., 1991).

2.6.2.6 SVD and Linear Independence

The other utilization of SVD brings a measure known as the condition number, which is relevant to the linear independence measure among matrix column vectors. The matrix *M* condition number, considering the Euclidean norm is (Leach, 1995):

$$Cond(M) = \frac{s_{max}}{s_{min}} \tag{2.9}$$

where s_{min} and s_{max} means the minimum and maximum singular values of matrix *M*. Matrix *M* is rank deficient, so if $s_{min} = 0$, it is assumed $cond(M) = \infty$. By means of condition number, the column's independency can be determined. Thus, for all matrices $cond(M) \geq 1$, the columns are highly independent when $cond(M)$ is very close to 1.

Referring to singular values geometric interpretation, it is clear that the condition number is relevant to the hyperellipsoid axes related to the matrix. Because *cond(M)* is presented by a set of singular values, and such values show the length of axes with a maximum and minimum range, the condition number will define the hyperellipsoid eccentricity (Leach, 1995).

If the condition number is larger, the matrix is closer to being singular. Thus the identity matrix condition number equals 1. The singular matrix has an infinite condition number (Renkjumnong, 2007).

2.6.2.7 SVD-based orthogonal subspaces and rank approximation

In fact, SVD can decompose the matrix as some orthogonal features with which approximations of optimal sub rank might be calculated. The biggest object features in images identified utilizing SVD will in general correspond to those associated eigenimages with biggest singular values, and when image noise is related to those associated eigenimages, with small singular values. In addition, SVD is utilized to approximate the matrix, which decomposes the data as the optimal estimate of noise and signal components. This attribute is one of the critical aspects of SVD decomposition in case of compression, forensic and noise filtering, which also can be treated as adding more noise in an appropriate detectable manner (Sadek, 2012).

2.6.3 Feature Extraction Based on SVD

The SVD decomposition is able to transform any matrix to a diagonal matrix form. In the case of image processing, the below are the key theoretical reasons to apply SVD: 1) the image singular values stability are more efficient. Since the image is being imposed on some small disturbance, the singular values of the image will not have any specific

changes; 2) the image singular values are demonstrated instead of the intrinsic attributes of visual characteristics. From the linear algebra point of view, a grey image could be considered as a non-negative matrix, if the image is defined as M and expressed as $M \in R^{n\times n}$. Thus, SVD decomposition definition regarding the matrix M is the equation below (Wensheng WANG et al., 2006):

$$M = USV^T = (u_1, u_2, \ldots, u_n)\begin{bmatrix} \sqrt{\alpha_1} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \ldots & \sqrt{\alpha_n} \end{bmatrix}(v_1, v_2, \ldots, v_n) \quad \textbf{(2.10)}$$

Both $U \in R^{n\times n}$ and $V \in R^{n\times n}$ are real matrices, $\alpha_1 \geq \alpha_2 \geq \ldots \geq \alpha_n$, $\sqrt{\alpha_i}$ is singular value of M that could be defined specifically through formula (4) and $\sqrt{\alpha_i}$ would be the square root of the MM^T eigenvalue (Fu-bing & Jing-yu, 2005).

For any kind of real matrix M, decomposition of singular value is totally unique in $\alpha_1 \geq \alpha_2 \geq \ldots \geq \alpha_k$ limitations. Based on the idea that SVD decomposition of the image matrix is totally unique, the singular value eigenvector can be utilized for the image matrix in order to define a 2-D grey image. The singular vectors via dimensional transformation have a lot of critical properties as image features:

1) The stability of displacement and image dimensional transformation, which is a technique according to dimensional transformation to show that a matrix of an image of the same object includes good and accepted immutability. Specifically, it has to fulfill the condition below for dimensional transformation and image movement (Jin et al., 2009):

$$\|M - L\|_F < \varepsilon \quad \textbf{(2.11)}$$

M is the dimensional transformation matrix to original image and so L is the dimensional matrix via zoom or movement transformation to the original image. Based

on this theory, it can be confirmed whether the SVD of a matrix is stable. With small modifications in the matrix, SV eigenvectors will not have any specific changes. The definition below shows this matter (Jin et al., 2009; J. Lu & Wang, 2005).

Hypothesis: $M, L \in R^{n\times n}$ with singular values of $\alpha_1 \geq \alpha_2 \geq \ldots \geq \alpha_n$ and $\beta_1 \geq \beta_2 \geq \ldots \geq \beta_n$ respectively, and $M - L = \delta$, so the equation below is justifiable to any immutable norm on $R^{n\times n}$ (Jin et al., 2009):

$$\|diag(\alpha_1 - \beta_1, \ldots, \alpha_n - \beta_n)\| \leq \|L - M\| = \delta \quad \textbf{(2.12)}$$

When the norm is $\|M\|_F = \sqrt{\sum_{ij}|a_{ij}|^2}$, then Equation (2.12) will be as follows (Jin et al., 2009):

$$\sqrt{\sum_{i=1}^{n}(\beta_i - \alpha_i)^2} \leq \|L - M\|_F \quad \textbf{(2.13)}$$

Moreover, to calculate dimensional transformation and image movement, the following equation can be derived from Equations (2.12) and (2.13) (Jin et al., 2009):

$$\sqrt{\sum_{i=1}^{n}(\beta_i - \alpha_i)^2} \leq \varepsilon \quad \textbf{(2.14)}$$

2) The rotation transformation stability for an image based on new matrix technique according to dimensional transformation is utilized to provide an image, any kind of rotation transformation is similar to the relevant row replacement for the image matrix. Based on matrix theory, *i* and *j* rows of *A* switching matrix are as same as left side of matrix that are being multiplied through below equation (Jin et al., 2009):

$$A_{ij} = A - (a_i - a_j)(a_i - a_j)^T \quad \textbf{(2.15)}$$

where a_i and a_j are the *ith* and *jth* elements of matrix *A* respectively. If *M* is considered as the original image, A_{ij}. *M* shows the image when rotation transformation is performed, thus $(A_{ij}.M)(A_{ij}.M)^T$ will be (Jin et al., 2009):

$$\left|(A_{ij}.M)(A_{ij}.M)^T - \alpha A\right| = 0 \quad \textbf{(2.16)}$$

Since $A_{ij} = A_{ij}^T = A_{ij}^T$, thus Equation (2.16) can be written as follows (Jin et al., 2009):

$$\left|A_{ij}.MM^T.A_{ij}^T - \alpha A\right| = |MM^T - \alpha A| = 0 \quad \textbf{(2.17)}$$

According to this point, original image *M* and after rotation have same *Aij*. Thus an eigenvector has the property of rotation transformation immutability.

It is understood that the extracted features must have geometry and algebra invariability for the extracted techniques of image recognition (Jin et al., 2009). The above analysis demonstrates that *Sv* features have such stability for a matrix according to dimensional transformation to introduce a single image (JING et al., 1999).

2.6.4 Why SVD?

Based on the brief explanation of SVD in section 2.6, SVD is a strong method in various matrix analyses and calculations. Employing SVD of the matrix in calculations, instead of the original matrix, includes the benefit of robustness to numerical error. In addition, SVD exposes the matrix geometric structure, a critical dimension of a lot of matrix computations. The matrix could be defined as a transformation from one correlated vector space to an uncorrelated one. The SVD components can quantify the changes among the underlying geometry of such vector spaces.

SVD can be used in many applications such as least-square issues to solving the linear equations of systems, image processing and signal processing. Each of them uses main SVD properties – its relationships to matrix rank and its capability to approximate the matrices of a specific rank. A lot of critical linear algebra dimensions rely on defining the matrix rank, making SVD a highly used and important method. Therefore, SVD is applied in this research as a feature extraction technique in an image splicing detection method to investigate its effectiveness in this area of image application (Leach, 1995).

2.7 Feature Selection

The detection step in image splicing detection is performed by classifying the authentic and spliced images. Before starting the actual data classification, it is essential to process the data extracted from the images for various reasons, including redundancy, missing values, data outliers, inappropriate format, and data inconsistency. This step is called feature selection.

In many real-world applications suitable preprocessing transformations of input data can increase overall performance of algorithms (Rosipal et al., 2001). In general, there are some correlations and redundancies among input variables; thus feature selection restricts the input data by eliminating redundant features and keeping important dimensions in the feature vector. Humans and machine learning methods find it difficult to interpret raw data. Given that a feature matrix has rows which each represent a specific instance of an object, a large number of features exponentially increases the computational time. Thus, transforming the information into smaller sizes enhances method analysis and improves the training and testing phases during classification (Anusudha et al., 2010). Several experiments were conducted to test and analyze this idea.

Different approaches such as finding the linear or nonlinear manifold that lies within the high-dimensional data space can simplify interpretation. In this section, the PCA and kernel PCA are presented to improve the features extracted by the proposed SVD-based feature extraction techniques. Moreover, linearity (PCA) and nonlinearity (kernel PCA) were considered in selecting the feature selection techniques, in order to investigate their effects on the extracted features.

2.7.1 PCA

PCA is the most common and popular linear feature selection approach (Ghodsi, 2006; Jackson; Jolliffe, 1986). It has been used for years because of its conceptual simplicity and computational efficiency. The approach is applied in many areas such as noise reduction, pattern recognition, regression estimation, and image indexing (Schölkopf et al., 1997). It maps a dataset of n dimensions to a linear subspace with d dimensions, where $d < n$, and attempts to maintain most of the variability in the mapped dimensions. PCA is considered a second-order approach depending on the covariance matrix of the variables. The approach has different names in different fields such as singular value decomposition, Karhunen–Love transform, Hotelling transform, and the empirical orthogonal function method (Fodor, 2002).

PCA is based on finding the d orthogonal linear vectors, known as principal components, of n dimensions with maximum variance. Therefore, the number of selected features is not more than n. The approach works well if the most significant modes of variability are almost linear. Hence, high dimensional samples are best remade from their low dimensional linear projections. Otherwise, PCA becomes ineffective if the most vital significant modes of variability are nonlinear (Ghodsi, 2006). In mathematical terms,

PCA finds Y as the new feature vector set with d dimension ($d \leq D$), in which X is the original feature vector set with D dimension (van der Maaten et al., 2009).

$$\mathrm{Y} = \mathrm{XM} \tag{2.18}$$

To find linear mapping M, PCA attempts to maximize the following function:

$$\mathrm{M^T cov(X) M} \tag{2.19}$$

where $cov(X)$ is the covariance of the original feature vector set X. However, M consists of d principal eigenvectors of the sample covariance matrix of the zero-mean data (van der Maaten et al., 2009). Therefore, the following eigen problem must be solved for the d principal components λ:

$$\mathrm{cov(X) M} = \lambda \mathrm{M} \tag{2.20}$$

2.7.2 Kernel PCA

PCA is a linear feature selection method. Some datasets have a nonlinear nature, and PCA cannot select the features of these datasets efficiently. Thus, kernel PCA was designed to address this problem. Kernel PCA was applied in some pattern recognition experiments (Schölkopf et al., 1997) and exhibited better recognition rates than linear PCA. Kernel PCA is a nonlinear form of PCA that attempts to identify complicated correlations between given features. It computes principal components in the original dataset through nonlinear mapping. It also discovers major components that are nonlinear in relation to the input space by running, which results from nonlinear mapping in which the low-dimensional hidden structures are likely to be simple (Ghodsi, 2006).

Kernel PCA locates the principal eigenvectors of the kernel matrix instead of the covariance matrix (van der Maaten et al., 2009). Thus, the computational complexity of kernel PCA is independent of the dimensionality of the feature set, which allows it to work on feature sets with different possible dimensionalities. Kernel PCA does not require any nonlinear optimization; it only needs to solve an eigenvalue problem as in the case of standard PCA. Thus, kernel PCA is free of local minima trap during training. The original feature set must be mapped to a higher dimensional feature set to calculate kernel PCA (Schölkopf et al., 1997):

$$\Phi : R^N \to F, x \to X \tag{2.21}$$

Then, the covariance matrix of data is calculated to obtain the principal components by solving the eigenvalue problem using the following equations (Schölkopf et al., 1997):

$$C_F = \frac{1}{N}\sum_{1}^{N} \Phi(x_i)\Phi(x_i)^T \tag{2.22}$$

$$C_F v = \lambda v \tag{2.23}$$

Subsequently, the eigenvector can be expressed as a linear combination of features (Schölkopf et al., 1997):

$$v = \sum_{1}^{N} \alpha_i \Phi(x_i) \tag{2.24}$$

$$\alpha_i = \frac{1}{\lambda N} v \tag{2.25}$$

Therefore, the kernel matrix is defined as follows:

$$k_{ij} = \mathcal{K}(x_i, x_j) = \left(\Phi(x_i).\Phi(x_j)\right) = \Phi(x_i)^T \Phi(x_j) \tag{2.26}$$

where *k_ij* represents the elements of kernel matrix *K*, *x* is the feature set, and $\mathcal{K}$ is the kernel function with conditions that result in a positive semi-definite kernel *K* (van der Maaten et al., 2009). *Φ(x_i)* may not be zero-mean, such that the features must be centered. The corresponding kernel is obtained using the following equation (van der Maaten et al., 2009):

$$k_{ij}^{c} = k_{ij} - \frac{1}{N}\sum_{i=1}^{N} k_{ik} - \frac{1}{N}\sum_{j=1}^{N} k_{jk} + \frac{1}{N^2}\sum_{i,k}^{N} k_{ik} \quad \textbf{(2.27)}$$

Consequently, the following equation represents the low-dimensional feature set *y_i*:

$$y_i = \sum_{i=1}^{N} \alpha_{ji}\mathcal{K}(x, x_i) \quad \textbf{(2.28)}$$

where *α_ji* represents the *j*th value in the vector *α_i*.

2.8 Classification

Image splicing detection is a two-class classification procedure. It needs an appropriate binary classifier in order to differentiate between spliced and authentic images. SVMs, Support Vector Machines, are initiated from Statistical Learning Theory (Vladimir & Vapnik, 1995). They have been broadly used in fields of machine vision, for example handwriting digits, character and text recognition (Joachims, 1998; Vladimir & Vapnik, 1995), classifications of satellite image (C. Huang et al., 2002; Mather & Koch, 2011) as well as image forgery detection (Shi et al., 2007b; Zhao et al., 2011). Some SVMs such as Artificial Neural Networks, as well as other classifiers which are nonparametric, are well-known for their robustness (Foody & Mathur, 2004a, 2004b). The functionality of SVMs is through nonlinearly projecting the training data inside the input space to have an infinite space dimension using kernel function. These outcomes in linearly specific

datasets can be divided through linear classifiers. Such a process makes the classification able to remote those sensing datasets that are often nonlinearly separable in input spaces.

In a lot of examples, classification in feature spaces of high dimensionality leads to over-fitting within an input space, but, in over-fitting of SVMs, it is monitored by a structural risk minimization principle (Vladimir & Vapnik, 1995). The misclassification risks are empirically reduced through increasing the margin among decision boundaries and data points (Mashao, 2003). In practice, such criteria will be softened to reduce cost factors, including both classifiers' complexity and the extent to which marginal points are being misclassified. The existing trade-off among such variables can be managed via an error parameter of margins (mainly the designated C) that is appropriately tuned via a cross-validation processes (Mashao, 2003). Those functions which are projecting the data from input space into the feature space are known as kernel machines or simply kernels, including Gaussian (generally known as radial basis functions), polynomial and quadratic functions. These functions have specific parameters that must be defined before classification, and they are often defined via a process of cross-validation. A more in-depth SVMs mathematical treatise can be identified in (Anthony et al., 2007; Campbell, 2001; Cristianini & Shawe-Taylor, 2000; Vladimir & Vapnik, 1995). In this study, SVM is applied as a classifier to distinguish the authentic and spliced images.

2.9 Image Splicing detection methods

An insertion of material from other sources into an image is one of the most well-known methods to overturn the contained message in visual media. Modern methods and new editing software provide us with convenient composite image creation tools, resulting in outcomes that are not easily detectable by the human eye. The matting and blending methods are useful to mask all of the boundaries of spliced areas and to grant

the image highly uniform dimensions. Moreover, the composite image creation may need geometric transformations. Scaling, rotation and also translation are usually required to ensure that spliced object is in line with the original image scale and perspective. Typically, the geometric transforms include re-sampling, which requires interpolation in turn (bicubic, bilinear and closest neighbor). The process of re-sampling generates some artifacts in the image histogram, and so gives an appropriate clue for the detection of composites.

It should also be considered that inserted material does not necessarily have to initiate from natural pictures. Since computer graphics are evolving, more realisitc objects in 3D can be rendered and modeled to be spliced as image composite. In addition, the 3D scene structure extraction from images lets us alter objects through a morphing process, and here the splicing includes an artificial or remodeled version of a portion of an original image (Redi et al., 2011). Such methods are used while the forger attempts to manipulate a facial expression as examined in the rewriting of videos (Bregler et al., 1997).

So far, several blind (passive) image splicing detection methods have been proposed and developed, and their number is still rising. These different methods have similar procedures, including preprocessing, feature extraction, and classification. The main differences between the existing methods are in the preprocessing and feature extraction phases. The extracted features have different sizes and algorithms. Therefore, this section classifies the existing approaches based on the preprocessing and feature extraction algorithms applied. In next parts of this section, some of the most influential methods for image splicing detection will be reviewed.

2.9.1 Markov-based algorithms

This category describes the features extracted by the Markov algorithm. Markov features reflect the statistical changes caused by image splicing. In fact, it characterizes the correlation between image pixels that are changed by the image splicing process. Shi et al. (Shi et al., 2007b) proposed a natural image model by combining moments of characteristic functions of wavelet sub-bands and the Markov transition probabilities of the difference between 2-D arrays and 2-D arrays of multi-size block discrete cosine transform (MBDCT). They applied the Markov process on the difference 2-D arrays that were computed along horizontal, vertical, diagonal, and minor diagonal directions instead of applying it to the image/coefficient 2-D arrays directly.

Their model is based on probability theory (Viniotis, 1998), which assumes the spliced process to be an additive noise to the original image and thus the spliced image is a convolution of distribution of the original image and the splicing noise. According to the shape of the distribution of additive splicing noise, the concentration along specific directions in a Markov transition probability matrix could be found and indicate image splicing. The equations to compute the elements of transition probability matrix (features) are as follows (He et al., 2012; Shi et al., 2007b):

$$P1_h(i,j) = \frac{\sum_{u=1}^{S_u-2}\sum_{v=1}^{S_v}\delta(E_h(u,v)=i, E_h(u+1,v)=j)}{\sum_{u=1}^{S_u-2}\sum_{v=1}^{S_v}\delta(E_h(u,v)=i)} \quad \textbf{(2.29)}$$

$$\mathrm{P1_v(i,j)} = \frac{\sum_{\mathrm{u=1}}^{\mathrm{S_u-1}}\sum_{\mathrm{v=1}}^{\mathrm{S_v-1}}\delta(\mathrm{E_h(u,v)=i, E_h(u,v+1)=j})}{\sum_{\mathrm{u=1}}^{\mathrm{S_u-1}}\sum_{\mathrm{v=1}}^{\mathrm{S_v-1}}\delta(\mathrm{E_h(u,v)=i})} \quad \textbf{(2.30)}$$

$$\mathrm{P2_h(i,j)} = \frac{\sum_{\mathrm{u=1}}^{\mathrm{S_u-1}}\sum_{\mathrm{v=1}}^{\mathrm{S_v-1}}\delta(\mathrm{E_v(u,v)=i, E_v(u+1,v)=j})}{\sum_{\mathrm{u=1}}^{\mathrm{S_u-1}}\sum_{\mathrm{v=1}}^{\mathrm{S_v-1}}\delta(\mathrm{E_v(u,v)=i})} \quad \textbf{(2.31)}$$

$$\mathrm{P2_v(i,j)} = \frac{\sum_{\mathrm{u=1}}^{\mathrm{S_u}}\sum_{\mathrm{v=1}}^{\mathrm{S_v-2}}\delta(\mathrm{E_v(u,v)=i, E_v(u,v+1)=j})}{\sum_{\mathrm{u=1}}^{\mathrm{S_u}}\sum_{\mathrm{v=1}}^{\mathrm{S_v-2}}\delta(\mathrm{E_v(u,v)=i})} \quad \textbf{(2.32)}$$

Zhang et al. (J. Zhang et al., 2009) developed a splicing detection method by merging the Markov features applied in (Shi et al., 2007a) and DCT features. Their detection method achieved an accuracy rate of 91.5%, with the use of the 109-D feature vector.

Sutthiwan et al. (Sutthiwan et al., 2010) developed an image splicing detection method for color images based on MBDCT, moment-based and Markov-based features. They extracted features from the Cr channel in the chroma channel space, which is the most sensitive channel in the YCbCr during color image tampering. Their natural image model is based on the one developed in (Shi et al., 2007b). It took the advantage of the de-correlation which exists between block discrete cosine transform (BDCT) coefficients. At the end, they applied boosting feature selection (BFS) to make an optimal selection between dimensions and to increase the accuracy rate.

Zhao et al. (Zhao et al., 2012) proposed an image splicing detection method using the third-order statistical feature extraction method. They applied the conditional co-occurrence probability matrix (CCPM) and combined it with Markov transition probability matrix and co-occurrence matrix. The Markov chain reflects the dependences between two neighboring states. They expanded the basic Markov chain features to third order statistical features by considering the correlations between three neighboring states which shows the future state by determining the current state and previous state. Since the dimensionality increases by increasing the number of orders in the features, they also applied principle component analysis (PCA) to reduce dimensionality.

He et al. (He et al., 2012) proposed a Markov-based image splicing detection algorithm, which expands the DCT Markov approach developed in (Shi et al., 2007b) and combines it with Markov features extracted from the DWT domain. The expanded DCT Markov has the advantage of capturing both the intra-block and inter-block correlations

between the DCT coefficients as proposed in (C. Chen & Shi, 2008). The equations for the expanded parts are as follows (He et al., 2012):

$$G_h(i,j) = x(i,j) - x(i+8,j) \quad \textbf{(2.33)}$$

$$G_v(i,j) = x(i,j) - x(i,j+8) \quad \textbf{)2.23(}$$

$$P3_h(i,j) = \frac{\sum_{u=1}^{S_u-16}\sum_{v=1}^{S_v}\delta(G_h(u,v)=i, G_h(u+8,v)=j)}{\sum_{u=1}^{S_u-16}\sum_{v=1}^{S_v}\delta(G_h(u,v)=i)} \quad \textbf{(2.34)}$$

$$P3_v(i,j) = \frac{\sum_{u=1}^{S_u-8}\sum_{v=1}^{S_v-8}\delta(G_h(u,v)=i, G_h(u,v+8)=j)}{\sum_{u=1}^{S_u-8}\sum_{v=1}^{S_v-8}\delta(G_h(u,v)=i)} \quad \textbf{(2.35)}$$

$$P4_h(i,j) = \frac{\sum_{u=1}^{S_u-8}\sum_{v=1}^{S_v-8}\delta(G_v(u,v)=i, G_v(u+8,v)=j)}{\sum_{u=1}^{S_u-8}\sum_{v=1}^{S_v-8}\delta(G_v(u,v)=i)} \quad \textbf{(2.36)}$$

$$P4_v(i,j) = \frac{\sum_{u=1}^{S_u}\sum_{v=1}^{S_v-16}\delta(G_v(u,v)=i, G_v(u,v+8)=j)}{\sum_{u=1}^{S_u}\sum_{v=1}^{S_v-16}\delta(G_v(u,v)=i)} \quad \textbf{(2.37)}$$

where $i,j \in [-T,-T+1,\dots,0,\dots,T-1,T]$, S_u, and S_v are the dimensions of the original image and

$$\delta(A=i,B=j) = \begin{cases} 1, & \text{if } A=i \text{ and } B=n \\ 0, & \text{otherwise} \end{cases} \quad \textbf{(2.38)}$$

In addition, they took advantage of multi-resolution analysis in the DWT domain to characterize the residual correlation by modeling the three kinds of dependency among wavelet coefficients across positions, scales and orientations. Many of the approaches usually apply wavelet sub-bands independently, while dependencies exist across positions, scales and orientations (Srivastava et al., 2003). In (He et al., 2012), this property was merged with Markov features to produce an efficient tool for image splicing detection. Subsequently a feature selection method (SVM-RFE) was implemented to reduce the computational cost.

Markov features have been verified as one of the most effective features for image splicing detection methods. However, its dimensionality is very high and depends on the threshold that is determined in the feature extraction.

2.9.2 Moment-based algorithms

Moment-based algorithms are other image splicing detection algorithms. Similar to Markov-based methods, moment-based algorithms reflect the statistical changes caused by image splicing methods. They are based on moments of characteristic function using prediction-error 2-D array and wavelet decomposition. One advantage of using prediction-error 2-D array is to decrease the effects caused by diversity of the image content and concurrently to enhance the statistical artifacts introduced by splicing. Furthermore, the moment-based methods take the advantage of decorrelated coefficients of the same level in different wavelet subbands.

An example of applying moment-based algorithms in detecting image splicing is the method proposed in (Fu et al., 2006) and (X. Li et al., 2010). They extracted the features using Hilbert-Huang Transform (HHT) and a moment-based model to distinguish the authentic images from the spliced ones.

Shi et al. (Shi et al., 2007b) also proposed an image splicing detection method based on natural image model. Their statistical features consisted of moments of characteristic functions of wavelet sub-bands and Markov transition probabilities of difference 2-D arrays. The moment features are basically extracted from the 1-D characteristic functions (discrete Fourier transform (DFT) of the first-order histograms), as well as from the 2-D characteristic functions (2-D DFT of the second-order histograms). The second-order histogram contains more information than the first-order in detecting splicing in an image

as it includes two pixels at a time. They followed the block diagram indicated in Figure 2.19 to extract the moment features:

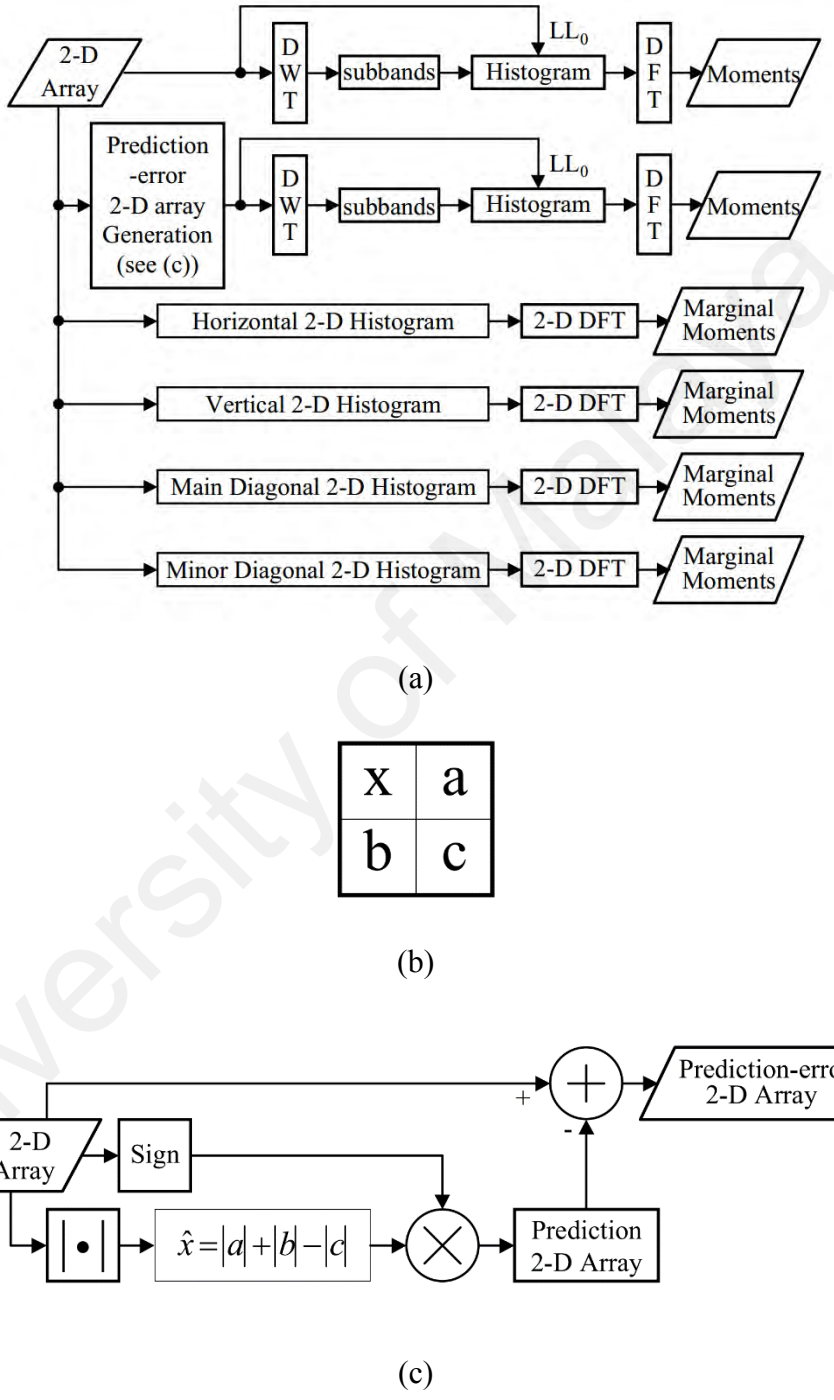


Figure 2.19: Moment extraction feature, (a) general block diagram, (b) prediction context, (c) prediction error 2-D array generation

The first diagram contains capturing statistical moments from DWT sub-bands. As mentioned earlier, DWT is applied in image forensics getting an advantage in reflecting transient or localized changes in spatial and frequency domain. They applied DWT to the image pixel 2-D array, and prediction-error 2-D arrays to get the statistical moments (Shi et al., 2007b). A prediction-error 2-D array is applied to decrease the diversity of image content by predicting the value of a pixel using its three neighboring pixels. The equations to calculate the prediction-error 2-D array are as follow (Shi et al., 2007b):

$$\bar{x} = sign(x) \cdot \{|a| + |b| + |c|\} \quad \textbf{(2.39)}$$

$$\Delta x = x - \bar{x} = x - sign(x) \cdot \{|a| + |b| + |c|\} \quad \textbf{(2.40)}$$

To calculate the statistical moments, the 1-D characteristics function (CF), which is the DFT of the first-order histogram of each wavelet sub-band is needed. The histogram of every image is considered as probability mass function (pmf) as it is determined by discrete values. If pmf is multiplied by a correspondingly shifted unit impulse results in a probability density function (pdf). Now, if unit impulses in DFT are ignored, then pdf could be considered as pmf and therefore the histogram of an image. Therefore, CF is simply the Fourier transform of the pdf (with a reversal in the sign of the exponent). The absolute moments of the 1-D CF are defined as follows (Shi et al., 2007b):

$$M_l = \frac{\sum_{i=1}^{K/2} x_i^l |H(x_i)|}{\sum_{i=1}^{K/2} |H(x_i)|} \quad \textbf{(2.41)}$$

where $H(x_i)$ is the CF component at frequency x_i, K is the total number of different values assumed by all of coefficients in a sub-band under consideration, i.e., the tap length of DFT, and l is an integer representing the order of moment.

Furthermore, Shi et al. (Shi et al., 2007b) enhanced the splicing detection capability by applying second-order statistics into their natural image model, since they are more efficient than first-order ones, which consider one pixel at a time and do not reflect the intensity/position correlation among neighboring pixels.

It is considered that a second-order histogram (Pratt, 1991) is a joint occurrence measure for pairs of separated pixels through a certain orientation and distance. Such separation is denoted via the existing distance between two pixels ρ, at a line angle which links the two pixels regarding the horizontal axis, θ. Hence, the second-order histogram is calculated as (Shi et al., 2007b):

$$h_d(j_1, j_2;\ \rho, \theta) = \frac{N(j_1, j_2;\ \rho, \theta)}{N_T(\rho, \theta)} \tag{2.42}$$

where $N(j_1, j_2, \rho, \theta)$ equals the number of pixel pairs and the value of the first pixel is j_1 and the value of the second pixel is j_2 , and also $N_T(\rho, \theta)$ equals the total amount of pixel pairs in the picture with a separation (ρ, θ). In addition, the second-order histogram is known as the co-occurrence matrix or dependency matrix.

After using 2-D DFT for the second-order histogram, the 2-D CF is achieved. The two marginal moments regarding 2-D CF so are computed by (Shi et al., 2007b):

$$M_{u,l} = \frac{\sum_{j=1}^{K/2}\sum_{i=1}^{K/2} u_i^l \left|H(u_i, v_j)\right|}{\sum_{j=1}^{K/2}\sum_{i=1}^{K/2}\left|H(u_i, v_j)\right|} \tag{2.43}$$

$$M_{v,l} = \frac{\sum_{j=1}^{K/2}\sum_{i=1}^{K/2} v_i^l \left|H(u_i, v_j)\right|}{\sum_{j=1}^{K/2}\sum_{i=1}^{K/2}\left|H(u_i, v_j)\right|} \tag{2.44}$$

where $H(u_i, v_j)$ is a component of 2-D CF at DFT frequency (u_i, v_j) as well as l being the integer demonstrating the moment order.

Zhang et al. (Z. Zhang, Zhou, et al., 2008) proposed an innovative splicing detection method applying image quality metrics and moment features. They obtained 42-D moment features on MBDCT (2, 4, and 8) arrays. Then combined these features with those extracted from seven image quality measures (IQM). The IQMs applied in their method are the ones most sensitive to splicing detection. The advantage of using IQMs is to reliably predict either perceived quality across different scenes and distortion types or to predict algorithmic performance.

Anusudha et al. (Anusudha et al., 2010) also developed an image splicing detection method by combining first-order moment-based with phase congruency, which is more sensitive to sharp transitions than feature extraction methods. They also applied a neural network as their classifier, and their results show a detection accuracy of 91.70%.

Generally, accuracy based on moment features is almost low and not satisfying. Moment-based features are usually applied in combination with other methods, and are therefore difficult to be analyzed individually.

2.9.3 Hilbert-Huang Transfer (HHT)

Another category of image splicing detection algorithm belongs to Hilbert-Huang Transfer based algorithms. Since image splicing detection process is a time shifting, nonlinear and non-stationary procedure, HHT is a suitable tool to analyze it. One example of the work related to HHT-based image splicing detection algorithm is used in (Fu et al., 2006). They combined HHT-based and moment-based feature extraction methods to detect the spliced images from the original ones.

HHT comprises of two parts, Empirical Mode Decomposition (EMD) and Hilbert Transform (N. E. Huang et al., 1998). Then the characteristics of the Intrinsic Mode

Functions (IMFs) are extracted from Hilbert spectral analysis (HAS). An intrinsic mode function (IMF) is a function that satisfies the following two conditions (N. E. Huang et al., 1998):

- In the whole data set, the number of extrema and the number of zero crossings must either equal or differ at most by one.
- At any point, the mean value of the envelope defined by the local maxima and the envelope defined by the local minima is zero.

The sifting algorithm is applied to extract the IMFs of a signal (N. E. Huang et al., 1998). Then the data is divided into several IMF components and Hilbert transform is performed on each component. For a given signal *x(t)*, the Hilbert transform can be expressed as follows (Fu et al., 2006):

$$y(t) = \frac{1}{\pi} P \int_{-\infty}^{\infty} \frac{x(\acute{t})}{t - \acute{t}} d\acute{t} \quad \textbf{(2.45)}$$

where *P* indicates the Cauchy principal value. With this definition, *x(t)* and *y(t)* can be used to define an analytic signal *z(t)* (Fu et al., 2006):

$$z_i(t) = x_i(t) + jy[x_i(t)] = a_i(t)e^{j\varphi_i(t)} \quad \textbf{(2.46)}$$

where *ai*(*t*) and *φi*(*t*) are the amplitude and phase of this analytical signal, which can be expressed as follows (Fu et al., 2006):

$$a_i(t) = \sqrt{x_i^2(t) + y^2[x_i(t)]} \quad \textbf{(2.47)}$$

$$\varphi_i(t) = \arctan \frac{y[x_i(t)]}{x_i(t)} \quad \textbf{(2.48)}$$

Furthermore, the instantaneous frequency is defined as (Fu et al., 2006; X. Li et al., 2010):

$$f(t) = \frac{1}{2\pi}\frac{d\varphi_i\ (t)}{dt} \quad \textbf{(2.49)}$$

HHT is mostly applied in signal processing. This method was not a promising one in image splicing detection, though. Its detection rate is only 85.87%. However, the proposed algorithm is low-dimensional with only 72-D.

2.9.4 Run-Length (RL) Algorithms

Run-Length is not applied extensively as a feature extraction method, but the results obtained in (Zhao et al., 2011) show that this method can be used as an image splicing detection approach. While the run length of an image represents its structure and texture, the splicing procedure modifies the pixel correlations and the structure of the image. Therefore, the RL feature extraction method can represent discontinuities and nonconformity and can be efficiently used as an image splicing detection method (Dong et al., 2009; Zhao et al., 2011).

This method was first used by (Galloway, 1975) in texture analysis to classify a set of terrain samples. Later, (Tang, 1998) developed a new run-length algorithm for the extraction of texture features based on the multi-level dominant eigenvector estimation method that improves classification accuracy.

The early work in this category was performed by (Dong et al., 2009) which applied RL histograms based on a similar method used in steganalysis algorithms (Dong & Tan, 2008). The first three moments of characteristic functions are extracted from RL histograms in each of four directions. They also combined the RL features with an edge-based statistics moment as a feature extraction method to reflect the inconsistencies of global pixel correlations made by image splicing techniques.

Zhao et al. (Zhao et al., 2011) studied run-length as a single feature extraction method for detecting image splicing in color images. A run in an image is defined as the number of pixels with the same grey-level value in a specific direction. For a given image, a run-length matrix $p_\theta(i,j)$ is defined as the number of runs with grey level i and run length j along a specific direction. Hence, the run-length vector is defined as follows (Zhao et al., 2011):

$$p_{\theta r}(j) = \sum_{i=1}^{M} p_\theta(i,j) \quad 1 \leq j \leq N \tag{2.50}$$

where M shows the number of grey levels and N is the maximum value of the run lengths. Vector $p_{\theta r}(j)$ demonstrates the sum distribution of runs with length j in a given image. In Equation (2.50), the run length represents the spread of the structure and the texture of the image. For example, the image with a long run length is smoother than that with a short run length because the latter consists of different regions with different structures. To equally emphasize all run lengths, a grey level run-length pixel number matrix is used, which is defined as follows (Tang, 1998; Zhao et al., 2011):

$$p_{\theta pr}(j) = \sum_{i=1}^{M} p_{\theta p}(i,j) \quad 1 \leq j \leq N \tag{2.51}$$

where

$$p_{\theta p}(i,j) = p_\theta(i,j) \,.\, j \tag{2.52}$$

$p_{\theta pr}(j)$ is the feature vector applied in (Zhao et al., 2011) and is referred to as RLRN. Four RLRN vectors are captured in four directions (0◦, 45◦, 90◦, and 135◦) to distinguish the spliced images from the authentic ones.

He et al. (He et al., 2011) improved the original RL method by applying approximate RL with higher accuracy and fewer features. Their proposed method was based on the idea that all the run lengths are not affected in the splicing procedure. Thus, only the run lengths captured from the edge pixels can be used in the detection procedure as the splicing methods modify the extra edges by making them sharper than the original ones. They also compute the runs along respective gradient directions of each edge pixel.

The RLRN method is simply computed and its detection accuracy is high in chroma channels (94.7%). However, the detection rate of RLRN is only high in colored images (chroma channels), while for greyscale images it has very low accuracy.

In 2014, Moghaddasi et al. (Moghaddasi, Jalab, Md Noor, et al., 2014) improved the RLRN image splicing detection method (Zhao et al., 2011) by applying a dimension reduction method such as Kernel PCA on the features extracted from RLRN method. Kernel PCA removes redundant features and thus the detection performance is increased. Their experimental results show an improvement of approximately 29.5% in the detection accuracy compared to original work (88.74% vs. 68.50%) respectively.

2.9.5 Co-occurrence Matrix

Co-occurrence matrix is another feature extraction method applied in detecting spliced images. This matrix is defined as a distribution of co-occurring values at a given offset over an image. The following equations show how the second-order matrix is calculated (Albregtsen, 2008; Haralick & Shanmugam, 1973):

$$C_{\Delta x,\Delta y}(i,j) = \sum_{p=1}^{n}\sum_{q=1}^{m}\begin{cases}1, & if\ I(p\ ,q) = i\ and\ I(p+\ \Delta x, q+\ \Delta y) = j\\ 0, & otherwise\end{cases} \qquad \textbf{(2.53)}$$

where n and m are the dimensions of image I, i and j are the image intensity values, p and q are the image spatial positions and the offset $(\Delta x, \Delta y)$ depends on the direction (θ) used and the distance (d) at which the matrix is computed.

Grey Level Co-occurrence Matrix (GLCM) is a kind of co-occurrence matrix applied in image splicing detection. Since the splicing procedure modifies image edges by making them sharper, one of the clues which could be used in image splicing detection is studying the image edges. GLCM serves the image edges to extract second-order texture information from the spliced images.

An example of splicing detection methods using GLCM can be seen in (Wei Wang et al., 2009). They applied GLCM to edge images along four directions to extract the features from the image chroma component. According to their method, the edges introduced by Cb and Cr components are sharper than those in Y component. Thus they applied the predict-error method to reduce the diversity in image contents and obtain the edge image. Then GLCM was applied on edge images to extract the features. Finally, the features were reduced using boosting feature selection. Their results show acceptable detection accuracy. Similar to RLRN method, GLCM is only applicable for colored images and its results in chroma channels are more than 88%.

2.9.6 Phase Congruency

Phase Congruency is a sensitive measure that captures sharp transitions including lines, edges, and corners that are left behind in the spliced images. It detects the edges based on the changes in illumination and contrast. A wide range of feature types could be captured by applying phase congruency.

The model presented by Morrone et al. named the Local Energy Model. (Morrone et al., 1986) explains that sharp characteristics are considered at the points of highest phase congruency in a picture. The phase congruency or PC was first described by Morrone and Owens (Morrone & Burr, 1988) in terms of expansion for Fourier series of a single signal at location *x*:

$$PC_1(x) = max_{\overline{\emptyset}(x)\in[0,2\pi]} \frac{\sum_n A_n \cos\left(\emptyset_n(x) - \overline{\emptyset}(x)\right)}{\sum_n A_n} \quad \textbf{(2.54)}$$

where A_n is considered as the amplitude of *nth* Fourier component and $\emptyset(x)$ is assumed as the local phase for the *nth* Fourier component in position *x*, $\overline{\emptyset}(x)$ is in fact amplitude weighted mean local phase angle for position *x*. If PC equals the maximal value of 1, all of the frequency components are considered in phase or PC will have values from 0 to 1.

Kovesi (Kovesi, 1999) extended the 1-D phase congruency equation to 2-D, one that is able to be applied to image processing. 2-D phase congruency can be calculated by applying a 1-D equation over several orientations and combining the results in some way. In fact, Kovesi solved the calculation problems of phase congruency according to Equation (2.4), and made a new and more sensitive measure from it.

Due to the properties of phase congruency in detecting sharp transitions Chen, W. et al. (W. Chen et al., 2007) proposed a phase congruency-based image splicing detection method. They merged the phase congruency-based features with the statistical moments of characteristic functions of four wavelet sub-bands in three levels. However, the result obtained from their proposed method shows only a comparable accuracy of 82.32%.

Phase congruency as a feature extraction method can be applied in image splicing detection, since it reflects the inconsistencies caused by effective splicing process.

However, calculating the phase congruency map of an image is computationally intensive and thus this method has not commonly been applied in this field.

2.9.7 Three order methods

Since image splicing presents sharp edges on the tampered image, finding these artifacts is a key detection task. While second order statistical features have been confirmed as effective characteristics, statistics with higher orders have been suggested to find the splicing artifacts in (Tian-Tsong Ng et al., 2004b) as well as (W. Chen et al., 2007) , but the rate of detection is unsatisfactory. Also Zhao, X. et al. (Zhao et al., 2012) demonstrated the relationships between three neighboring factors in block DCT domain as third order features statistically, which consists of second order Markov transition probability matrix (2^{nd} Markov), conditional co-occurrence probability matrix (CCPM), as well as second order co-occurrence probability matrix (2^{nd} CPM).

The third order characteristics are highly informative compared to lower order ones, but feature dimensionality is exponentially dependent on the order. Based on their investigations, the CCPM outperforms 2^{nd} Markov, 2^{nd} CPM, 1^{st} Markov as well as first order co-occurrence matrix of probability (1^{st} CPM) characteristics.

Markov chain is usually employed to present the existing dependencies among neighboring states. The first order features of Markov were confirmed to be the most influential characteristics for image splicing detection (Shi et al., 2007b). In the first order features of Markov, the dependency is limited in two different neighboring states. In addition, Zhao, X. et al. (Zhao et al., 2012) enhanced the first order characteristics of Markov to third order statistical characteristics through assuming the relationship between three neighboring states .

$$CCPM \equiv \begin{bmatrix} P(\omega_1,\omega_1|\omega_1) & P(\omega_2,\omega_1|\omega_1) & \dots & P(\omega_1,\omega_1|\omega_1) \\ \vdots & \vdots & & \vdots \\ P(\omega_1,\omega_1|\omega_N) & P(\omega_2,\omega_1|\omega_N) & \dots & P(\omega_N,\omega_N|\omega_N) \end{bmatrix}$$

$$2^{nd}Markov \equiv \begin{bmatrix} P(\omega_1|\omega_1,\omega_1) & P(\omega_1|\omega_1,\omega_2) & \dots & P(\omega_1|\omega_N,\omega_N) \\ \vdots & \vdots & & \vdots \\ P(\omega_N|\omega_1,\omega_1) & P(\omega_N|\omega_1,\omega_2) & \dots & P(\omega_N|\omega_N,\omega_N) \end{bmatrix} \quad \textbf{(2.55)}$$

$$2^{nd}CPM \equiv \begin{bmatrix} P(\omega_1,\omega_1,\omega_1) & P(\omega_1,\omega_1,\omega_2) & \dots & P(\omega_1,\omega_N,\omega_N) \\ \vdots & \vdots & & \vdots \\ P(\omega_N,\omega_1,\omega_1) & P(\omega_N,\omega_1,\omega_2) & \dots & P(\omega_N,\omega_N,\omega_N) \end{bmatrix}$$

Here, $P(\omega_{ik},\omega_{ik-1}|\omega_{ik-2})$ is the co-occurrence probability for $(\omega_{ik},\omega_{ik-1})$ of state ω_{ik-2}, $P(\omega_{ik}|\omega_{ik-1},\omega_{ik-2})$ demonstrates the second order transition probability of Markov, feature state is defined via previous and current state, $P(\omega_{ik},\omega_{ik-1},\omega_{ik-2})$ is in fact the joint probability of these three states.

The third order feature matrices statistically are all-directional, which means they could be utilized to show the relationship among three adjacent phases in eight different directions. In their research, CCPM, 2nd Markov, as well as 2nd CPM are used for modeling the vertical down and horizontal right directional relations of three neighboring states. In their findings presented for third order statistical features, CCPM (88.8%) will outperform 2nd CPM (85.5%) and Markov (86.8%).

However, the dimensionality of features exponentially depends on order (686-D); for modern supervised machine learning algorithms, high-dimensionality usually causes computational complexity and over-fitting (Bengio et al., 2005). Therefore, PCA is employed in their work to reduce the higher order introduced problems.

2.9.8 Local Binary Pattern (LBP)

The Local Binary Pattern was utilized by Ojala, Pietikäinen, & Harwood (1996) for classification of texture since its invariance to universal intensity variations. The original operator of LBP (G. Zhang et al., 2004), explains a LBP code of each single pixel of a picture. To calculate LBP code, the 3×3 pixels neighboring is in fact thresholded through its value of intensity. When the pixel value of neighbors is lower than center, it would have binary digit equal to 0, otherwise it is 1. The binary digits of the neighbors are combined to create a binary code. The LBP code in fact is a decimal value of that specific binary code. Figure 2.20 demonstrates the LBP code calculation process. Then, the neighborhood LBP operator size is expanded (Jabid et al., 2010).

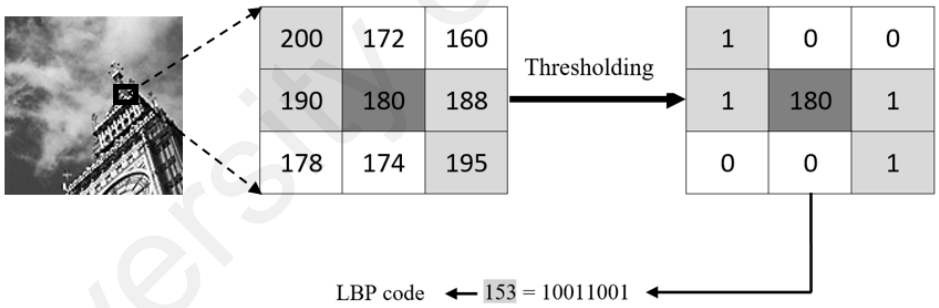


Figure 2.20: LBP Computation Process

The LBP operator can also be denoted via $LBP_{P,R}$ and is calculated as below (Alahmadi et al., 2013):

$$LBP_{P,R} = \sum_{i=1}^{p-1} S(p_i - p_c)2^i \quad \textbf{(2.56)}$$

while P is the number of neighborhood pixels and R would be its radius, p_c would be the value of the center pixel and so the thresholding equation is known as below (Alahmadi et al., 2013):

$$S(p_i - p_c) = \begin{cases} 0, & p_i - p_c < 0 \\ 1, & p_i - p_c \geq 0 \end{cases} \quad \textbf{(2.57)}$$

When the tampering is performed, the original image texture will be distorted. Since the LBP has the capability to show differences among textures, it is an effective tool to identify forgeries (Alahmadi et al., 2013).

Several LBP variants have been utilized. Zhang et al, (Y. Zhang et al., 2012) developed a splicing detection technique according to LBP operator using multi-size block DCT achieved from images. The coefficients of BDCT can demonstrate such modifications to a specific degree. The LBP method essence is that each single factor of a specific 2-D array is compared with neighboring factors and later binarized. Thus, the LBP records all of the occurrences of different patterns. LBP could be used in order to model whole magnitude components of 2-D arrays which are achieved through using MBDCT to testing images. It will consider that LBP operator is able to demonstrate local frequency distribution modification of host images appropriately. Their experimental outcomes demonstrate a detection accuracy of almost 89.93% for $LBP_{8,1}$.

Muhammad. G et al. (Muhammad et al., 2014) developed an image forgery detection approach based on SPT, steerable pyramid transform, and also LBP. Firstly, they converted a colored image to YCbCr color space and used SPT transform on the chrominance channels Cr and Cb, resulting in many multi-scale and also multi-oriented sub-bands. Later, features are extracted for each single SPT sub-band utilizing LBP

algorithms. The histograms from these sub-bands are then combined to generate feature vectors. Since their method is high-dimensional, they used a feature selection method to reduce the dimensionality of their method. Their experimental results show detection accuracy rates of 94.89%, 97.33% and 96.39% on chrominance channels. However, the detection rate for greyscale images is very low, at only 64.65%.

Hakimi. F et al. (Hakimi et al., 2015) enhanced the technique developed in (Y. Zhang et al., 2012) through using modified LBP rather than usual LBP, and also applied k-nearest neighbor algorithm (KNN) as a classifier. Such findings revealed detection accuracy rates of 96% and 98% on color images.

Generally, a summarized overview of the image splicing detection methods discussed is presented in Tables 2.1-2.4. The methods are reviewed and discussed from different aspects, including the nature of feature extraction methods, detection accuracy, and dimensionality.

2.9.9 Fractional Differential Texture Descriptors

Fractional calculus is widely applied in physical and engineering sciences. Fractional differentiation is also excellent in describing the general properties of various materials and processes. The fractional approach preserves low-frequency features in smooth areas and enhances texture details in areas where the grey level does not clearly change (Jalab & Ibrahim, 2012, 2013, 2015). Texture features represent high-level information that can be used to describe the objects and structure of images.

Rabha, W. I. et al. (Ibrahim et al., 2015) developed a new fractional differential approach for texture feature descriptors by focusing on the types of texture parameters used for image splicing detection. In their study, first they increased the quality of the

images by applying Machado entropy for texture enhancement as follows (Ibrahim et al., 2015; Machado, 2014):

$$S_\alpha(P) = \sum \frac{-P_i^{-\alpha}}{\Gamma(\alpha+1)}[\ln(P_i) + \psi(1) - \psi(1-\alpha)]P_i \qquad \textbf{(2.58)}$$

where P_i is the probability of occurrence, and $\Gamma(.)$ and $\psi(.)$ refer to the gamma and digamma functions, respectively. Accordingly, P_i is the probability distribution of the image pixel's intensity.

Then, they built a generalized fractional mask (Φ) by using the following generalized fractional differential operator (Ibrahim, 2011; Ibrahim et al., 2015):

$$D^{\alpha,\mu}h(x) := \frac{(\mu+1)^\alpha}{\Gamma(1-\alpha)}\frac{d}{dx}\int_0^x \frac{\zeta^\mu h(\zeta)}{(x^{\mu+1}-\zeta^{\mu+1})^\alpha}d\zeta; 0 < \alpha \leq 1 \qquad \textbf{(2.59)}$$

where *h(x)* is an analytic function. Finally, the value of the fractional differential operator (2.26) is computed through a numerical calculation that references the discrete form. Thus, the non-zero fractional differential coefficients ($\emptyset_i$) are obtained (Ibrahim et al., 2015) as:

$$\begin{aligned} \emptyset_0 &= \frac{(\mu+1)^\alpha}{(2^{\mu+1}-1)^2\Gamma(1-\alpha)(1-(\mu+1)\alpha)} \\ \emptyset_1 &= \emptyset_0(2^{1-(\mu+1)\alpha}-1) \\ &\vdots \\ \emptyset_{n-1} &= \emptyset_0[(n+1)^{1-(\mu+1)\alpha} - n^{1-(\mu+1)\alpha}] \end{aligned} \qquad \textbf{(2.60)}$$

By convoluting $\emptyset_i$ with $S_\alpha(P_i)$, the following equation is achieved:

$$\Phi_1 = \emptyset_1 \times S_\alpha(P_1), \ldots, \Phi_{n-1} = \emptyset_{n-1} \times S_\alpha(P_{n-1}) \quad \textbf{(2.61)}$$

The 2-D fractional mask coefficients of all images can be obtained in eight directions including 0°, 45°, 90°, 135°, 180°, 225°, 270°, and 315°. This algorithm primarily aims to split the image into non-overlapping blocks and to apply Equation (2.61) with optimally different values of α and μ to extract the texture features. After texture enhancement utilizing fractional calculus, the GLCM is applied to extract the texture features from each image block, after using fractional texture enhancement based on fractional differential masks.

Generally, the features extracted from this method have high-dimensionality (1764-D), although a feature reduction method (i.e. Kernel PCA) can be applied to decrease the dimensionality of the extracted features. Their results show a best detection accuracy of 91.88% with only 40 dimensions. Although their results are not high enough, the method they proposed show how various mathematical concepts could be promisingly applied to develop a new image splicing detection scheme.

2.10 Summary

Throughout the chapter, the most recent image splicing detection schemes have been reviewed and discussed. The methods have been investigated based on various aspects such as feature extraction scheme, detection rate, computational complexity, and dimensionality. In addition, different concepts regarding image forgery detection have been presented to clarify various parts of the research. Tables 2.1-2.4 show a summarized view of the image splicing detection methods discussed, based on their detection rate, dimensionality, some advantages and drawbacks.

Table 2.1: Comparison between the reviewed image splicing detection methods

Method	Detection Rate	Dimensionality	Dimension Reduction Method	Comment
Fu et al. (Fu et al., 2006) Moment + HHT-based Features	85.87% (DVMM v1)	72	N/A	N/A
Shi et al. (Shi et al., 2007b) Moment-based + Markov-based features	91.8% (DVMM v1)	266	N/A	Dimensionality is high and time-consuming.
Chen, W. et al. (W. Chen et al., 2007) Moment-based + 2-D phase congruency	82.32% (DVMM v1)	120	N/A	N/A
Zhang et al. (Z. Zhang, Zhou, et al., 2008) Image quality metrics + moment features	88.8% (DVMM v1)	196	N/A	N/A
Dong et al. (Dong et al., 2009) Run-Length and Edge Statistics Based Approach	84.36% (DVMM v1)	163	N/A	N/A

Table 2.2: Continued, comparison between the reviewed image splicing detection methods

Method	Detection Rate	Dimensionality	Dimension Reduction Method	Comment
Wang et al. (Wei Wang et al., 2009) Gray Level Co-occurrence Matrix (GLCM)	90.5% (Custom)	400	boosting feature selection (BFS) 50-D	Only for colored images.
Zhang et al. (J. Zhang et al., 2009) Markov + DCT features	91.5% (DVMM v1)	109	N/A	N/A
Anusudha et al. (Anusudha et al., 2010) First-order moment-based + phase congruency	91.7% (DVMM v1)	198	N/A	N/A
Sutthiwan et al. (Sutthiwan et al., 2010) Moment-based + Markov-based features	98% (CASIA v1)	266	boosting feature selection (BFS)	Accuracy rate is very high. However, the dimensionality is also high.
He et al. (He et al., 2011) Approximate run length + DWT	80.58% (DVMM v1)	30	N/A	N/A

Table 2.3: Continued, comparison between the reviewed image splicing detection methods

Method	Detection Rate	Dimensionality	Dimension Reduction Method	Comment
Zhao et al. (Zhao et al., 2011) Run length Run number	94.7% (CASIA v1) 85% (DVMM v2)	60	N/A	An efficient tool for colored images not gray scale ones.
He et al. (He et al., 2012) DCT Markov approach + DWT Markov approach	93.55% (DVMM v1)	7290	SVM recursive feature elimination (SVM-RFE) 100-D	It is an effective tool for image splicing detection. However its dimensionality is very high.
Zhao et al. (Zhao et al., 2012) -conditional co-occurrence probability matrix (CCPM) -second order Markov transition probability matrix (2nd Markov) -second order co-occurrence probability matrix (2nd CPM)	88.8% (DVMM v1) 86.8% (DVMM v1) 85.5% (DVMM v1)	686	PCA 100-D	Dimensionality of features exponentially increases by increasing the order. The algorithms are high dimensional.

Table 2.4: Continued, comparison between the reviewed image splicing detection methods

Method	Detection Rate	Dimensionality	Dimension Reduction Method	Comment
Zhang et al. (Y. Zhang et al., 2012) LBP + MBDCT	89.93% (DVMM v1)	945	PCA	N/A
Muhammad. G et al. (Muhammad et al., 2014) LBP + SPT	94.89% (CASIA v1) 97.33% (CASIA v2) 96.39% (DVMM v2)	3584	Feature Selection	It is only for colored images and also the method is very high dimensional.
Moghaddasi. Z et al. (Moghaddasi, Jalab, Md Noor, et al., 2014) Improved RLRN	88. 28% (DVMM1) 88.74% (CASIA)	50	Kernel PCA	N/A
Hakimi. F et al. (Hakimi et al., 2015) Improved LBP	97.21% (CASIA v1) 95.13% (DVMM v2)	N/A	PCA	N/A
Ibrahim et al. (Ibrahim et al., 2015) Fractional Differential Texture Descriptors	91.88% (DVMM1)	40	Kernel PCA	It is in high dimension and only done on gray-scale images.

CHAPTER 3: RESEARCH METHODOLOGY

3.1 Introduction

The methodology defines what should be done during the life-cycle of system development, and also how such a system can be managed. In addition, it shows each phase of the system to be developed during its life-cycle as well as tasks which should be accomplished in each single phase by a specific time. The methodology for system development guides systems into the goals and objectives, and during development, sets it in an appropriate direction.

There are various methodologies to develop a system and each of them have their own tasks and phases, as well as characteristics. In the current chapter, the methodology will describe the development of the proposed image splicing detection systems, their core phases and those tasks which should be accomplished in each phase. Phase one explores the requirements and their analysis, and the other two phases include designing and implementing the two proposed methods. Each of these methods have been implemented in order to improve the detection rate.

3.2 Research Phases

The applied methodology for this research is according to the Waterfall model which contains several fundamental phases to develop a system. This sort of software development model has many phases, for example requirement collection and analysis, design and implementation, and also test and evaluation (Parekh, 2016). All of the activities which should be accomplished are briefly mentioned as follows:

i. Requirement Collection and Analysis

In this stage all of the necessary requirements to develop a system are collected. Such requirements consist of setting the goals and objectives, as well as the scope of this system. It also includes defining the development schedule, specifying the needed terms and gathering whatever the system requires to develop.

ii. System Design

Before the system is implemented, it should be clarified what is going to be created and what it looks like. Hence, in this level, the steps and the first-level design of the proposed methods will be presented.

iii. Implementation

At this level, the system is constructed and the proposed algorithms in the design phase will be translated to programming code or could be simulated. Then the system will be created

iv. Test and Evaluation

After constructing the system, it should be tested and evaluated in order to make sure it can provide the essential requirements. In this research, the output of system is examined through many test cases.

Thus, referring to the methodology discussed, this research also covers collecting the requirements and their analysis, design and construction, test and evaluation. Figure 3.1 illustrates the key processes to be performed in development of the proposed image splicing detection methods.

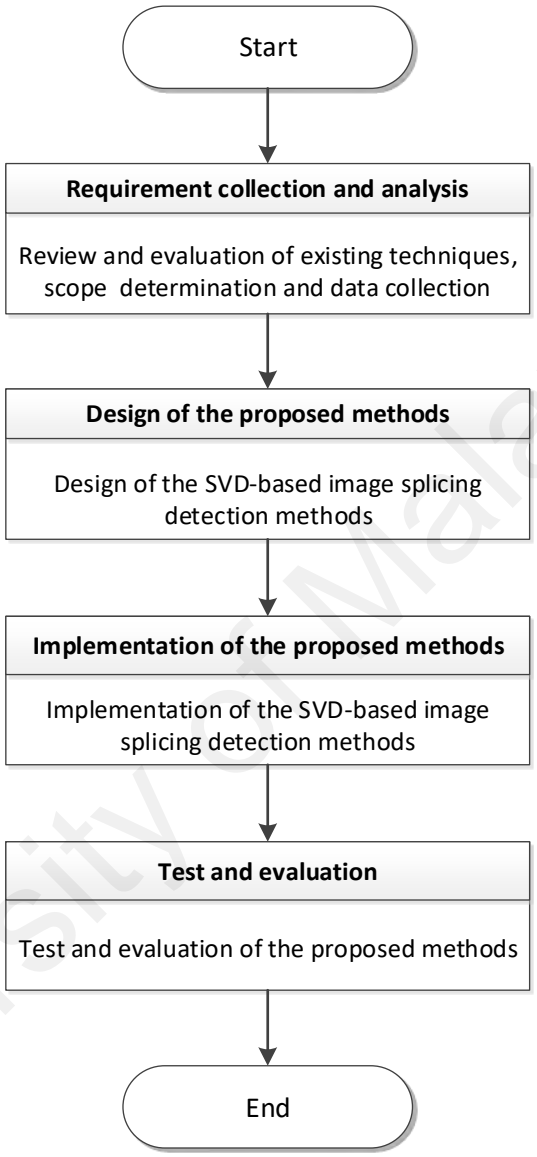


Figure 3.1: Operational Framework

3.3 Requirement Collection and Analysis

The first phase to establish splicing detection schemes is to assess previous methods, analyze them all in order to identify a gap and so define the system requirements. In addition, this phase contains many activities for example:

i. Assessment and study of existing methods

ii. Definition of system scope

iii. Assumptions and hypothesis

3.3.1 Assessment and study of existing methods

To identify the problem statement and objectives of the research, there is a need to evaluate and analyze the existing splicing detection methods. A comprehensive literature review is focused on discovering the problems of past splicing detection schemes. This literature review, presented in the second chapter, discusses the advantages and drawbacks of different existing splicing detection techniques (Objective no. 1). Analyzing such literature leads to proposing two SVD-based splicing detection approaches.

3.3.2 Scope of System

According to our problem statement mentioned in the first chapter and by evaluating the existing methods, the scope of the proposed systems is defined as follows:

3.3.2.1 Software

The software used for constructing the proposed methods is MATLAB R2016a which stands for Matrix Laboratory. This product can be applied in different areas such as math and computation, algorithm development, modeling simulation and prototyping, scientific and engineering graphics. It also provides an easy-to-use environment for programming, in addition to its various prepared components used in the mentioned areas

(The MathWorks, 2012). Because of these advantages MATLAB can be an appropriate choice in implementing our algorithm.

Since MATLAB R2016a can be run on Windows 10 operating systems, the selected operating system for implementing the proposed algorithm is Windows 10.

3.3.2.2 Hardware

The hardware used for implementing this research should be able to support any version of MATLAB. Our hardware consists of:

- CPU: 2.40 GHz Intel (R) Core™ i3
- HD: SATA 500 GB
- RAM: 4.00 GB

3.3.2.3 Data

Three standard image datasets (grey and color) have been applied to evaluate the proposed methods. The first image dataset was the Columbia Image Splicing Detection Evaluation Dataset provided by the Digital Video MultiMedia (DVMM) Laboratory, Columbia University (2007) (Tian-Tsong Ng et al., 2004c). This dataset contains 1845 greyscale images (933 authentic images and 912 spliced images) in BMP format. DVMM v1 was the only greyscale image dataset designed for image splicing detection evaluation. Almost all splicing detection methods were applied to DVMM v1, and so it was also used in this research for better comparison with other methods.

The second image dataset was designed by the Chinese Academy of Sciences, Institute of Automation (CASIA), with high resolution images. The CASIA tampered image

detection evaluation database (Dong & Wang, 2011) is another image dataset designed to evaluate image splicing detection methods. Version 1.0 of this dataset includes 1721 color images (800 authentic images and 921 spliced images) of 384 × 256 pixels in JPEG format, and was used in our experiments to evaluate the proposed approaches.

The second version of DVMM is the third image dataset applied in this research. DVMM v2 (Tian-Tsong Ng et al., 2009) contains 363 color images (183 authentic images and 180 spliced images). The authentic images were taken by four different cameras. All the images are in high resolution in uncompressed RAW or TIFF formats with dimensions ranging from 757 × 568 to 1152 × 768.

Examples of the three image datasets are presented in Figure 3.2 (DVMM v1), Figure 3.3 (CASIA) and Figure 3.4 (DVMM v2) in which the first row is for authentic images and the second row includes the spliced images (Dong & Wang, 2011; Tian-Tsong Ng et al., 2004c; Tian-Tsong Ng et al., 2009).

Figure 3.2: Example of DVMM v1 Image Dataset

Figure 3.3: Example of CASIA Image Dataset

Figure 3.4: Example of DVMM v2 Image Dataset

3.3.3 Assumptions and Hypothesis

SVD is an effective feature extraction method widely applied in different image processing applications. In this study, it is considered that the application of SVD as feature extraction method in image splicing detection method would be likely to distinguish between the authentic images and spliced ones with an acceptably high accuracy rate. Consequently, this hypothesis was investigated and two SVD-based image splicing methods were proposed and developed (see Chapter 4).

3.4 Structures of the proposed methods

In this section, the overall design of the proposed methods is explained briefly. As mentioned previously, this research proposes two SVD-based image splicing detection methods, one with Logarithm (Log) and the other one with condition number (CN). The general design framework of image splicing detection schemes comprises image preprocessing, feature extraction, feature selection and classification. This general structure of the proposed methods is presented in this section. Figure 3.5 demonstrates the general structure, the activities that are accomplished and the objectives that are achieved in each phase.

In the preprocessing step, the given image is prepared for feature extraction. This preparation includes image multi-blocking, greyscale conversion, channel separation (RGB or YCbCr), and frequency domain transformation (DCT). The preparation phase is performed to specify which image attributes best reflect the splicing operation in a spliced image.

In the second phase, features are extracted from the prepared images and it is essential to find a proper method to extract the best features from those images. In this research, the concept of SVD is being used to propose feature extraction methods for the spliced images. The design of these two feature extraction techniques is presented in Figure 3.6.

When the features are extracted from the images, they should be prepared for classification to distinguish the authentic images from the spliced ones. The extracted features might be in high dimension and also contain some redundancies that could be removed to improve the detection rate. Therefore, it is essential to determine the nature of the features (linear and non-linear) and select an appropriate feature selector to prepare

the extracted features for classification. Subsequently, this step is performed by the application of PCA and Kernel PCA as the feature selectors.

To identify the authentic images and the spliced ones, there is a need to classify the prepared features, which makes the last step in every image splicing detection method. This step is also of major importance, since it separates original and forged images. Therefore, a suitable classifier (SVM) should be selected. All these steps are performed in sequence to construct the image splicing detection schemes.

The proposed SVD-based image splicing detection approaches are same in their general design process but differ in the core calculation (Log and CN), which is described in next chapter. Figure 3.6 illustrates the first-level design of the proposed methods. First the given image in Grey, R, G, B, Y, Cb, or Cr channel is divided into $n \times n$ blocks in which $n = 3, 4, ..., 27$. Then the SVD-based features are extracted from the blocks in spatial or frequency domains. Subsequently, non-linear combinations of data are generated applying 1^{st}-4^{th} order statistics, i.e. mean, variance, skew, and kurtosis to better present the distribution of the extracted values. Finally, SVM is used to classify the authentic and spliced images and calculate the average detection rate. All these steps are explained in detail in next chapter.

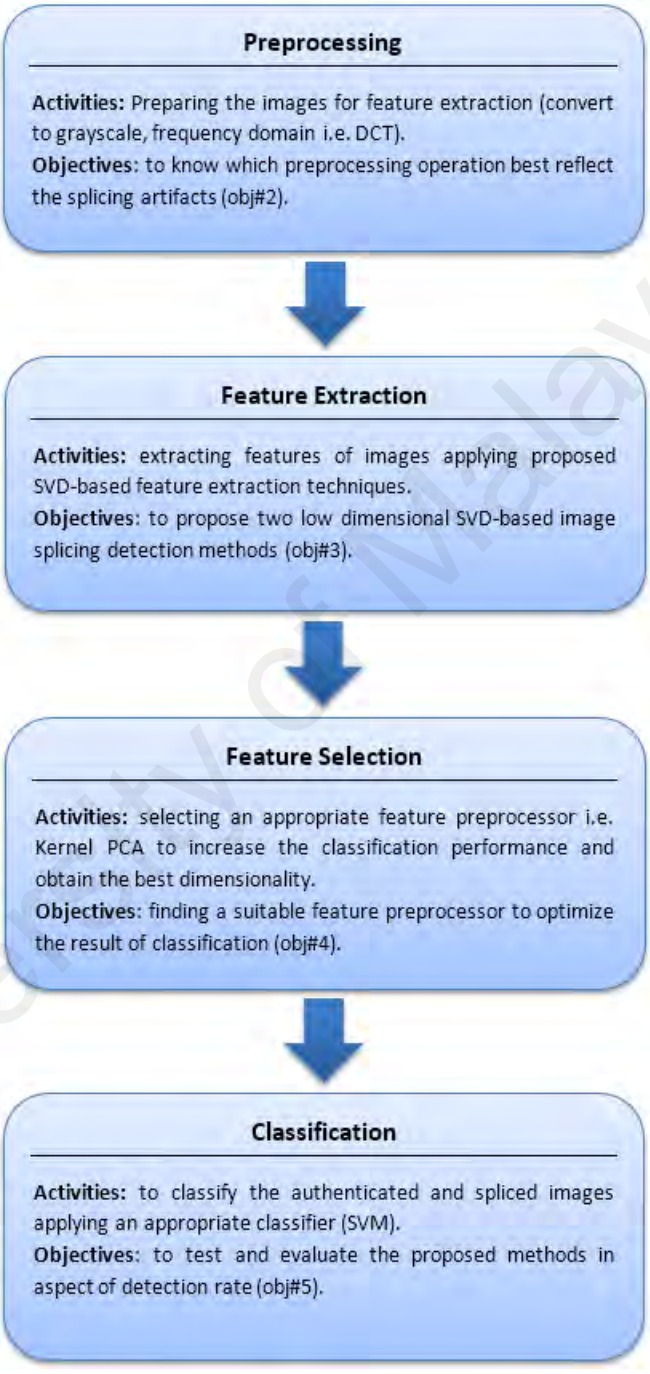


Figure 3.5: General Steps of The Proposed Methods

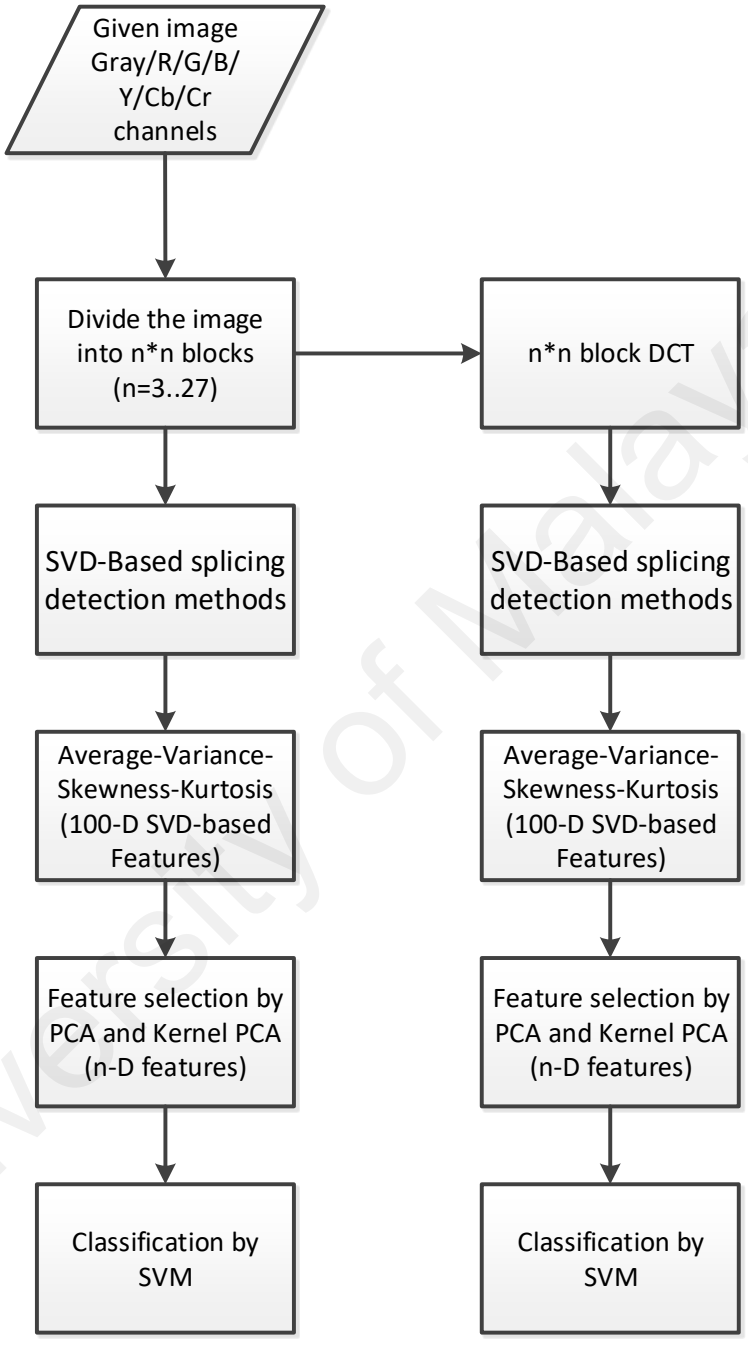


Figure 3.6: First Level Design of the Proposed Methods in Spatial (a) and Frequency (b) domains

3.5 Summary

The methodology applied in this research for the proposed splicing detection methods has been explained throughout this chapter. Then the general structure of the proposed methods with their first-level design has been presented and described. Generally, the phases followed in this research are: requirement collection and analysis, system design, implementation, test and evaluation. However, the details of the proposed methods are explained in detail in the design and implementation chapter (Chapter 4).

CHAPTER 4: RESEARCH DESIGN AND IMPLEMENTATION

4.1 Introduction

In this chapter, the detailed design and implementation of the proposed image splicing detection methods are elaborated. Both of these proposed methods are based on SVD (SVD-Log and SVD-CN), which is able to detect authentic images from spliced ones with a high detection rate and reasonable computational complexity. The main goal of the proposed schemes is to apply SVD in feature extraction for detecting spliced images leading to low-dimensionality, high detection accuracy within a good enough time.

This chapter is organized as follows. First, the image preprocessing phase is presented and the image is prepared for feature extraction. Then the feature extraction for both proposed methods is presented and elaborated. Consequently, section 4.4 discusses the selection of the appropriate feature selector to prepare the features for classification, and finally the selected features are classified to obtain the detection rate in section 4.5.

4.2 Image Preprocessing

As demonstrated in overall design flow in chapter 3, the given image from three datasets is preprocessed first. Preprocessing improves image content by reducing undesired distortions and/or enhancing image features relevant for further processing. In this study, some common preprocessing operations performed include conversion between color channels, frequency domains transformation, and image blocking – all of which are presented in this section.

4.2.1 Color Spaces

According to the image formation described in second chapter (Section 2.2), each color image is presented based on a color model (usually 3D or 4D). The image splicing operation causes some inconsistencies in the statistical features of an image, such as abnormally sharp transience at the splicing edges. These inconsistencies are reflected in each color model or color channel differently. However, some splicing information might not be clear in one of the color models or channels. Therefore, selection of an appropriate color model or color channel is of great importance in image splicing detection methods.

Several color models (e.g., RGB, HSV, YCbCr, CMY, YIQ, YUV, CIEL $*$ a $*$ b $*$, XYZ, etc.) are now available for different applications including feature extraction, object recognition, tracking, etc (Stokman & Gevers, 2007). RGB is the most common color model applied in image splicing detection methods. The components of the RGB color model have high correlation with each other. To reduce the correlation between these three color components, the RGB color model can be converted to another Luminance-Chrominance (L-C) color space including CIELAB, YUV, YCbCr, and so on. Since another color model, YCbCr, is commonly applied in image splicing detection approaches, it is also selected as one of the color models applied in this research. Figure 4.1 shows the relationship between RGB color space and YCbCr color model (Curtis, 1994).

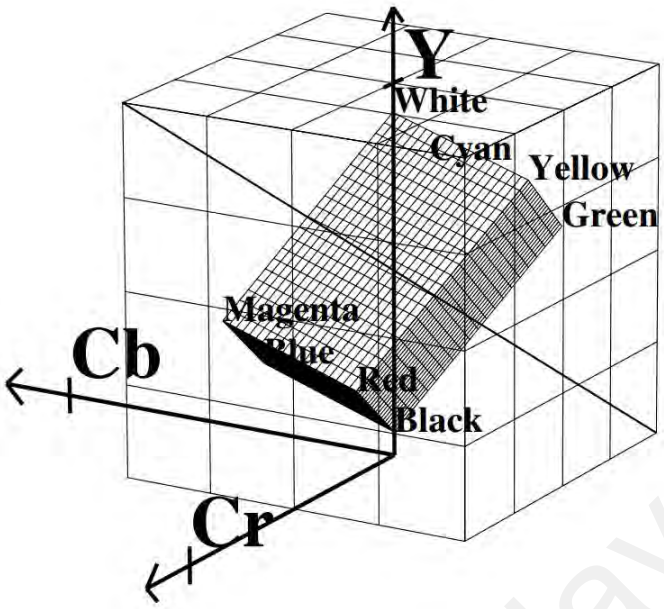


Figure 4.1: RGB and YCbCr color spaces

YCbCr values can be obtained by linear transform from R, G, and B channels which is formulated as follows (Mitra & Acharya, 2005):

$$\begin{pmatrix} Y \\ Cb \\ Cr \end{pmatrix} = \begin{pmatrix} 0.29900 & 0.58700 & 0.11400 \\ -0.16874 & -0.33126 & 0.50000 \\ 0.50000 & -0.41869 & -0.08131 \end{pmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix} \quad \textbf{(4.1)}$$

Y is the linear combination of R, G, and B channels; Cb and Cr channels are the blue-difference and red difference chroma components, respectively. Y channel keeps most of the edges compared to Cb and Cr channels. From Equation (6.1) it can be observed that G and B channels have the maximum and minimum weights in Y channel (with the coefficients 0.587 and 0.114 respectively). Thus, the edges in green areas are more likely to be bolder than the edges in red and blue portions. This remark is illustrated in an example presented in Figures 4.2 and 4.3. It is observed from Figures 4.2 and 4.3 that the edges in green areas are more visible in the Y channel than those in red and blue. It also can be noticed from Figure 4.3 that some edges which are not displayed properly in Y

channels are visible in Cb and Cr channels. Another point in these Figures is that some edges might not be clear in a single channel i.e. the edges caused by splicing are invisible in one channel but clear in another one. Therefore, it is essential to consider all the channels in image splicing detection methods.

Since each color channel preserves some specific information in each channel, R, G, B, Y, Cb, and Cr channels are tested in our proposed SVD-based image splicing detection approaches. Furthermore, greyscale image is also the most basic and simple image which only shows one value for each pixel. Since some of the image datasets are in greyscale and almost all the image splicing detection algorithms apply the images in greyscale, some of this study's experiments were also designed for application to greyscale images.

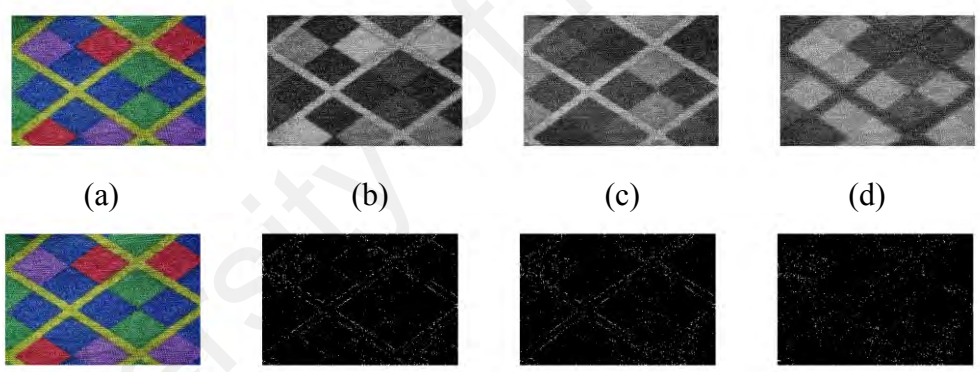


Figure 4.2: Image in RGB and its edges in different channels, (a) RGB Image, (b) Red channel, (c) Green channel, (d) Blue channel, (e) RGB Image, (f) edges in Red channel, (g) edges in Green channel, (h) edges in Blue channel

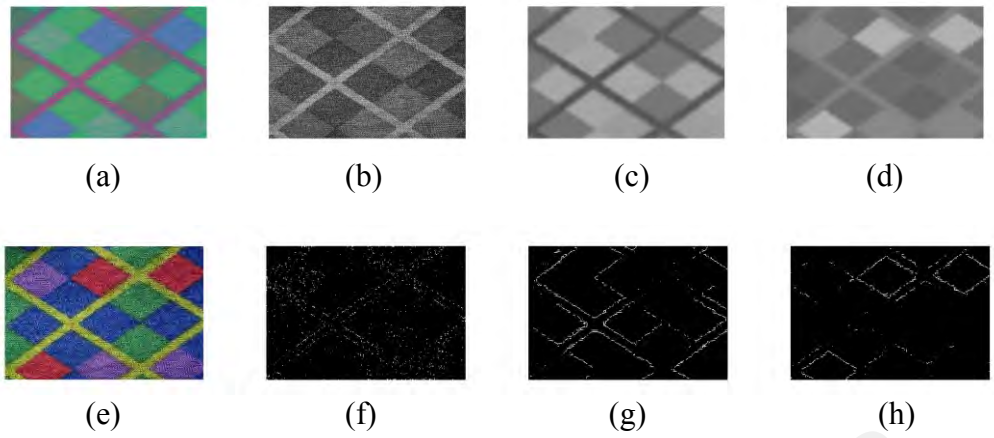


Figure 4.3: Image in YCbCr and its edges in different channels, (a) YCbCr Image, (b) Y channel, (c) Cb channel, (d) Cr channel, (e) YCbCr Image, (f) edges in Y channel, (g) edges in Cb channel, (h) edges in Cr channel

4.2.2 Multi-Size Blocking

Splicing operations modify the correlation between the pixels in the spliced images applying various patterns that make capturing these changes much more complicated. On the other side, digital images have strong dependency between their pixels in the spatial domain and the features captured from sub-blocks represent locality in the spatial domain rather than the whole image. Therefore, it is not effective to capture the features of images using one single-block-size approach. With various block sizes, the changes in spatial or frequency distribution of the images can be captured in different ways, which leads to better detection accuracy.

The idea of applying a multi-block-size approach was previously utilized in (Shi et al., 2007b). This idea is also used in this research to capture the features from different areas of the given images. By increasing the size of blocks, the number of blocks to capture the features is decreased. To select an appropriate set of block sizes, a balance between detection accuracy, feature extraction time and number of dimensions has been considered. With small block sizes, the elements of SVD cannot capture enough artifacts

caused by splicing. On the other hand, when the block size increases, the dimensionality also goes up, which does not improve detection performance. According to the experimental results presented in chapter 5, the suitable block size set for our features is [3, 4, 5, …, 27]. Figure 4.4 indicates a multi-block size diagram with *N* size (Shi et al., 2007b).

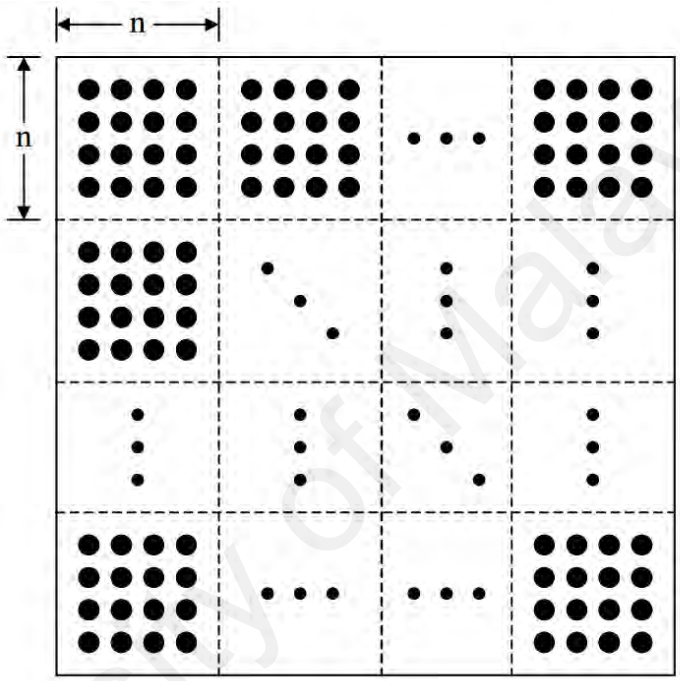


Figure 4.4: Mulit-Block diagram with $N \times N$ Block sizes, $N \in [3 \ldots 27]$

4.2.3 Frequency Domain

In the last few decades, transform coding has become one of the most essential parts of signal processing applications. It uses the assumption that adjacent pixels (frames) in image (video) have a certain level of correlation. A transformation is mapping the correlated data called spatial data into uncorrelated coefficients referred as transformation (Khayam, 2003). DCT is the most popular block-based transform. It de-correlates the

image data, then each transformed coefficient can be encoded freely without missing important information (Moghaddasi, Jalab, & Noor, 2014; Moghaddasi et al., 2015).

The splicing process modifies the local frequency distribution of the host images. Applying multi-block sizing DCT can reflect the changes made by the splicing process. Therefore, DCT is used in our methods to make use of its benefits in splicing detection schema. DCT is calculated for every sub-block in size n and then the features of every DCT sub-block are captured.

Figure 4.5 indicates the preprocessing phase applied in this research. The given image from the image datasets is in greyscale or color. The color image is converted to greyscale, R, G, B, Y, Cb, Cr channels. Every channel is divided into $n \times n$ blocks using multi-block size approach, and every block is converted to spatial and DCT. Then the image is ready for feature extraction.

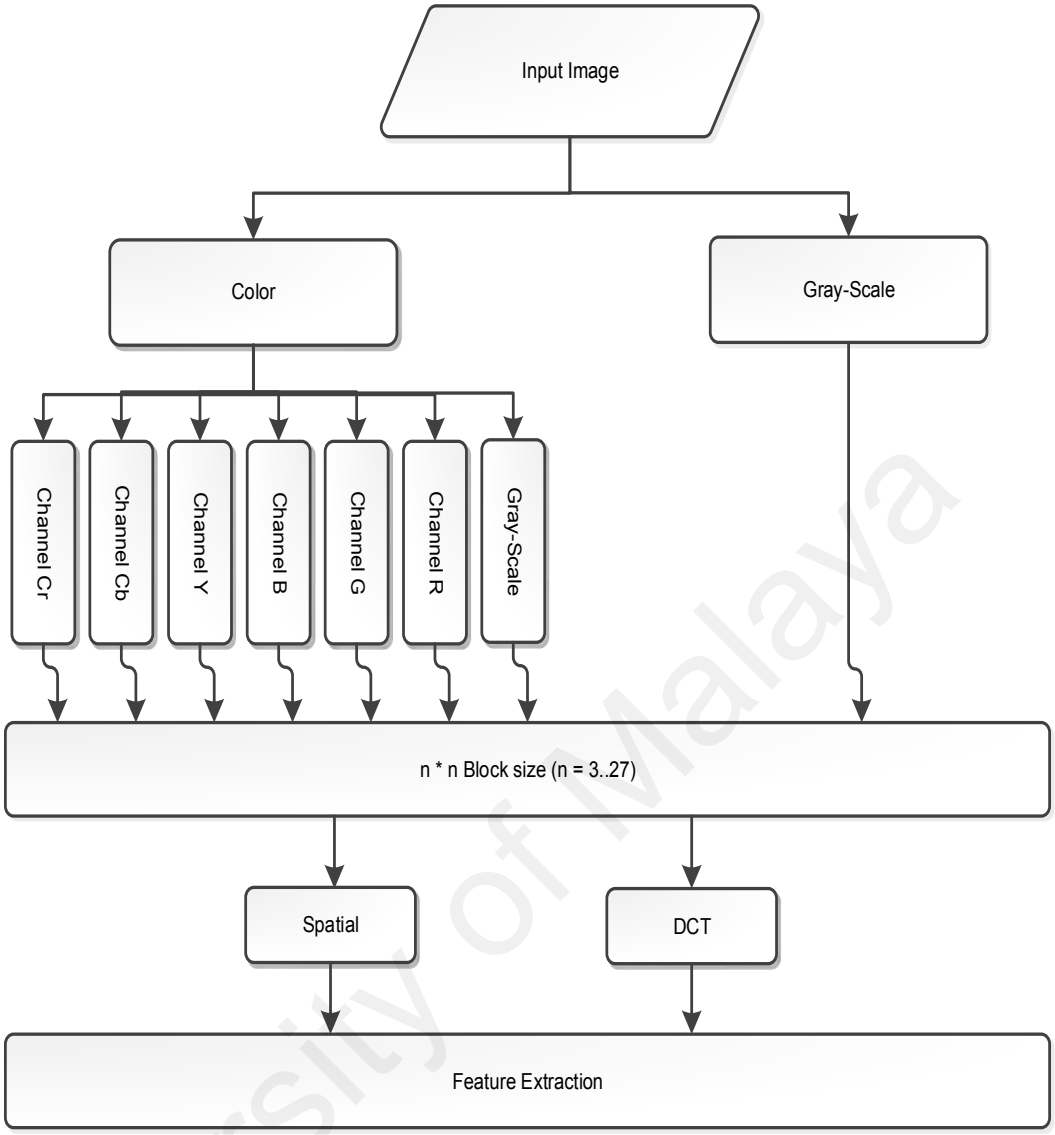


Figure 4.5: Flowchart of image preprocessing phase

Figure 4.6 illustrates the pseudo-code for image preprocessing phase. As already explained first different channels are captured from the given image, then the image is blocked and transformed. After that, the features are extracted from every block in spatial or frequency domain.

```
1.  read Img ← Image
2.  Img ← Gray(Img)/RGB(Img, R)/RGB(Img, G)/RGB(Img, B)/
3.            YCbCr(Img, Y)/YCbCr(Img, Cb)/YCbCr(Img, Cr)
4.  //multi-blocking
5.  For W ← 3..27
6.      (row, col) ← size(Img)
7.      m ← floor(row/W)
8.      n ← floor(col/W)
9.      For i ← 1..m
10.         For j ← 1..n
11.             B_{i,j} ← Pixels(B_{i,j}) or Freq(B_{i,j})
12.             SvB_W ← Feature_Extraction(B_{i,j})
13.         End for
14.     End for
15.     SvB_W ← mean(SvB_W), variance(SvB_W),
16.                skewness(SvB_W), kurtosis(SvB_W)
17. End for
18. Features ← (SvB_1, SvB_2, …, SvB_25)
```

Figure 4.6: Pseudo-code for Image Preprocessing

4.3 Feature Extraction

Image features include the global and local properties of an image such as average grey levels, intensity histogram shapes, circles, lines, texels, and contour shapes. Different methods have been developed to extract these features, and these methods have been applied in various image processing fields. Features extracted from images serve an important function in the detection and classification process, which aims to distinguish authentic images from spliced ones. In this section two proposed SVD-based feature extraction methods (SVD-Log and SVD-CN) for image splicing detection are presented.

4.3.1 Proposed Method 1: SVD-Log

As described in section 2.6, SVD is a method that transforms the correlated variables in a dataset into a set of uncorrelated ones to better demonstrate the different relationships among the original dataset (Baker, 2005). It can model the soft relationship between the image rows and columns which is modified by the splicing process. Therefore, the singular values could be applied as the features in image splicing detection to capture the modified correlation among the image pixels. Since the splicing process has various effects on the singular values, there is a need for a function to equally emphasize all singular values. Thus, the Logarithm of the inverse power of the singular values is proposed in (Gul & Kurugollu, 2010) to extract the features from the images for staganalysis. Due to the similarities between image steganography and image splicing operation, the Logarithm of the inverse power of the singular values is also applied in the first proposed method (Moghaddasi, Jalab, & Noor, 2014). Since there is a high dependency between the image pixels, the proposed method divides the image into subblocks with different block sizes (as described in section 4.2.2) to capture the interblock correlations. The following steps explain the SVD-Log feature extraction method:

Step 1: The image is divided into $n \times n$ blocks (n = 3, 4, 5, …, 27) according to the multi-size blocking approach described in section 4.2.2.

Step 2: For each block size n, calculate the singular value (Sv) vector for every subblock j (Gul & Kurugollu, 2010):

$$Sv_j = [\alpha_{1j}, \alpha_{2j}, \ldots, \alpha_{nj}] \quad \textbf{(4.2)}$$

Step 3: In this step, first the natural inverse Logarithm of each singular value is calculated, then the results are summed for each sub-block *j* (Gul & Kurugollu, 2010):

$$SVB_j = \sum_{i=1}^{n} \log(\frac{1}{\alpha}) \quad \textbf{(4.3)}$$

Step 4: Calculate 1st to 4th order statistics (average, variance, skewness, and kurtosis) to get various distributions of the values obtained from step 3 for each block size *n*:

$$\begin{aligned} F_n \; = [&avg(SvB_1, \ldots, SvB_n), var(SvB_1, \ldots, SvB_n), \\ &skewness(SvB_1, \ldots, SvB_n), Kurtosis(SvB_1, \ldots, SvB_n)] \\ &n = 3,4, \ldots, 27 \end{aligned} \quad \textbf{(4.4)}$$

There are 25 block sizes and for every block that is obtained from step 3, four numbers (average, variance, skewness, and kurtosis) are calculated. 1st to 4th order statistics have been applied to better reflect the distribution of the values captured in step 3. Therefore, the dimensionality of the proposed method is 100-D, (27 - 3 + 1 = 25) × 4 = 100. Figure 4.7 indicates the pseudo-code for feature extraction, applying the first proposed method (SVD-Log) in the same way as explained in steps 1 to 4.

1. Feature_Extraction($B_{i,j}$)
2. $(USV)_{i,j} \leftarrow SVD(B_{i,j})$
3. $Sv_{i,j} \leftarrow diag(S_{i,j})$
4. $Sv_{i,j} \leftarrow log(1/Sv_{i,j})$
5. $SvB_W \leftarrow sum(Sv_{i,j})$
6. End

Figure 4.7: Pseudo-code for the proposed method 1 (SVD-Log) feature extraction

4.3.2 Proposed Method 2: SVD-CN

In the second proposed method (SVD-CN), the concept of roughness measurement is applied instead of Logarithm. SVD-CN method has been proposed to simplify the calculation of Logarithm applied in the SVD-Log method, since CN only needs the maximum and minimum singular values instead of their distribution. CN also captures the splicing artifacts inside the image effectively. Roughness measure is inversely proportional to condition number. As explained in SVD properties (Section 2.6.2.6), CN is related to linear independence between the column vectors of a matrix. It is the ratio between the largest and smallest singular values of a matrix (Sadek, 2012).

$$Cond(M) = \frac{s_{max}}{s_{min}} \tag{4.5}$$

Sensitivity to changes increases with the increasing of the condition number. Random images (blocks) have low CNs (the lowest value is 1), while smooth images show high CNs (the highest value is ∞). In this algorithm, CN is calculated for every block in the process. To obtain a better emphasis on the CNs, the inverse of the CNs (roughness) is applied as follows (Sadek, 2012):

$$R(M) = \frac{1}{cond(M)} \tag{4.6}$$

Roughness measurement could be applied in different applications related to the human visual system (HVS) such as perceptual coding and perceptual data embedding (Sadek, 2012). It shows the rapid decreasing of singular values in the smooth images compared to those of random images. Therefore, roughness measurement can reflect the changes made in spliced images versus original images. Figure 4.8 (c) shows a graph of

singular values for the two images presented in Figures 4.8 (a) and 4.8 (b) (authentic and spliced) respectively. According to the graph (Figure 4.8 (c)), the singular values of the authentic image decreased rapidly in comparison with those of the spliced image.

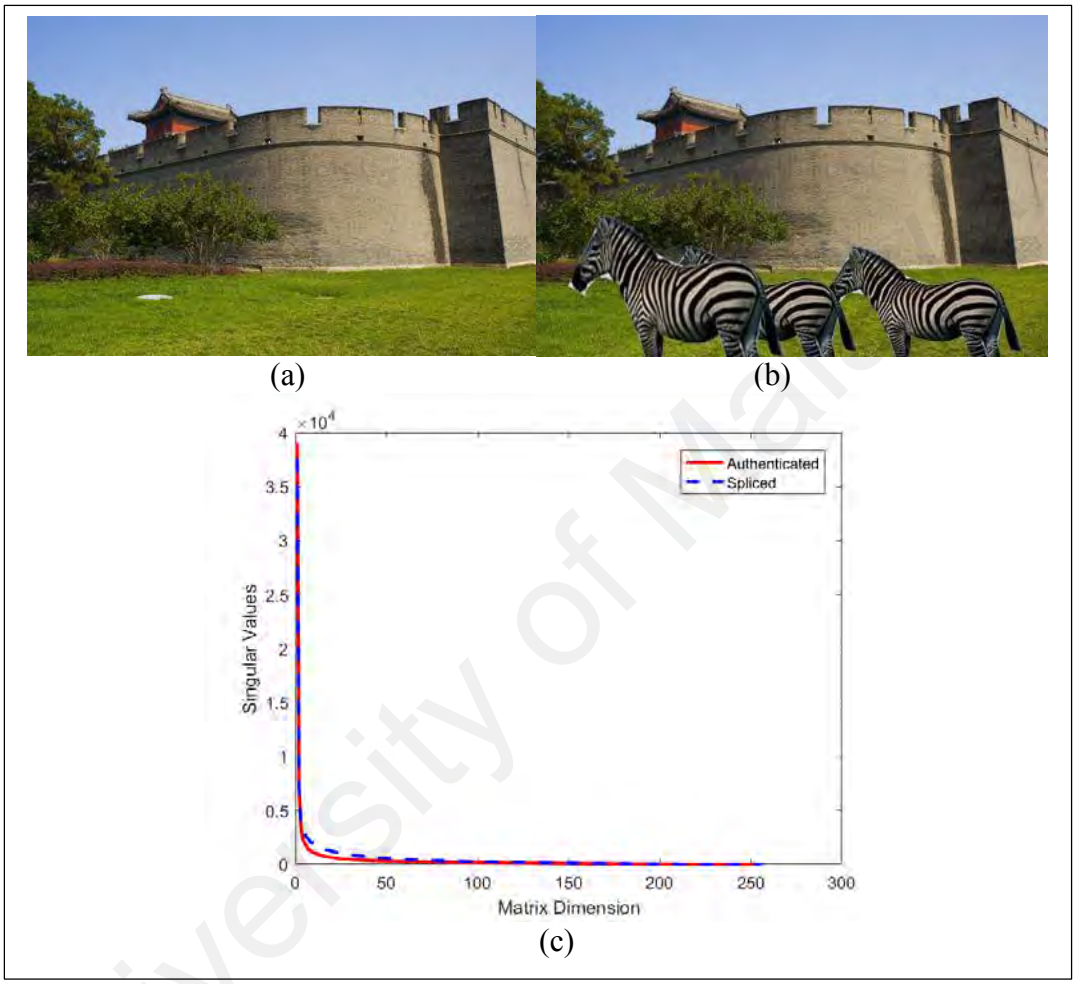


Figure 4.8: (a) Original Image, (b) Spliced Image, (c) Graph of singular values for images (a) and (b)

The steps applied in the SVD-CN feature extraction method are as follows:

Step 1: The given image is blocked into $n \times n$ blocks (n = 3, 4, 5, …, 27) according to the multi-size blocking approach described in section 4.2.2.

Step 2: For each block size n calculate the singular value (Sv) vector for every subblock j applying Equation (6.2).

Step 3: In this step the inverse of condition number (roughness measure) is calculated for the set of singular values obtained from second step for each subblock *j*:

$$SvB_j = \frac{\min([\alpha_{1j}, \alpha_{2j}, \ldots, \alpha_{nj}])}{\max([\alpha_{1j}, \alpha_{2j}, \ldots, \alpha_{nj}])} \quad \textbf{(4.7)}$$

Step 4: In this step the 1st to 4th order statistics (average, variance, skewness, and kurtosis) are calculated for the set of *SvB,* according to Equation (6.4).

Since the number of sub-blocks in size *n* is variable, different values are obtained from both methods; the 1st to 4th moments of values are calculated to be able to characterize the values and capture four numbers for every set of blocks in size *n*. Therefore, both methods have 100 dimensions for every given image. Figure 4.9 demonstrates the pseudo-code for the second proposed method (SVD-CN), as explained in steps 1 to 4.

1. Feature_Extraction($B_{i,j}$)
2. $(USV)_{i,j} \leftarrow SVD(B_{i,j})$
3. $Sv_{i,j} \leftarrow diag(S_{i,j})$
4. $Mx_{i,j} \leftarrow maximum(Sv_{i,j})$
5. $Mn_{i,j} \leftarrow minimum(Sv_{i,j})$
6. $CN_{i,j} \leftarrow Mx_{i,j} / Mn_{i,j}$
7. $SvB_W \leftarrow sum(1/CN_{i,j})$
8. End

Figure 4.9: Pseudo-code for the proposed method 2 (SVD-CN) feature extraction

Figure 4.10 illustrates the feature extraction phase for both proposed methods. All the steps and equations are explained completely in this section. All the features extracted by applying the first two phases are the input of the feature selection phase to be prepared for the classification phase.

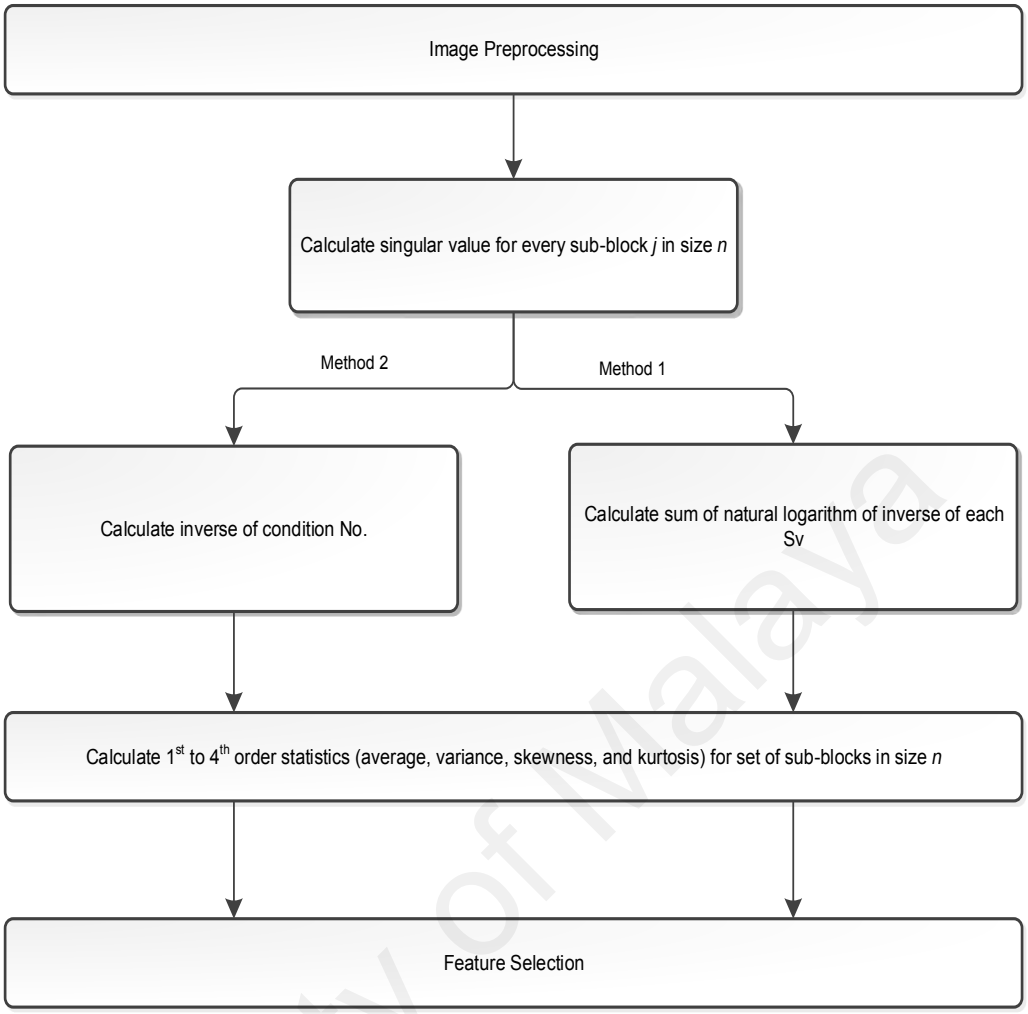


Figure 4.10: Flowchart of feature extraction phase

4.4 Feature Selection

When the features are extracted, it is the time to preprocess the extracted features before their classification. Feature selection is performed since the data might be redundant, not in appropriate format, contain outliers, and values might be missing or appear inconsistent. There are different approaches to make such data suitable for data mining, such as sampling, feature transformation, feature creation, feature reduction, etc. Generally, the data mining procedure can appropriately interpret the concepts in preprocessed data compared to the raw one.

The feature selection step is also performed in this study. As explained in section 2.7, two different feature selector methods (i.e. PCA and KPCA) are applied on both proposed SVD-based methods to make the extracted features ready for classification. The idea of selecting PCA and KPCA as feature processors is to examine the effect of two different methods (linear and non-linear) on the extracted features. However, high correlations are generally found among the extracted features using the proposed SVD-based image splicing detection methods. PCA and kernel PCA are applied to reduce the correlations by eliminating the information redundancies from the features.

Figure 4.11 shows the standard deviation distribution of the features extracted from greyscale images in DVMM v1 dataset, Figure 4.12 in R channels of CASIA v1 and Figure 4.13 the red channel of DVMM v2 before and after application of PCA and kernel PCA, respectively. The standard deviation measures show how data are spread out from the mean. In this case, a high standard deviation implies a high correlation between the features.

Figures 4.11-4.13 indicate that the original features are highly correlated and their standard deviations are spread over a wide range in 100-D. After applying PCA, the standard deviations mostly concentrate on the first few features and decrease as dimensionality increases. However, the standard deviations are higher than those in the original features, which indicates that PCA is not an appropriate preprocessor for the proposed features. The experimental results in chapter 5 also show the same outcome. In contrast, the standard deviations were greatly reduced after applying kernel PCA on the original features. The features after applying kernel PCA were obviously highly uncorrelated.

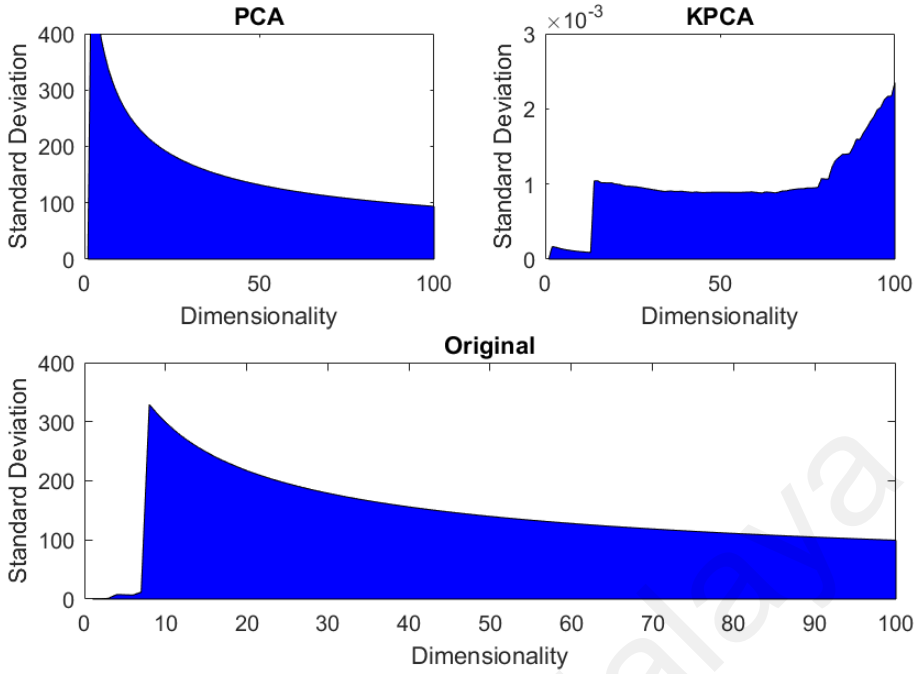


Figure 4.11: Standard deviation of the extracted features for image dataset DVMM v1

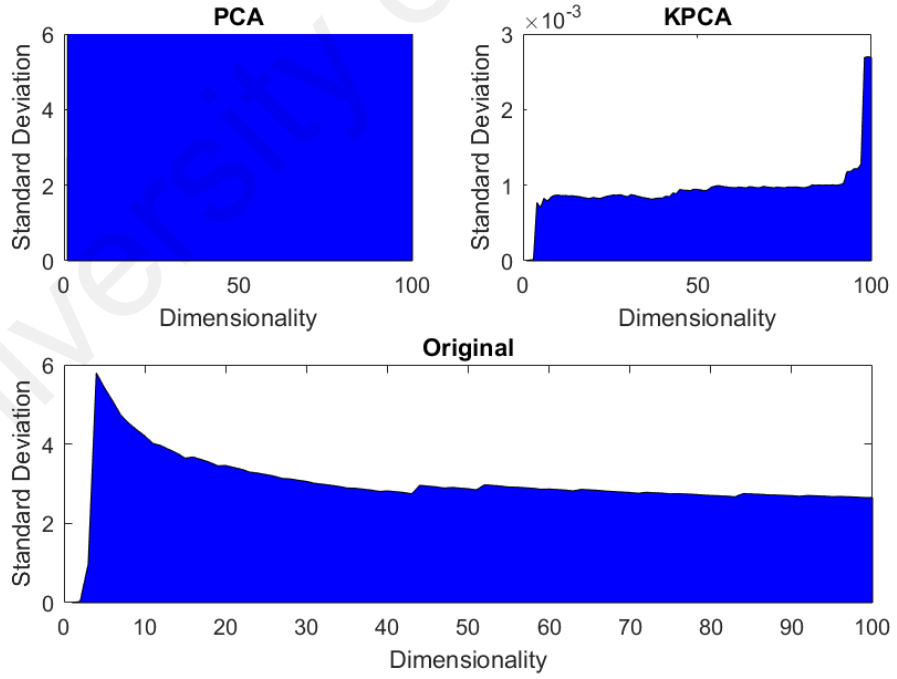


Figure 4.12: Standard deviation of the extracted features for image dataset CASIA, Red channel

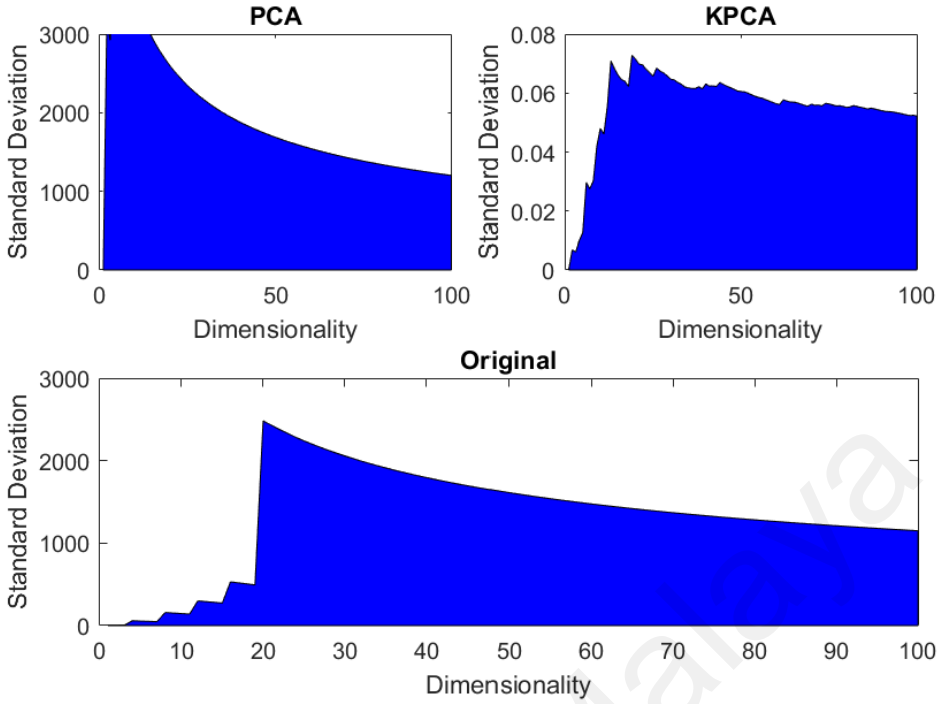


Figure 4.13: Standard deviation of the extracted features for image dataset DVMM v2, Red channel

Figure 4.14 illustrates the steps in preprocessing after the features are extracted by applying SVD-based splicing detection methods. MATLAB toolbox was utilized for dimensionality reduction (van der Maaten et al., 2009) to obtain features with different dimensions (10, 20, …, 90) from the extracted features using both proposed methods. Features with different dimensionalities are considered to obtain the optimal features with highest detection rate. Figure 4.15 shows the pseudo-code for feature selection phase.

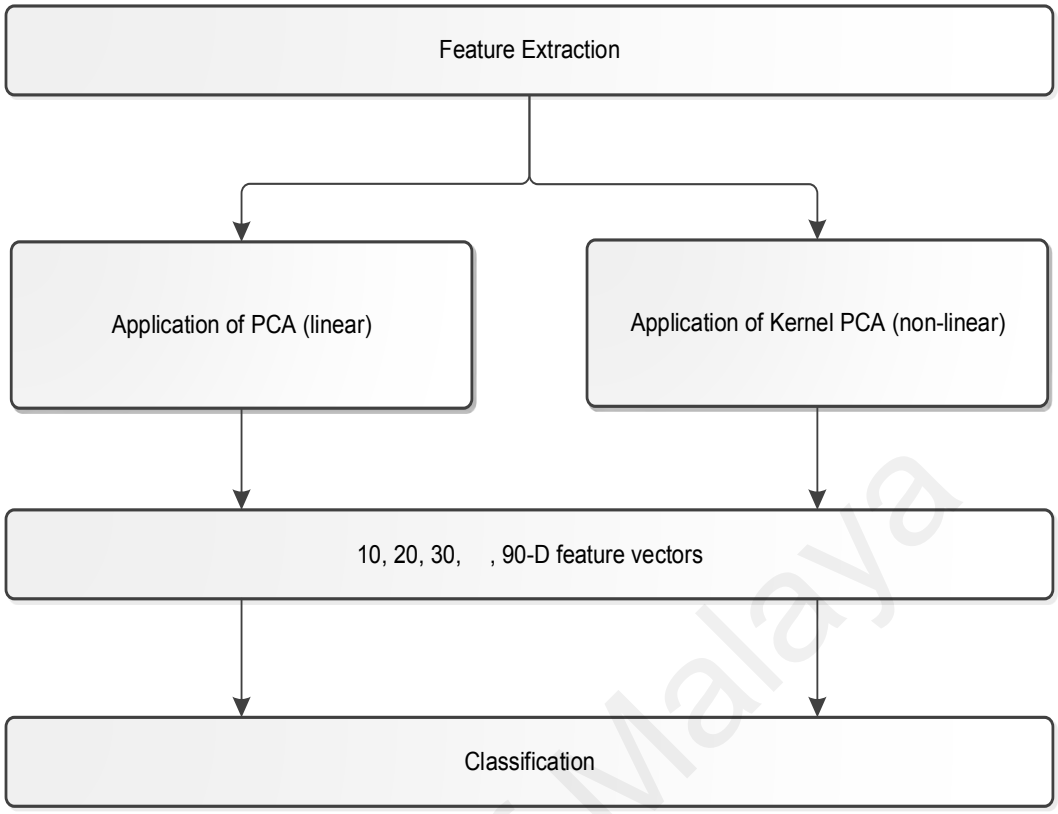


Figure 4.14: Flowchart of feature selection phase

```
1. Feature_Preprocessing(Features)
2. For D ∈ [10, 20, 30, …, 90]
3.       //calculate features with D dimension applying PCA or
4.        Kernel PCA
5.      Features ← Feature_Reduction(Features, D, KPCA/PCA)
6. End For
```

Figure 4.15: Pseudo-code for feature selection phase

4.5 Classification

The last phase in image splicing detection is classification. Image splicing detection is a binary classification to classify the images into authentic and spliced ones. In this research, LIBSVM (Chang & Lin, 2011) is applied as a classifier, which is a complete SVM toolbox in MATLAB. Radial Basis Function (RBF) is the selected kernel in the

experimental results. In addition, a grid search is employed to select the best parameters C and γ for classification (Hsu et al., 2010). To evaluate the performance of the classification the following concepts are applied (Kohl, 2012):

- TP: number of correctly detected authentic images
- TN: number of correctly detected spliced images
- FP: number of incorrectly detected authentic images
- FN: number of incorrectly detected spliced images

Sensitivity and specificity are statistical measures of the performance of a binary classification test, also known in statistics as classification function:

- Sensitivity (also called the true positive rate: TPR) measures the proportion of authentic images that are correctly detected.
- Specificity (also called the true negative rate: TNR) measures the proportion of spliced images that are correctly detected.

The following equations calculate sensitivity, specificity, and total accuracy of the classification results (Q. Gu et al., 2009; Kohl, 2012; X. Li et al., 2010):

$$TPR = \frac{TP}{TP + FN} \tag{4.8}$$

$$TNR = \frac{TN}{TN + FP} \tag{4.9}$$

$$(total)accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{4.10}$$

LIBSVM produces a model based on the input training set and then predicts the spliced features based on the input test set. The process of classification to get the best results is as follows (Hsu et al., 2010):

1. The data is first converted to feature vectors with labels to distinguish the authentic and spliced images.
2. A simple scaling is performed on the data to put the data in (-1, +1) range. The scaling process is very important, because it prevents the large numbers among the data dominating the ones with smaller values. In addition, scaling makes the kernel calculation much easier.
3. A kernel function for SVM is selected. Radial basis function (RBF) is one of the most common selected kernel. It is usually used when the class labels and the features are non-linearly interrelated.
4. RBF has two parameters: C and γ. SVM needs the optimal (C, γ) to be able to predict the testing data accurately.
5. Cross-validation is applied to look for the parameters (C, γ). Cross-validation divides the dataset into two subsets; one with known labels and the other one with unknown labels. The prediction rate obtained from the second dataset (unknown labels) will improve the performance of classification for the other datasets. The best recommendation is to apply grid-search on the parameters (C, γ) with cross-validation.
6. The parameters (C, γ) are applied to train the training dataset and makes the model.
7. Test the testing dataset by applying the model from the previous step.

Figures 4.16 and 4.17 show the flowchart of the classification phase and pseudo-code of this phase as elaborated in this section, respectively.

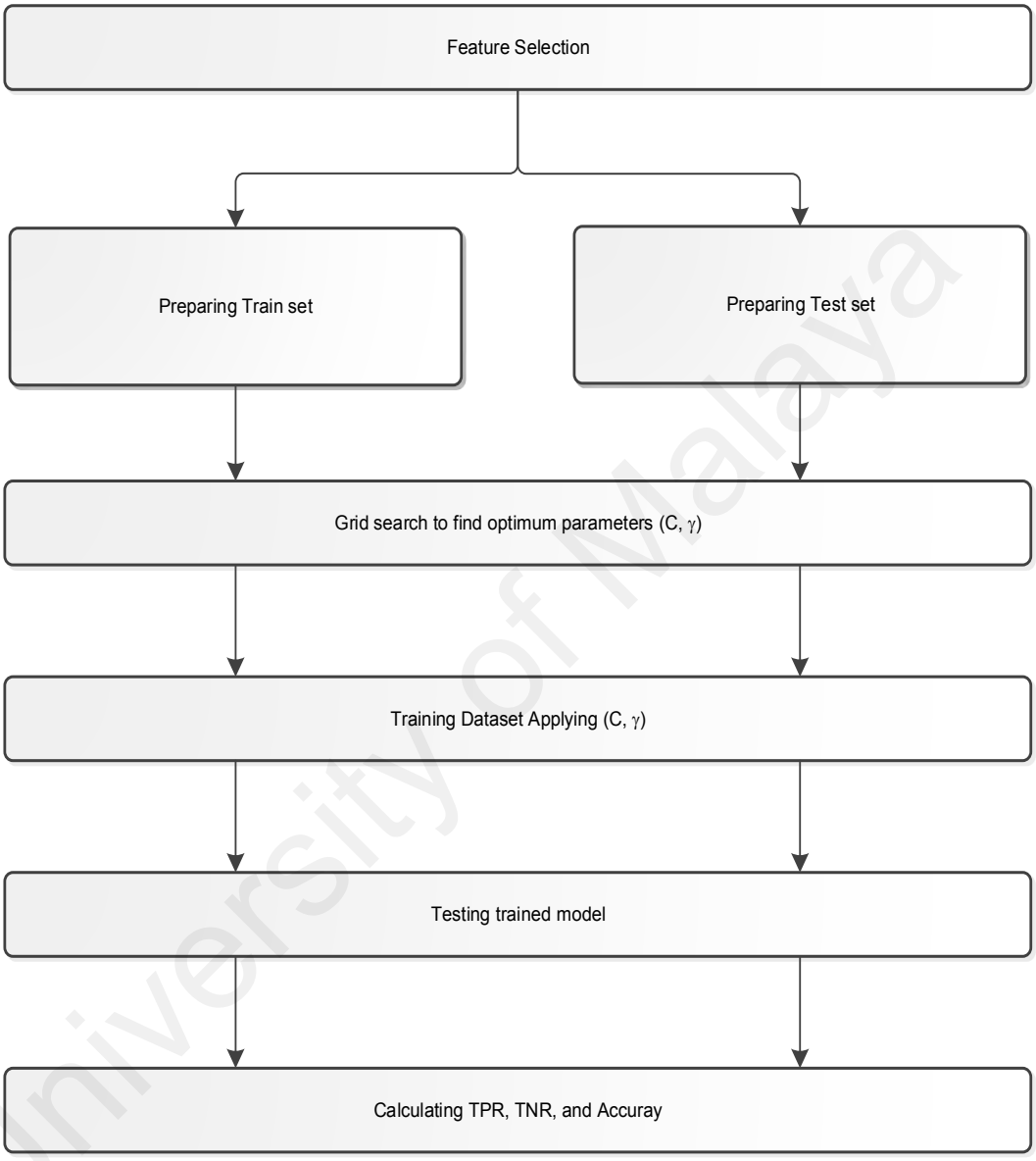


Figure 4.16: Flowchart of classification phase

```
1. Classification(Features)
2. Train_set ← 5/6 * Feature vectors
3. Test_set ← 1/6 * Feature vectors
4. Train_set ← scale(Train_set)
5. Test_set ← scale(Test_set)
6. For Iteration ← (1, 2, …, 30)
7.     //Training the train data in SVM
8.     Model ← svmtrain(train_set)
9.     //Testing the test data using the trained model in SVM
10.    Prediction ← svmpredict(test_set, model)
11. End For
12. Accuracy ← average(Prediction)
```

Figure 4.17: Pseudo-code for classification phase

4.6 Computational Complexity of the Proposed Methods

Calculating the complexity of an algorithm is a way to ensure that the algorithm is practical and feasible. Due to this declaration, the complexity of the proposed methods as calculated approximately is expressed in this section. To evaluate the complexity of the proposed schemes, a mathematical concept i.e. *Big-O* is applied.

1. Multi-Blocking:

$$O(MB) = (25 \times m) \quad \textbf{(4.11)}$$

where 25 is the number of block sizes (27 – 3 + 1) = 25, m is number of sub-blocks with size N.

2. Frequency Transformation:

$$O(DCT) \leq (N^2) \quad \textbf{(4.12)}$$

where $N \in [3 \ldots 27]$.

3. SVD Time Complexity:

   For a matrix with size $a \times b$ the time complexity is as follows (Cline & Dhillon, 2006; M. Gu & Eisenstat, 1995; Holmes et al., 2007):

$$O(SVD) = (\min(ab^2, a^2b)) \quad \textbf{(4.13)}$$

   Since all the matrices are square i.e. $a = b$ in this research thus *O(SVD)* can be rewritten as follows:

$$O(SVD) = (N^3) \quad \textbf{(4.14)}$$

   where $N \in [3 \ldots 27]$. Since the size of matrices are small (i.e. [3, 4, … 27]), SVD is feasible in the proposed algorithms.

4. SVD-Log and SVD-CN Time Complexity:

   The first proposed method utilizes Logarithm in its calculation and thus its time complexity is as follows:

$$O(SVD - Log) = (NLog(N)) \quad \textbf{(4.15)}$$

   However, the second proposed method applies condition number in its calculation which only focuses on the maximum and minimum Singular values. Therefore, its time complexity will be as:

$$O(SVD - CN) = (N) \quad \textbf{(4.16)}$$

   Where $N \in [3 \ldots 27]$ in Equations (4.15) and (4.16).

5. Feature Selection:

For a matrix with size $r \times l$ the time complexity of PCA and Kernel PCA are as follows (Günter et al., 2007):

$$O(PCA\ \&\ KPCA) = O(r \times l) \tag{4.17}$$

Therefore, the total complexity of the proposed algorithms is:

$$O\left(Total_{SVD-Log}\right) = 25m\,[N^2 + N^3 + NlogN] + O(r \times l) \tag{4.18}$$

$$O(Total_{SVD-CN}) = 25m\,[N^2 + N^3 + N] + O(r \times l) \tag{4.19}$$

Based on the total time complexities presented in Equations (4.18) and (4.19), the most time-consuming part is $25mN^3$. Table 4.1 indicates the runtime of feature extraction for different applied image datasets (i.e. DVMM v1, DVMM v2 and CASIA) utilizing the proposed methods. The results show that the runtime for method 2 (SVD-CN) is generally less than that for method 1 (SVD-Log) in all image datasets, due to applying CN instead of a Logarithm function, which was also verified by Equations (4.18) and (4.19). Moreover, DCT takes more time compared to spatial ones, due to the calculation time of DCT.

Table 4.1: Runtime of feature extraction applying the proposed methods

Methods	Time(s)		
	DVMM v1	**CASIA**	**DVMM v2**
SVD-Log (Spatial)	0.2158	1.2025	8.7383
SVD-Log (DCT)	0.2192	1.2877	9.1271
SVD-CN (Spatial)	0.2098	1.1396	7.7813
SVD-CN (DCT)	0.2227	1.1579	8.2538

4.7 Summary

In this chapter, the structural design of the two proposed image splicing detection methods have been presented. Each phase of the proposed schemes has been elaborated step by step. In addition, for every phase a flowchart has been drawn to illustrate the proposed schemes visually. There are four main phases (i.e. image preprocessing, feature extraction, feature selection and classification). The pseudo-code presented in each phase shows how the algorithms are implemented by applying different tools and toolboxes such as MATLAB, LIBSVM, and drtoolbx. The first SVD-based proposed method applies Logarithm (SVD-Log) in its core calculation, while the second one utilizes the concept of image roughness (SVD-CN). To improve the classification performance, two dimension reduction methods (i.e. PCA and Kernel PCA) are applied on the features. Finally, the TPR, TNR and detection accuracy have been calculated during the classification phase. Furthermore, the computational complexity of the proposed schemes has been presented to show the feasibility of the proposed detection algorithms.

CHAPTER 5: EXPERIMENTAL RESULTS AND DISCUSSION

5.1 Introduction

In this chapter, various experiments are designed and conducted to test and evaluate the two proposed SVD-based image splicing methods. The experimental results are presented based on TPR, TNR, accuracy, and Receiver Operating Characteristic (ROC) curves. The true positive rate (TPR) measures the proportion of positives that are correctly identified, while the true negative rate (TNR) measures the proportion of negatives that are correctly identified. Larger values of TPR, and TNR indicate better detection results. The classification performance of the proposed schemes is evaluated on three available image datasets – DVMM v1, DVMM v2, and CASIA – as explained in section 3.3.2. The classification system was implemented using MATLAB R2016a on a 3.30 GHz Intel (R) Core™ i3 processor with 4 GB RAM on a Windows 10 platform.

The Support Vector Machine (SVM) was the classifier applied in this research. SVM is a well-known supervised machine-learning applied in different methods, including pattern recognition. MATLAB codes for SVM are accessible in (Chang & Lin, 2011). LIBSVM is a known library that implements SVM. In this research, LIBSVM was used under the following conditions:

- Radial basis function (RBF) is used as a kernel function
- Grid search method is applied to obtain the best value for c and γ parameters so that the SVM classifier can accurately predict unknown data.

This chapter is divided into four sections, the first three sections are to evaluate the three available image datasets (DVMM v1, CASIA, and DVMM v2). In every section, the detection results are presented in different color models, with their channels separate in both spatial and frequency (DCT) domains. The results are also improved by applying

two-dimensional reduction methods (PCA and Kernel PCA). The last section presents a comprehensive comparison between the proposed schemes and the most recent developed image splicing detection methods.

5.2 Classification Performance of DVMM v1.0 Image Dataset

The first evaluated image dataset is DVMM v1.0. As explained in section 3.3, all the images in this dataset are in grayscale. In this section the classification rates of the features extracted from this image dataset using both proposed methods are presented.

5.2.1 Classification Performance of SVD-Log (Grayscale)

Table 5.1 illustrates the results from the original dimension of the extracted feature for the proposed SVD-Log with 100-D obtained from the DVMM v1.0 grayscale image dataset. A detection accuracy of 77.65% was achieved from the images in spatial domain, proving that a high correlation exists among the features, as indicated in Figure 4.11. Thus PCA and Kernel PCA are applied to remove the redundant features and improve SVM performance.

Table 5.1: Detection accuracy of SVD-Log with the original dimension of 100 in Spatial and Frequency Domains

Original	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
100-D	77.24	78.05	77.65	80.49	81.10	80.79

Table 5.2 presents the detection accuracy results after the application of PCA on SVD-Log with 10-90 dimensions in spatial and frequency domains. The results indicate that

applying PCA on the features before classification does not improve the classification performance. Thus PCA cannot remove the redundant dimensions from these feature vectors. The results presented in Table 5.2 indicate there is a high correlation between the features after PCA application, which also proven in (Figure 4.11).

Table 5.2: Detection accuracy of SVD-Log with PCA in different dimensions in Spatial and Frequency Domains

PCA	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
10-D	100	0	50	31.71	90.24	60.98
20-D	100	0	50	40.65	91.46	66.06
30-D	100	0	50	66.67	78.66	72.66
40-D	100	0	50	69.11	78.05	73.58
50-D	100	0	50	78.86	78.66	78.76
60-D	100	0	50	78.86	76.83	77.85
70-D	100	0	50	83.74	76.22	79.98
80-D	100	0	50	82.93	77.44	80.18
90-D	100	0	50	**82.11**	**78.66**	**80.39**

Table 5.3 indicates the detection accuracy that resulted from applying Kernel PCA on the extracted features from SVD-Log. The same result with Table 5.2 is demonstrated in Table 5.3, which shows that Kernel PCA also cannot improve the classification process.

The results from Tables 5.1, 5.2, and 5.3 show the highest accuracy for SVD-Log in the spatial domain is for the original feature vectors (100-D) with 77.65% which is not a satisfactory result. Therefore, the other transforms (DCT) are investigated to obtain a better detection rate.

The results obtained from the features in DCT and SVD-Log are also presented in Tables 5.1, 5.2, and 5.3. Table 5.1 indicates the result from the original dimension of 100 as being 80.79%. Although the result shows an increase compared to the detection rate presented for the spatial domain in 100-D, the accuracy is still not high enough. Thus, the features are normalized to improve the classification performance. Tables 5.2 and 5.3 demonstrate the detection rates in different dimensions after applying PCA and Kernel PCA on the extracted features from SVD-Log in DCT.

The results after applying the PCA dimension reduction method are presented in Table 5.2, and indicate a great decrease from 80.39% to 60.98% when the dimension is reduced (90-D to 10-D).

However, the detection rates after the application of Kernel PCA are promisingly increased. These results show that greyscale images with 50-D exhibit a considerable increase in detection accuracy compared with those from the original dimension (80.79% vs. 98.78%), respectively. Figure 5.1 illustrates the receiver operating characteristic (ROC) curves of the original features, after the application of PCA in 90 dimensions and Kernel PCA in 50 dimensions, respectively. According to this Figure, Kernel PCA has the best effect on the features, while PCA has approximately the same result as the original features.

The area under the curve (AUC) of 1.00 also verifies that the best classification performance for the features is with Kernel PCA. This area measures discrimination, which identifies the classification performance (Sokolova et al., 2006).

Table 5.3: Detection accuracy of SVD-Log with Kernel PCA in different dimensions in Spatial and Frequency Domains

KPCA	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
10-D	100	0	50	0.00	100	50
20-D	100	0	50	91.06	88.41	89.74
30-D	100	0	50	100.00	96.95	98.48
40-D	100	0	50	100.00	96.95	98.48
50-D	100	0	50	**100.00**	**97.56**	**98.78**
60-D	100	0	50	98.37	96.95	97.66
70-D	100	0	50	99.19	96.95	98.07
80-D	100	0	50	99.19	96.95	98.07
90-D	100	0	50	99.19	96.95	98.07

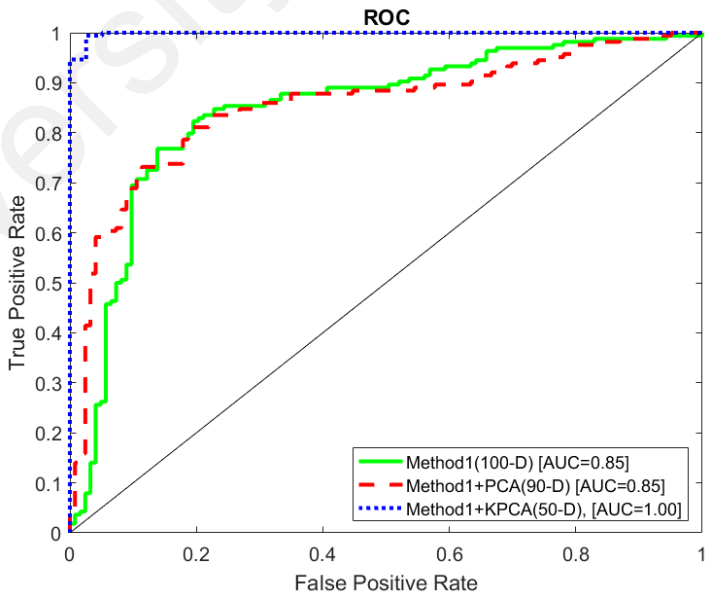


Figure 5.1: Comparison of Detection Performance between features in 100-D, with PCA in 90-D, and Kernel PCA in 50-D using SVD-Log

5.2.2 Classification Performance of SVD-CN (Gray-Scale)

Table 5.4 indicates the result for the second proposed method with the original dimensionality of 100 in spatial and frequency domains. The result shows a low detection rate of 65.28% was achieved in spatial domain, which is 12% less than the detection rate obtained from the first method with same conditions. Therefore, the extracted features need to be normalized by removing redundant features utilizing PCA and Kernel PCA.

Table 5.4: Detection accuracy of SVD-CN with the original dimension of 100 in Spatial and Frequency Domains

Original	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
100-D	67.11	63.46	65.28	59.35	68.90	64.13

The application of PCA on the extracted features using SVD-CN results in a detection accuracy of the range 54.79-66.20, which is very similar to the one achieved from the original dimensionality in Table 5.4. The results demonstrate that a high correlation exists among the extracted features, even after the application of PCA. Thus PCA as a linear dimension reduction method cannot efficiently increase the detection rate of the extracted features.

Table 5.5: Detection accuracy of SVD-Log with PCA in different dimensions in Spatial and Frequency Domains

PCA	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
10-D	59.87	63.46	61.67	40.65	79.88	60.26
20-D	56.58	62.18	59.38	30.89	83.54	57.22
30-D	59.87	55.13	57.50	33.33	82.93	58.13
40-D	62.50	51.92	57.21	39.84	76.83	58.33
50-D	48.68	60.90	54.79	37.40	76.83	57.11
60-D	63.82	53.21	58.51	39.84	71.95	55.89
70-D	55.26	62.82	59.04	43.90	73.78	58.84
80-D	64.47	64.74	64.61	53.66	71.95	62.80
90-D	**63.82**	**68.59**	**66.20**	52.85	73.78	63.31

Unlike PCA, Kernel PCA improves the classification performance incredibly. Table 5.6 indicates the results for the application of Kernel PCA on the extracted features from SVD-CN. The results vary in a wide range from 49.49% to 96.04%, which shows how dimensionality affects the detection rate. The highest detection rate is for 90 dimensions, with 96.04%. Thus, by applying Kernel PCA, the detection rate increased about 30%, which is a remarkable rise.

For the frequency domain (DCT), the results show almost same trend for original dimensionality, and also after PCA (Tables 5.4 and 5.5) as the results obtained from the spatial domain. Thus, the detection rates demonstrate that a high correlation still exists between the extracted features.

Table 5.6 indicates other results than Tables 5.4 and 5.5. Kernel PCA improves the classification rate to 97.15% for 90 dimensions for the features extracted from DCT. It also shows that Kernel PCA (as a non-linear dimension reduction method) efficiently normalizes the extracted features and decreases the correlation among the features, which is also proven in Figure 4.11.

Table 5.6: Detection accuracy of SVD-Log with Kernel PCA in different dimensions in Spatial and Frequency Domains

KPCA	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
10-D	0.00	100.00	50.00	0.00	100.00	50.00
20-D	0.00	100.00	50.00	0.81	100.00	50.41
30-D	0.81	99.39	50.10	0.81	100.00	50.41
40-D	0.81	99.39	50.10	0.81	99.39	50.10
50-D	0.81	98.17	49.49	0.81	98.78	49.80
60-D	92.68	83.54	88.11	96.75	95.73	96.24
70-D	95.12	94.51	94.82	97.56	96.34	96.95
80-D	96.75	94.51	95.63	96.75	94.51	95.63
90-D	97.56	94.51	96.04	**96.75**	**97.56**	**97.15**

A comparison of ROC curves for original features, after PCA (90-D) and after Kernel PCA (90-D) using SVD-CN are demonstrated in Figure 5.2. The curves in this Figure show that the performance of features after the application of Kernel PCA is the best among the three curves.

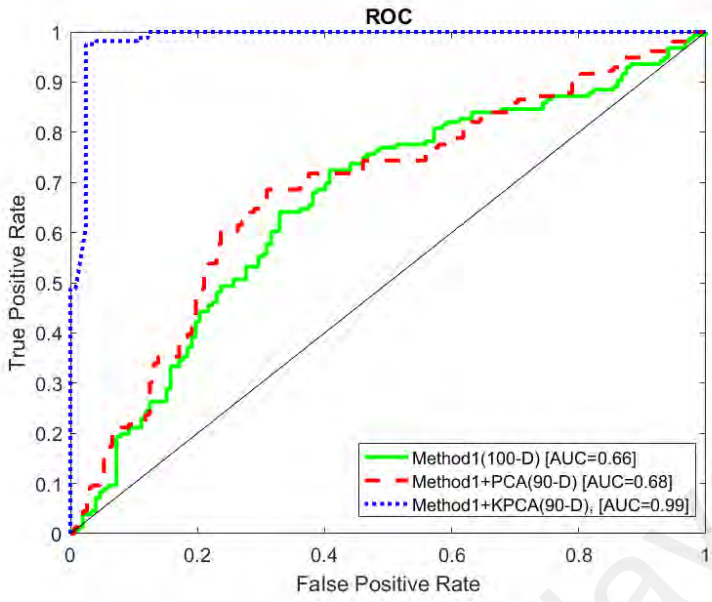


Figure 5.2: Comparison of Detection Performance between features in 100-D, with PCA in 90-D, and Kernel PCA in 90-D using SVD-CN

5.3 Classification Performance of CASIA Image Dataset

As explained in section 4.2.1, splicing operation makes some modification in the statistical features of the spliced image. These modifications have different reflections in each color channel or model. Therefore, it is important to find the color channel or model that best reflects the splicing artifacts.

The second image dataset applied to evaluate the proposed methods is the CASIA image dataset. CASIA, as mentioned in section 3.3, is a colored dataset and thus the features are extracted from the images in grey, R, G, B, Y, Cb, and Cr channels to investigate how the proposed image splicing detection schemes affect the images in different channels. Moreover, the individual channels are combined together to inspect their performance without considering any individual channels in both proposed methods.

5.3.1 Classification Performance of SVD-Log (Grayscale)

First of all, the images are converted to greyscale and their classification results are presented. Table 5.7 indicates the accuracy rate in spatial and frequency domains (greyscale) for the first proposed method with original dimensions of 100. Just like the DVMM v1 dataset, the accuracy rate of 59.06% shows that the features need some feature selection methods (i.e. PCA and Kernel PCA) to improve the classification process.

Table 5.7: Detection accuracy of SVD-Log with the original dimension of 100 in Spatial and Frequency Domains

Original	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
100-D	66.23	51.88	59.06	68.18	63.16	65.67

Tables 5.8 and 5.9 demonstrate the detection accuracies of the features extracted using SVD-Log after applying PCA and Kernel PCA in different dimensions. PCA and Kernel PCA again cannot improve the classification performance for these extracted features in the spatial domain. The results show the same trend as DVMM v1, in which the detection rates from the extracted features using SVD-Log in the spatial domain are not improved by applying PCA and KPCA.

After obtaining unsatisfactory results in the spatial domain, the features are extracted in DCT using SVD-Log. The detection rates for these features are generally higher than the detection rates in the spatial domain (greyscale). Table 5.9 indicates the highest detection accuracy of 95.47% for these features in 90-D after the application of Kernel PCA.

Table 5.8: Detection accuracy of SVD-Log with PCA in different dimensions in Spatial and Frequency Domains

PCA	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
10-D	100.00	0.00	50.00	77.27	28.57	52.92
20-D	100.00	0.00	50.00	83.12	27.82	55.47
30-D	100.00	0.00	50.00	70.13	60.15	65.14
40-D	100.00	0.00	50.00	67.53	63.91	65.72
50-D	100.00	0.00	50.00	68.18	61.65	64.92
60-D	100.00	0.00	50.00	66.23	64.66	65.45
70-D	100.00	0.00	50.00	66.23	60.90	63.57
80-D	100.00	0.00	50.00	**68.83**	**64.66**	**66.75**
90-D	100.00	0.00	50.00	67.53	62.41	64.97

Table 5.9: Detection accuracy of SVD-Log with Kernel PCA in different dimensions in Spatial and Frequency Domains

KPCA	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
10-D	100.00	0.00	50.00	100.00	0.00	50.00
20-D	100.00	0.00	50.00	100.00	0.00	50.00
30-D	100.00	0.00	50.00	100.00	0.75	50.38
40-D	100.00	0.00	50.00	100.00	0.75	50.38
50-D	100.00	0.00	50.00	99.35	0.75	50.05
60-D	100.00	0.00	50.00	94.81	89.47	92.14
70-D	100.00	0.00	50.00	96.75	90.98	93.87
80-D	100.00	0.00	50.00	96.75	92.48	94.62
90-D	100.00	0.00	50.00	**95.45**	**95.49**	**95.47**

5.3.2 Classification Performance of SVD-CN (Grayscale)

The same experimental processes have been conducted for SVD-CN in different channels of CASIA image dataset (i.e. Grey, R, G, B, Y, Cb, and Cr). Tables 5.10, 5.11, and 5.12 show the detection rates for the features extracted from SVD-CN in spatial and frequency domains (greyscale). Table 5.12 demonstrates the highest result of 92.41% for 70-D after the application of Kernel PCA. Compared to SVD-Log, the features extracted from SVD-CN in the spatial domain have higher detection rates after the application of Kernel PCA, which shows the nonlinear nature of these features. However, like the results presented for DVMM v1, Table 5.11 shows that the extracted features from both methods do not have linear natures, and thus PCA cannot improve their performance.

For the frequency domain results it is observed that only the results presented in Table 5.12 are considerable. According to this table the highest accuracy rate is for the features with only 50 dimensions (95.32%). Generally, the results in grayscale images show that some splicing artifacts are missing when the color image is converted to grayscale. However, the detection accuracies for both proposed methods are more than 95% which is a considerable rate.

Table 5.10: Detection accuracy of SVD-CN with the original dimension of 100 in Spatial and Frequency Domains

Original	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
100-D	59.21	66.67	62.94	58.44	60.90	59.67

Table 5.11: Detection accuracy of SVD-CN with PCA in different dimensions in Spatial and Frequency Domains

PCA	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
10-D	100.00	0.00	50.00	79.22	32.33	55.78
20-D	100.00	0.00	50.00	66.88	35.34	51.11
30-D	100.00	0.00	50.00	83.12	22.56	52.84
40-D	100.00	0.00	50.00	66.23	50.38	58.30
50-D	100.00	0.00	50.00	81.82	26.32	54.07
60-D	100.00	0.00	50.00	76.62	40.60	58.61
70-D	100.00	0.00	50.00	73.38	51.13	62.25
80-D	100.00	0.00	50.00	**74.68**	**53.38**	**64.03**
90-D	100.00	0.00	50.00	73.38	54.14	63.76

Table 5.12: Detection accuracy of SVD-CN with Kernel PCA in different dimensions in Spatial and Frequency Domains

KPCA	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
10-D	100.00	0.00	50.00	100.00	0.00	50.00
20-D	100.00	0.75	50.38	100.00	0.00	50.00
30-D	98.70	1.50	50.10	100.00	0.75	50.38
40-D	92.21	82.71	87.46	86.36	95.49	90.93
50-D	92.21	91.73	91.97	**97.40**	**93.23**	**95.32**
60-D	90.26	87.22	88.74	98.05	89.47	93.76
70-D	96.10	88.72	92.41	92.21	90.23	91.22
80-D	93.51	87.22	90.36	92.21	92.48	92.34
90-D	94.16	89.47	91.81	91.56	90.23	90.89

Figure 5.3 shows the ROC curves for detection performance of original features and after the application of Kernel PCA with 90 and 50 dimensions as the best results using SVD-Log and SVD-CN respectively. According to this Figure, the highest detection accuracy is for SVD-Log, with more than 98%, which is a remarkable detection rate.

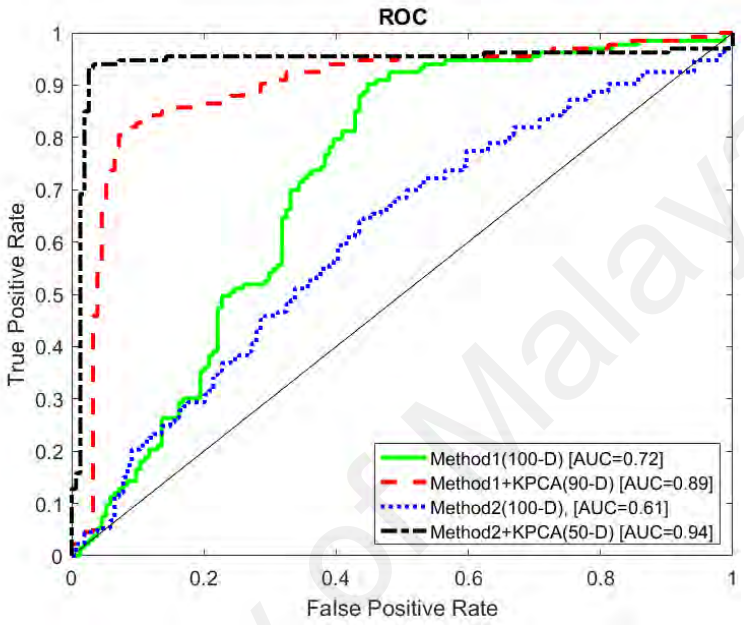


Figure 5.3: Comparison of Detection Performance between features in 100-D, Kernel PCA in 90-D, and Kernel PCA in 50-D using SVD-Log & SVD-CN (Gray-scale)

5.3.3 Classification Performance of SVD-Log (Red Channel)

Table 5.13 shows the detection accuracy for the features extracted in red channel utilizing SVD-Log. The detection accuracy of 59.18% again indicates the necessity of a method such as PCA and Kernel PCA to improve the classification performance. These results are presented in Table 5.14 and 5.15. However, the detection rates obtained from applying PCA and Kernel PCA on the extracted features in spatial domain of red channel are not compatible with these dimension reduction methods, and thus the classification decreased dramatically.

Due to unsatisfactory results from spatial domain, the features extracted from red channel of the images in DCT are tested to investigate whether DCT can reflect the splicing artifacts in the images or not. The results for the original dimensions in Table 5.13 show an increase in the detection rate from 59.18 to 68.64. Furthermore, applying Kernel PCA on the extracted features in different dimensions (Table 5.15) indicates a promising increase in accuracy to 96.60% in 80 dimensions. In addition, the results demonstrate how dimensionality can affect the classification performance.

Table 5.13: Detection accuracy of SVD-Log with the original dimension of 100 in Spatial and Frequency Domains

Original	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
100-D	69.48	48.87	59.18	73.38	63.91	68.64

Table 5.14: Detection accuracy of SVD-Log with PCA in different dimensions in Spatial and Frequency Domains

PCA	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
10-D	100.00	0.00	50.00	76.62	32.33	54.48
20-D	100.00	0.00	50.00	72.73	46.62	59.67
30-D	100.00	0.00	50.00	72.73	54.14	63.43
40-D	100.00	0.00	50.00	74.03	56.39	65.21
50-D	100.00	0.00	50.00	70.13	55.64	62.88
60-D	100.00	0.00	50.00	68.18	59.40	63.79
70-D	100.00	0.00	50.00	70.78	69.92	70.35
80-D	100.00	0.00	50.00	67.53	64.66	66.10
90-D	100.00	0.00	50.00	68.83	64.66	66.75

Table 5.15: Detection accuracy of SVD-Log with Kernel PCA in different dimensions in Spatial and Frequency Domains

KPCA	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
10-D	100.00	0.00	50.00	100.00	0.00	50.00
20-D	100.00	0.00	50.00	100.00	0.00	50.00
30-D	100.00	0.00	50.00	99.35	0.75	50.05
40-D	100.00	0.00	50.00	100.00	0.75	50.38
50-D	100.00	0.00	50.00	7.14	100.00	53.57
60-D	100.00	0.00	50.00	87.66	88.72	88.19
70-D	100.00	0.00	50.00	88.31	93.98	91.15
80-D	100.00	0.00	50.00	**95.45**	**97.74**	**96.60**
90-D	100.00	0.00	50.00	95.45	96.24	95.85

5.3.4 Classification Performance of SVD-CN (Red Channel)

The same experimental result with SVD-Log in red channel was also conducted for SVD-CN. Table 5.16 indicates the detection accuracy for the original features in spatial domain. The result demonstrates a detection accuracy of 69.34%, which is similar to the one obtained from SVD-Log in Table 5.13 (68.64%). However, the detection accuracy shows the need to apply Kernel PCA on the extracted features before the classification process. These detection rates are presented in Table 5.18, which shows an incredible increase compared to the rate from the original dimension. Except for 10 and 20 dimensions, the other dimensions have a detection accuracy of more than 95.92%, and the highest one is 97.62% for only 40-D.

Moreover, the results after the application of Kernel PCA in DCT also show rates of more than 90% for most of the dimensions. The highest accuracy is 93.92% for 50 dimensions. Generally, compared to SVD-Log, SVD-CN is more stable in Red channel since its experimental results in spatial and frequency domains are both satisfactory, thus it reflects the splicing artifacts in better way.

Table 5.16: Detection accuracy of SVD-CN with the original dimension of 100 in Spatial and Frequency Domains

Original	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
100-D	62.99	50.38	56.68	74.03	64.66	69.34

Table 5.17: Detection accuracy of SVD-CN with PCA in different dimensions in Spatial and Frequency Domains

PCA	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
10-D	74.03	42.11	58.07	62.34	55.64	58.99
20-D	56.49	54.89	55.69	66.88	43.61	55.25
30-D	72.08	42.11	57.09	64.94	52.63	58.78
40-D	46.10	76.69	61.40	46.75	74.44	60.59
50-D	64.94	44.36	54.65	64.94	44.36	54.65
60-D	70.13	40.60	55.37	66.88	44.36	55.62
70-D	74.68	51.88	63.28	70.78	52.63	61.71
80-D	74.68	57.89	66.29	75.97	56.39	66.18
90-D	74.03	63.91	68.97	73.38	60.15	66.76

Table 5.18: Detection accuracy of SVD-CN with Kernel PCA in different dimensions in Spatial and Frequency Domains

KPCA	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
10-D	100.00	0.00	50.00	0.00	50.00	0.00
20-D	100.00	0.00	50.00	0.00	50.00	0.00
30-D	96.10	96.99	96.55	48.87	73.14	48.87
40-D	**96.75**	**98.50**	**97.62**	90.23	93.16	90.23
50-D	98.05	96.99	97.52	91.73	93.92	91.73
60-D	98.70	94.74	96.72	86.47	90.31	86.47
70-D	97.40	96.24	96.82	85.71	90.91	85.71
80-D	98.70	93.98	96.34	90.98	92.57	90.98
90-D	99.35	92.48	95.92	93.23	93.69	93.23

Figure 5.4 illustrates a comparison between the detection performance of SVD-Log and SVD-CN in Red channel, utilizing ROC concept. The curves indicate that both methods have almost the same performance (only 1% difference) in Red channel.

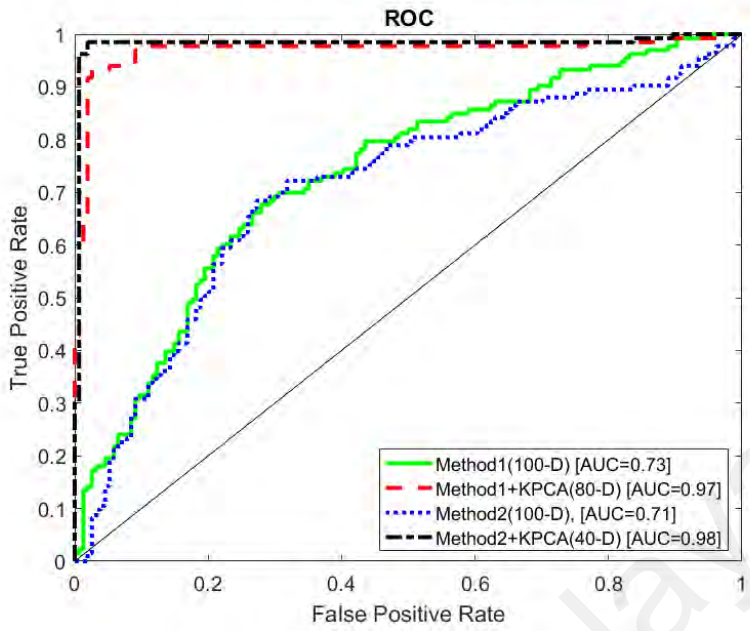


Figure 5.4: Comparison of Detection Performance between features in 100-D, Kernel PCA in 80-D, and Kernel PCA in 50-D using SVD-Log & SVD-CN (Red Channel)

Since the detection accuracy for all aforementioned experimental results only became acceptable after the application of Kernel PCA on the extracted features, the other experimental results (application of PCA) are omitted from this chapter to have more concentration on the valuable results.

Furthermore, the experimental results from the extracted features in the spatial domain applying SVD-Log do not indicate any promising improvement after the application of Kernel PCA. Thus these experimental results are also eliminated from the presented results for SVD-Log method.

5.3.5 Classification Performance of SVD-Log (Green Channel)

The second channel tested is the green channel. The detection accuracy from the green channel for the first proposed method in the spatial domain and in original dimensionality

of 100 is 59.52% (Table 5.19), which is very similar to the detection rates obtained from Red, Blue and grey channels with the same conditions (59.18, 58.94, and 59.06). This similarity verifies the strong correlation among the three channels because of the color filter array interpolation process. However, the accuracy result presented in Table 5.19 is an unsatisfactory rate and must be improved by applying Kernel PCA. Table 5.20 indicates a substantial increase in the accuracy of the features extracted in the frequency domain (DCT) from 65.77 to 97.57 with only 30 dimensions. Thus Kernel PCA can improve these kinds of features due to their nonlinear nature.

Table 5.19: Detection accuracy of SVD-Log with the original dimension of 100 in Spatial and Frequency Domains

Original	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
100-D	60.39	58.65	59.52	66.88	64.66	65.77

Table 5.20: Detection accuracy of SVD-Log with Kernel PCA in Frequency Domain

KPCA	Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)
10-D	100.00	0.00	50.00
20-D	97.40	93.98	95.69
30-D	**97.40**	**97.74**	**97.57**
40-D	98.70	95.49	97.10
50-D	96.10	93.23	94.67
60-D	97.40	93.98	95.69
70-D	95.45	95.49	95.47
80-D	95.45	96.24	95.85
90-D	94.81	90.98	92.89

5.3.6 Classification Performance of SVD-CN (Green Channel)

The same experimental results were also conducted for SVD-CN in Green channel. Table 5.22 demonstrates an incredible detection rate of 99.17% with 60 dimensions compared to the original dimension of 64.92%. It also indicates that the individual channels in the RGB color model can effectively reflect the splicing artifacts in the proposed SVD-based schemes.

Furthermore, Table 5.22 shows the highest accuracy for 60 dimensions with 96.67% for features in DCT. Compared to the detection rates obtained from the features in the spatial domain, there is a 2.5% decrease in the features extracted from SVD-CN in DCT. In comparison with SVD-Log, Kernel PCA improves the classification for SVD-CN in both domains (DCT vs. spatial and DCT) respectively.

Table 5.21: Detection accuracy of SVD-CN with the original dimension of 100 in Spatial and Frequency Domains

Original	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
100-D	61.54	68.31	64.92	64.42	70.42	67.42

Table 5.22: Detection accuracy of SVD-CN with Kernel PCA in different dimensions in Spatial and Frequency Domains

KPCA	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
10-D	100.00	0.00	50.00	4.81	100.00	52.40
20-D	5.77	100.00	52.88	5.77	100.00	52.88
30-D	98.08	97.18	97.63	73.08	50.00	61.54
40-D	98.08	98.59	98.33	100.00	92.48	96.24
50-D	98.08	99.30	98.69	98.05	93.23	95.64
60-D	**99.04**	**99.30**	**99.17**	99.35	93.98	96.67
70-D	99.04	99.30	99.17	98.05	86.47	92.26
80-D	96.15	99.30	97.72	97.40	90.23	93.81
90-D	98.08	99.30	98.69	93.27	92.96	93.11

The ROC curves for best detection performance of methods 1 & 2 and the curves for their original feature are presented in Figure 5.5. The black curve shows the incredible detection performance for SVD-CN.

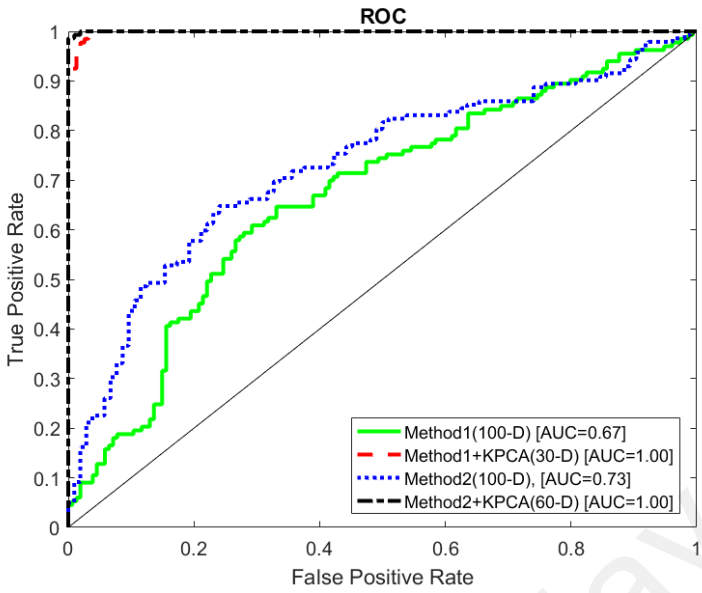


Figure 5.5: Comparison of Detection Performance between features in 100-D, Kernel PCA in 30-D, and Kernel PCA in 60-D using SVD-Log & SVD-CN (Green Channel)

5.3.7 Classification Performance of SVD-Log (Blue Channel)

The third channel is Blue channel. Tables 5.23 and 5.24 show the detection rates for the features extracted from SVD-Log in the spatial domain and DCT, before and after application of Kernel PCA respectively. The detection rate of 70.66% shows the same trend with those from Grayscale, red, and green channels, which shows that a high correlation exists among these channels. Table 5.24 demonstrates an increase of 25.24% after the application of Kernel PCA in 70 dimensions.

Table 5.23: Detection accuracy of SVD-Log with the original dimension of 100 in Spatial and Frequency Domains

Original	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
100-D	62.99	54.89	58.94	70.19	71.13	70.66

Table 5.24: Detection accuracy of SVD-Log with Kernel PCA in Frequency Domain

KPCA	Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)
10-D	3.85	100.00	51.92
20-D	3.85	100.00	51.92
30-D	4.81	100.00	52.40
40-D	8.65	100.00	54.33
50-D	96.10	95.49	95.80
60-D	92.21	98.50	95.35
70-D	**94.81**	**96.99**	**95.90**
80-D	95.45	95.49	95.47
90-D	94.81	95.49	95.15

5.3.8 Classification Performance of SVD-CN (Blue Channel)

However, SVD-CN shows a different detection result in the spatial domain of Blue channel, compared to the result from SVD-Log. A considerable detection rate of 97.20% is derived from the features in Blue channel and spatial domain with only 40 dimensions (Table 5.26).

Moreover, Table 5.26 shows an improvement in the detection rate for SVD-CN in Blue channel and frequency domain (DCT), with 99.30% in 40 dimensions. Since DCT has the capability to decorrelate the pixels in an image, thus this decorrelation better reflects the splicing artifacts and the detection rate is increased.

Table 5.25: Detection accuracy of SVD-CN with the original dimension of 100 in Spatial and Frequency Domains

Original	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
100-D	65.58	62.41	64.00	71.15	69.72	70.44

Table 5.26: Detection accuracy of SVD-CN with Kernel PCA in different dimensions in Spatial and Frequency Domains

KPCA	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
10-D	2.88	100.00	51.44	3.85	100.00	51.92
20-D	3.85	99.30	51.57	5.77	99.30	52.53
30-D	91.35	95.07	93.21	96.15	94.37	95.26
40-D	97.40	96.99	97.20	**100.00**	**98.59**	**99.30**
50-D	94.16	95.49	94.82	99.04	98.59	98.82
60-D	93.51	89.47	91.49	100.00	97.18	98.59
70-D	94.16	90.23	92.19	98.08	97.89	97.98
80-D	92.86	93.23	93.05	99.04	97.89	98.46
90-D	91.56	94.74	93.15	98.08	97.18	97.63

Figure 5.6 shows a comparison between the highest detection rates for both methods in Blue channel based on ROC curves. According to the curves, the best detection performance is for features from SVD-CN in 40 dimensions.

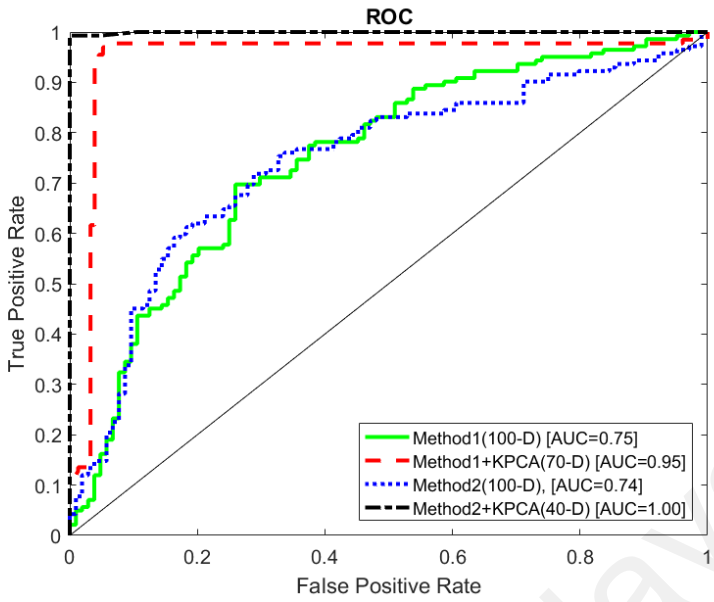


Figure 5.6: Comparison of Detection Performance between features in 100-D, Kernel PCA in 70-D, and Kernel PCA in 40-D using SVD-Log & SVD-CN (Blue Channel)

5.3.9 Classification Performance of SVD-Log (Y Channel)

The second color image model applied in our experimental results is the YCbCr model as explained in section 5.2.1. Tables 5.27 and 5.28 show the detection results for the features in Y channel after applying SVD-Log. According to these Tables, the accuracy rates in Y channel also indicate the same trend with those from the RGB channels, which verifies the correlation between the luminance channel (Y) and the RGB channels (i.e., Y is a linear combination of the R, G, and B channels). For DCT, as anticipated, the luma channel has a very similar detection accuracy with R, G, and B channels of 98.22% in 40 dimensions, after the application of Kernel PCA.

Table 5.27: Detection accuracy of SVD-Log with the original dimension of 100 in Spatial and Frequency Domains

Original	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
100-D	62.99	54.14	58.56	68.27	71.83	70.05

Table 5.28: Detection accuracy of SVD-Log with Kernel PCA in Frequency Domain

KPCA	Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)
10-D	100.00	0.00	50.00
20-D	98.05	94.74	96.39
30-D	98.70	95.49	97.10
40-D	**98.70**	**97.74**	**98.22**
50-D	97.40	94.74	96.07
60-D	96.75	93.23	94.99
70-D	96.75	93.23	94.99
80-D	96.75	93.23	94.99
90-D	96.75	91.73	94.24

5.3.10 Classification Performance of SVD-CN (Y Channel)

Tables 5.29 and 5.30 demonstrate the results obtained from the Y channel of the images in the spatial domain after applying SVD-CN. The results exhibited a lower detection rate (95.71) for the luma channel compared to the ones from R, G, and B channels (95.71 vs 97.62, 99.17, and 97.20 respectively). It can be observed that some splicing information has been missed in Y channel for SVD-CN, and thus the distribution of singular values (SVD-Log) is more informative than only the maximum and minimum singular values (SVD-CN).

Even the results obtained from luma channel in the frequency domain (DCT) follow the same trend as spatial domain, and show a detection accuracy of 92.89% derived from the features after the application of Kernel PCA, which is less than the results of R, G, and B channels (93.92, 96.67, and 98.82) respectively. Therefore, the results show that the linear independency between the image columns still remains high in luma channel.

Table 5.29: Detection accuracy of SVD-CN with the original dimension of 100 in Spatial and Frequency Domains

Original	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
100-D	65.58	51.13	58.36	71.15	66.20	68.68

Table 5.30: Detection accuracy of SVD-CN with Kernel PCA in different dimensions in Spatial and Frequency Domains

KPCA	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
10-D	3.85	100.00	51.92	2.88	100.00	51.44
20-D	5.77	99.30	52.53	6.73	100.00	53.37
30-D	4.81	100.00	52.40	6.73	100.00	53.37
40-D	99.04	85.92	92.48	93.27	92.25	92.76
50-D	**94.23**	**97.18**	**95.71**	94.23	91.55	92.89
60-D	92.31	97.18	94.75	94.23	90.85	92.54
70-D	92.31	90.85	91.58	94.23	88.73	91.48
80-D	92.31	92.96	92.63	90.38	92.96	91.67
90-D	92.31	95.77	94.04	92.31	90.14	91.22

Figure 5.7 also illustrates the best detection accuracy of more than 98% for SVD-CN in luma channel, based on ROC curves.

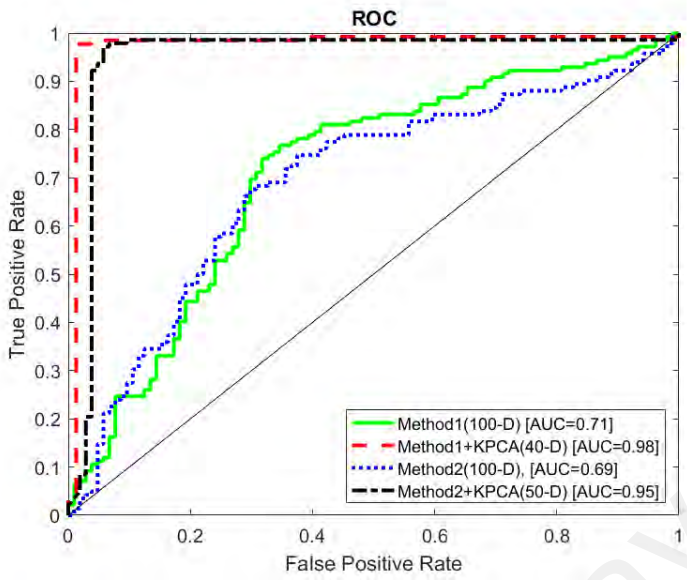


Figure 5.7: Comparison of Detection Performance between features in 100-D, Kernel PCA in 40-D, and Kernel PCA in 50-D using SVD-Log & SVD-CN (Y Channel)

5.3.11 Classification Performance of SVD-Log (Cb Channel)

In order to investigate the effectiveness of chroma channels in detecting spliced images, the features extracted in both methods in spatial and frequency domains are evaluated. Table 5.31 shows the detection accuracy of Cb channel in original dimensionality in both domains for SVD-Log. Unlike the detection performance in other channels which ranged from 55% to 70% in original dimensionality, an accuracy of 80.65% was derived from the features in the spatial domain of the Cb channel.

For the frequency domain, Table 5.31 indicates a detection rate of 89.56%, which is a substantial increase compared to the rate obtained from spatial domain (80.65%). However, the highest accuracy for Cb channel applying SVD-Log is 93.24% in 90 dimensions, which shows a substantial reduction compared to other aforementioned channels.

Table 5.31: Detection accuracy of SVD-Log with the original dimension of 100 in Spatial and Frequency Domains

Original	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
100-D	81.73	79.58	80.65	90.38	88.73	89.56

Table 5.32: Detection accuracy of SVD-Log with Kernel PCA in Frequency Domain

KPCA	Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)
10-D	82.69	88.03	85.36
20-D	81.73	78.87	80.30
30-D	89.42	83.80	86.61
40-D	95.19	80.28	87.74
50-D	89.42	83.10	86.26
60-D	95.19	90.85	93.02
70-D	93.27	90.85	92.06
80-D	94.23	89.44	91.83
90-D	**94.23**	**92.25**	**93.24**

5.3.12 Classification Performance of SVD-CN (Cb Channel)

Tables 5.33 and 5.34 demonstrate the detection rates for SVD-CN in the Cb channel of the images (both domains) with original and different dimensions respectively. Table 5.33 shows that, unlike SVD-Log, the Cb channel in SVD-CN has similar trends with other channels. However, after the application of Kernel PCA, a considerable accuracy of 98.22 is obtained for only 20 dimensions. This shows that the Cb features vectors have too much redundancy and thus Kernel PCA improved the detection accuracy by removing them from the features set.

Furthermore, for the frequency domain, the highest rate is achieved in 80 dimensions with 98.46%. Compared to SVD-Log, the Cb channel shows better performance in SVD-CN and thus it verifies that condition number better reflects the splicing artifacts in the images. Figure 5.8 also illustrates a comparative view of the maximum detection performance of SVD-Log and SVD-CN with original, 40 and 50 dimensions based on ROC curves.

Table 5.33: Detection accuracy of SVD-CN with the original dimension of 100 in Spatial and Frequency Domains

Original	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
100-D	63.46	66.90	65.18	26.92	92.96	59.94

Table 5.34: Detection accuracy of SVD-CN with Kernel PCA in different dimensions in Spatial and Frequency Domains

KPCA	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
10-D	2.88	100.00	51.44	1.92	100.00	50.96
20-D	98.70	97.74	98.22	3.85	100.00	51.92
30-D	96.10	92.48	94.29	96.15	97.18	96.67
40-D	96.10	91.73	93.92	92.31	96.48	94.39
50-D	98.05	90.98	94.51	96.15	97.18	96.67
60-D	97.40	90.98	94.19	96.15	95.77	95.96
70-D	96.75	90.23	93.49	93.27	96.48	94.87
80-D	97.40	90.98	94.19	**99.04**	**97.89**	**98.46**
90-D	94.81	90.98	92.89	96.15	95.07	95.61

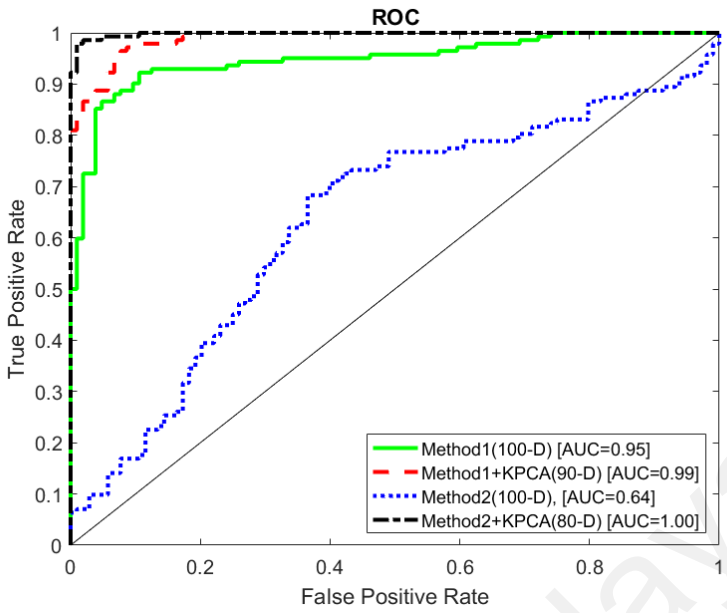


Figure 5.8: Comparison of Detection Performance between features in 100-D, Kernel PCA in 90-D, and Kernel PCA in 80-D using SVD-Log & SVD-CN (Cb Channel)

5.3.13 Classification Performance of SVD-Log (Cr Channel)

The last channel tested is the chroma channel, called Cr. Tables 5.35 and 5.36 indicate the detection results for features extracted from Cr channels using SVD-Log with original and different dimensions respectively. The highest detection accuracy of 84.21% was obtained from the original dimensions in the spatial domain, which compared to Cb channel (80.65) is an increase of 3.56%.

In the frequency domain (DCT), Cr channel using SVD-Log showed the highest accuracy of 92.46% in its original dimension. However, after the application of Kernel PCA, the accuracy rate is decreased according to Table 5.36. This verifies that the features vectors in Cr channel are all informative, and that there is no redundant feature to be eliminated from the features.

Table 5.35: Detection accuracy of SVD-Log with the original dimension of 100 in Spatial and Frequency Domains

Original	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
100-D	84.62	83.80	84.21	95.45	89.47	92.46

Table 5.36: Detection accuracy of SVD-Log with Kernel PCA in Frequency Domain

KPCA	Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)
10-D	80.52	90.98	85.75
20-D	77.92	86.47	82.19
30-D	89.42	83.80	86.61
40-D	95.19	80.28	87.74
50-D	92.31	79.58	85.94
60-D	**98.08**	**82.39**	**90.24**
70-D	97.12	80.28	88.70
80-D	98.08	78.17	88.12
90-D	97.12	78.87	87.99

5.3.14 Classification Performance of SVD-CN (Cr Channel)

The same experimental results were also conducted for SVD-CN in both domains for the Cr channel. Unlike SVD-Log, the detection accuracy obtained from the original dimensions was very low, at only 63.21% (Table 5.37). However, Kernel PCA improves this accuracy to 91.06% for 30 dimensions, which verifies that a high correlation exists among the image columns.

Since the results from the spatial domain of the images in the Cr channel using SVD-CN were not high enough, the images are transformed to the frequency domain using DCT to check the effectiveness of DCT on the Cr channel. Table 5.37 shows a detection rate of 81.72% for the original dimension, which exhibits a substantial growth compared to one from the spatial domain (63.21%). However, after the application of Kernel PCA, a considerable accuracy of 99.30% for 60 dimensions was obtained (Table 5.38). It also shows that SVD-CN is more sensitive to the chroma channel than luma channel. These observations are also illustrated in Figure 5.9, based on ROC curves for both proposed methods.

Table 5.37: Detection accuracy of SVD-CN with the original dimension of 100 in Spatial and Frequency Domains

Original	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
100-D	70.78	55.64	63.21	79.22	84.21	81.72

Table 5.38: Detection accuracy of SVD-CN with Kernel PCA in different dimensions in Spatial and Frequency Domains

KPCA	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
10-D	100.00	0.00	50.00	4.81	100.00	52.40
20-D	93.51	84.21	88.86	5.77	100.00	52.88
30-D	94.16	87.97	91.06	77.88	95.77	86.83
40-D	94.16	84.21	89.18	98.08	99.30	98.69
50-D	91.56	80.45	86.00	99.04	98.59	98.82
60-D	90.91	83.46	87.18	**100.00**	**98.59**	**99.30**
70-D	90.91	90.98	90.94	93.27	97.89	95.58
80-D	90.26	88.72	89.49	95.19	97.89	96.54
90-D	88.31	86.47	87.39	93.27	98.59	95.93

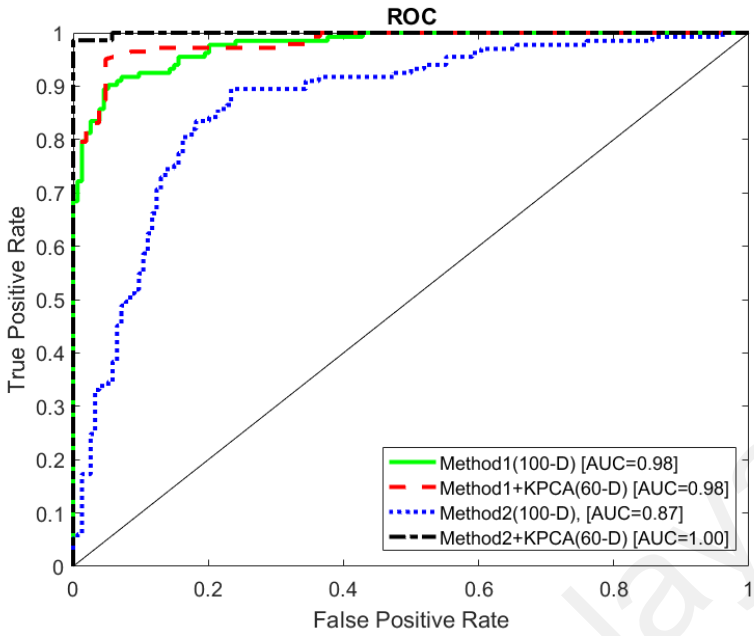


Figure 5.9: Comparison of Detection Performance between features in 100-D, Kernel PCA in 60-D, and Kernel PCA in 60-D using SVD-Log & SVD-CN (Cr Channel)

5.3.15 Classification Performance of SVD-Log (RGB color model)

The other experimental result conducted is to investigate the effectiveness of one color space, such as RGB or YCbCr, on both proposed methods. Thus the R, G, and B channel are combined as one feature set, and its classification performance is presented in Tables 5.39 and 5.40. The reason behind this kind of experimental procedure is that if the results are satisfactory, then there is no need to find an optimal channel. According to Table 5.39, the highest detection rate in the spatial domain is 68.87% for the original dimensions, which shows a similar trend with the results from each channel individually.

Thus the images are transformed to the frequency domain (DCT) to check how SVD-Log behaves in RGB and the frequency domain. Both tables indicate a considerable growth in detection accuracies. The highest detection rate is 95.45% for only 20 dimensions after the application of Kernel PCA.

Table 5.39: Detection accuracy of SVD-Log with the original dimension of 100 in Spatial and Frequency Domains

Original	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
100-D	67.31	70.42	68.87	69.23	76.06	72.64

Table 5.40: Detection accuracy of SVD-Log with Kernel PCA in Frequency Domain

KPCA	Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)
10-D	1.92	100.00	50.96
20-D	**92.31**	**98.59**	**95.45**
30-D	95.19	92.25	93.72
40-D	94.23	94.37	94.30
50-D	97.12	92.25	94.68
60-D	94.23	93.66	93.95
70-D	91.35	92.25	91.80
80-D	92.31	92.96	92.63
90-D	94.23	92.96	93.59

5.3.16 Classification Performance of SVD-CN (RGB Color model)

The same results in combining RGB channels are obtained for SVD-CN. Table 5.41 shows the accuracy of the images using SVD-CN in the original dimension. Table 5.42 improves the detection rate by applying Kernel PCA from 66.88% to 98.59% in 60 dimensions, which is an acceptable result compared to other experimental results obtained.

For the frequency domain, the highest accuracy is for 60 dimensions, with 93.47%, which is less than the one obtained from the spatial domain (98.59%). Generally, some of the individual channels have better detection rates compared to the combined one, and it demonstrates that all of the individual channels affect the performance of the combined channels. Figure 5.10 shows the ROC curves for the best detection rates in original, 20 and 60 dimensionality using both methods.

Table 5.41: Detection accuracy of SVD-CN with the original dimension of 100 in Spatial and Frequency Domains

Original	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
100-D	68.27	65.49	66.88	74.04	66.20	70.12

Table 5.42: Detection accuracy of SVD-CN with Kernel PCA in different dimensions in Spatial and Frequency Domains

KPCA	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
10-D	3.85	100.00	51.92	5.77	100.00	52.88
20-D	4.81	99.30	52.05	6.73	100.00	53.37
30-D	97.12	94.37	95.74	60.58	50.70	55.64
40-D	96.15	97.89	97.02	93.27	86.62	89.94
50-D	99.04	95.77	97.41	90.38	90.14	90.26
60-D	**100.00**	**97.18**	**98.59**	93.27	93.66	93.47
70-D	97.12	96.48	96.80	93.27	91.55	92.41
80-D	99.04	96.48	97.76	90.38	90.85	90.61
90-D	97.12	95.77	96.45	87.50	93.66	90.58

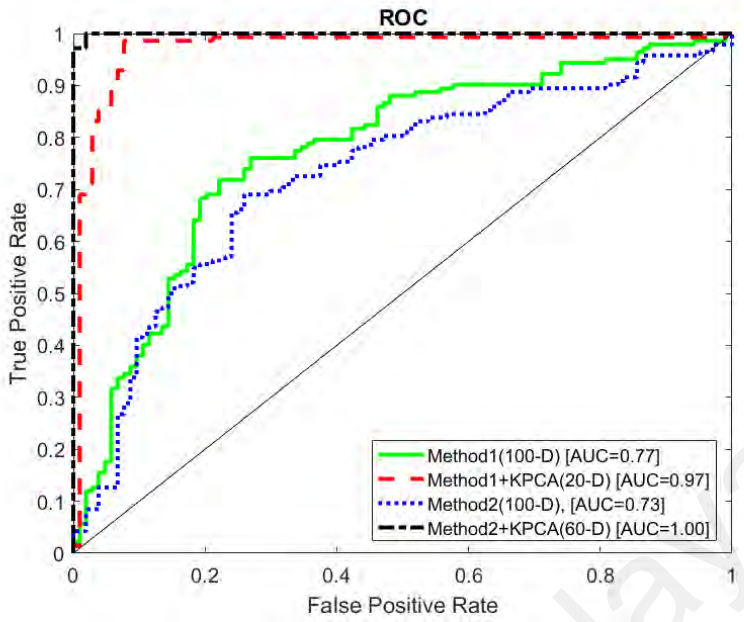


Figure 5.10: Comparison of Detection Performance between features in 100-D, Kernel PCA in 20-D, and Kernel PCA in 60-D using SVD-Log & SVD-CN (RGB Channel)

5.3.17 Classification Performance of SVD-Log (YCbCr Color Model)

Now luma and chroma channels (Y, Cb, and Cr) are combined to examine how the proposed methods behave in these combined channels. The detection results are presented in Tables 5.43 and 5.44 for the images in YCbCr color space using SVD-Log, before and after application of Kernel PCA, respectively. The maximum rate obtained is 79.85% for the original dimension in the spatial domain (Table 5.43). According to the results, YCbCr color space is not very informative in the spatial domain of the images, and thus the SVD contribution cannot reflect the splicing artifacts in the image dataset.

Due to unsatisfactory results in the spatial domain, YCbCr features are tested in the frequency domain (DCT). The original dimension exhibits a good result of 90.29%, which indicates a considerable growth compared to the one from spatial domain (79.85%). However, the maximum detection accuracy obtained for SVD-Log in combination of

YCbCr channels is 91.35% with 70 dimensions. Generally, compared to RGB channels, SVD distribution as applied in SVD-Log cannot effectively reflect the splicing artifacts in the YCbCr color model, and thus the detection rates are not very acceptable in this color space.

Table 5.43: Detection accuracy of SVD-Log with the original dimension of 100 in Spatial and Frequency Domains

Original	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
100-D	83.77	75.94	79.85	89.61	90.98	90.29

Table 5.44: Detection accuracy of SVD-Log with Kernel PCA in Frequency Domain

KPCA	Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)
10-D	89.42	84.51	86.97
20-D	85.58	84.51	85.04
30-D	86.54	83.10	84.82
40-D	85.58	85.92	85.75
50-D	94.23	85.92	90.07
60-D	87.50	86.62	87.06
70-D	**93.27**	**89.44**	**91.35**
80-D	91.35	89.44	90.39
90-D	91.35	89.44	90.39

5.3.18 Classification Performance of SVD-CN (YCbCr Color Model)

Luma and chroma channels are also combined to obtain the experimental results for SVD-CN. Table 5.46 shows incredible detection rates for the YCbCr color model in the spatial domain of the tested images. All the accuracy rates in different dimensions are higher than 97%, and the highest one is 99.62% with only 10 dimensions. It is observed that the combined channels have a better detection rate than the individual ones, and thus the columns in YCbCr color channels are highly independent.

Unlike the spatial domain, the combined channels in the frequency domain (DCT) of the images do not exhibit considerable detection rates. Table 5.46 shows the detection accuracies in different dimensions ranged from 80.17 to 86.19 which is not acceptable. Thus Y, Cb, and Cr channels have a negative impact when combined together in DCT transform.

Table 5.45: Detection accuracy of SVD-CN with the original dimension of 100 in Spatial and Frequency Domains

Original	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
100-D	32.69	93.66	63.18	29.81	81.69	55.75

Table 5.46: Detection accuracy of SVD-CN with Kernel PCA in different dimensions in Spatial and Frequency Domains

KPCA	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
10-D	**100.00**	**99.25**	**99.62**	92.31	78.17	85.24
20-D	99.35	97.74	98.55	90.38	79.58	84.98
30-D	99.35	97.74	98.55	85.58	76.06	80.82
40-D	100.00	96.99	98.50	80.77	79.58	80.17
50-D	100.00	96.99	98.50	81.73	83.10	82.41
60-D	99.35	95.49	97.42	77.88	88.03	82.96
70-D	99.35	96.24	97.80	74.04	89.44	81.74
80-D	98.70	95.49	97.10	84.62	89.44	87.03
90-D	99.35	94.74	97.04	83.65	88.73	86.19

Figure 5.11 shows a comparison between the best detection rates in the YCbCr feature set using both methods based on ROC curves. According to the displayed curves, YCbCr has the best reflection among all the aforementioned accuracies in the CASIA image dataset.

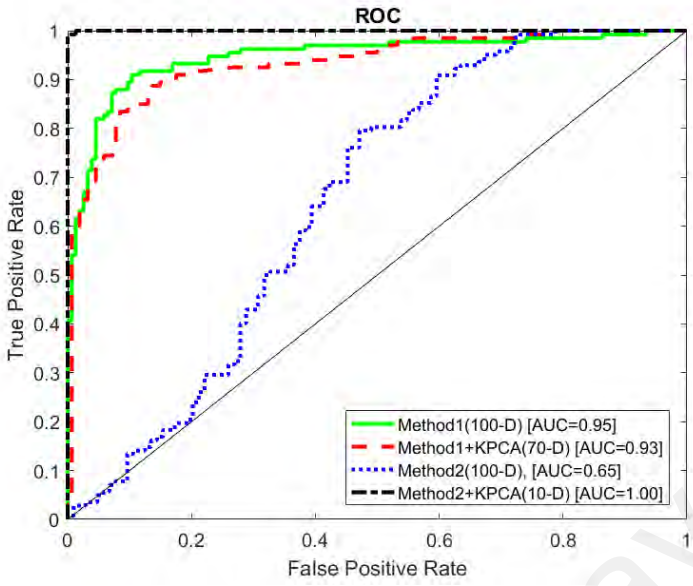


Figure 5.11: Comparison of Detection Performance between features in 100-D, Kernel PCA in 70-D, and Kernel PCA in 10-D using SVD-Log & SVD-CN (YCbCr Channel)

5.4 DVMM v2.0 Image Dataset

The third image dataset is DVMM v2. As mentioned in section 3.3, the images in this image dataset are in high resolution and thus the splicing artifacts are likely to be less detectable. To evaluate this image dataset, the same experimental procedures are conducted as were carried out on the CASIA dataset, and the results are presented in this section.

5.4.1 Classification Performance of SVD-Log (Grayscale)

Tables 5.47 and 5.48 show the detection rate of the images in greyscale for SVD-Log. Similar to the results in the other evaluated image datasets, the detection rates of the features extracted from the spatial domain of the images using SVD-Log are very low

(67.20%). This indicates that SVD-Log alone cannot capture the splicing modifications in the spatial domain of the images.

Unlike the spatial domain, the features extracted from the images in DCT form show considerable detection rates. It verifies that DCT can effectively reflect the splicing artifacts by de-correlating the data inside the image. Table 5.48 indicates a perfect detection rate of 100% with only 10 and 20 dimensions. The other dimensions also have a good accuracy of 96.77%, which is an acceptable enough detection rate.

Table 5.47: Detection accuracy of SVD-Log with the original dimension of 100 in Spatial and Frequency Domains

Original	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
100-D	66.67	67.74	67.20	76.67	74.19	75.43

Table 5.48: Detection accuracy of SVD-Log with Kernel PCA in Frequency Domain

KPCA	Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)
10-D	**100.00**	**100.00**	**100.00**
20-D	**100.00**	**100.00**	**100.00**
30-D	100.00	93.55	96.77
40-D	100.00	90.32	95.16
50-D	100.00	93.55	96.77
60-D	100.00	93.55	96.77
70-D	100.00	93.55	96.77
80-D	100.00	93.55	96.77
90-D	100.00	90.32	95.16

5.4.2 Classification Performance of SVD-CN (Grayscale)

Table 5.49 demonstrates the accuracies for the features in the spatial and frequency domains after applying SVD-CN. Unlike SVD-Log, SVD-CN can capture the splicing clues in the spatial domain of the images. According to Table 5.50, the highest detection accuracy for greyscale images in the spatial domain is 96.72% with 10 dimensions, which is still a considerable detection accuracy.

However, according to Table 5.50, the accuracy of the greyscale features in DCT after applying SVD-CN decrease to 90.16%, compared with the spatial domain rates. The results show that SVD-Log and SVD-CN have inverse behavior in spatial and DCT domain for greyscale images, in which the former is better in the frequency domain and the latter in the spatial domain.

Table 5.49: Detection accuracy of SVD-CN with the original dimension of 100 in Spatial and Frequency Domains

Original	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
100-D	66.67	67.74	67.20	60.00	70.97	65.48

Table 5.50: Detection accuracy of SVD-CN with Kernel PCA in different dimensions in Spatial and Frequency Domains

KPCA	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
10-D	**96.67**	**96.77**	**96.72**	93.33	70.97	82.15
20-D	96.67	90.32	93.49	90.00	90.32	90.16
30-D	100.00	74.19	87.10	83.33	87.10	85.22
40-D	96.67	83.87	90.27	86.67	77.42	82.04
50-D	93.33	96.77	95.05	86.67	77.42	82.04
60-D	96.67	90.32	93.49	80.00	80.65	80.32
70-D	96.67	93.55	95.11	93.33	67.74	80.54
80-D	93.33	87.10	90.22	86.67	74.19	80.43
90-D	93.33	93.55	93.44	90.00	74.19	82.10

A comparative view of the best detection rates in the original and 10 dimensions for both methods is illustrated in Figure 5.12, based on ROC curves. According to this Figure, the true positive rate is always high against the false positive rate.

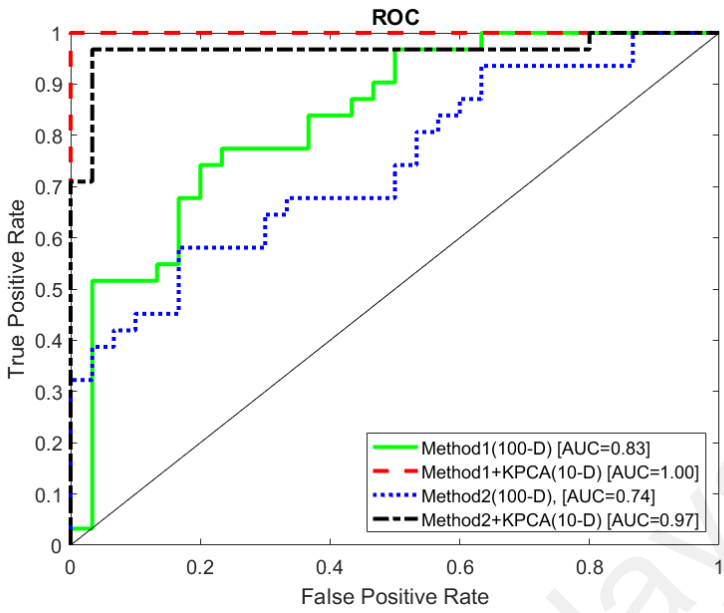


Figure 5.12: Comparison of Detection Performance between features in 100-D, Kernel PCA in 70-D, and Kernel PCA in 10-D using SVD-Log & SVD-CN (Gray-scale)

5.4.3 Classification Performance of SVD-Log (Red Channel)

The first color model examined by the proposed methods in DVMM v2 image dataset is the RGB color model. Table 5.51 shows a detection rate of 83.49% for Red channel images in the spatial domain using SVD-Log.

Table 5.52 indicates the results for Red channel of the images in the frequency domain (DCT). As anticipated, from SVD-Log a considerable accuracy of 96.72% is also obtained for Red channel. However, the rates for Red channel decreased incredibly compared to the rates from greyscale features.

Table 5.51: Detection accuracy of SVD-Log with the original dimension of 100 in Spatial and Frequency Domains

Original	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
100-D	76.67	90.32	83.49	70.00	78.79	74.39

Table 5.52: Detection accuracy of SVD-Log with Kernel PCA in Frequency Domain

KPCA	Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)
10-D	**96.67**	**96.77**	**96.72**
20-D	93.33	96.77	95.05
30-D	86.67	87.10	86.88
40-D	90.00	96.77	93.39
50-D	80.00	90.32	85.16
60-D	86.67	87.10	86.88
70-D	96.67	83.87	90.27
80-D	93.33	74.19	83.76
90-D	83.33	80.65	81.99

5.4.4 Classification Performance of SVD-CN (Red Channel)

Similarly, the accuracies obtained from the features in Red channel after applying SVD-CN, as demonstrated in Tables 5.53 and 5.54, are not high enough compared with those achieved from greyscale features. The detection rates ranges among 68.48% and 93.55% in spatial and frequency domains. The overall results from SVD-CN in Red

channel shows that this channel of image dataset does not effectively reflect the splicing traces left in the images.

Table 5.53: Detection accuracy of SVD-CN with the original dimension of 100 in Spatial and Frequency Domains

Original	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
100-D	30.00	78.79	54.39	60.00	70.97	65.48

Table 5.54: Detection accuracy of SVD-CN with Kernel PCA in different dimensions in Spatial and Frequency Domains

KPCA	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
10-D	90.00	81.82	85.91	90.00	90.32	90.16
20-D	80.00	93.94	86.97	96.67	87.10	91.88
30-D	70.00	96.97	83.48	**100.00**	**87.10**	**93.55**
40-D	90.00	93.94	91.97	96.67	87.10	91.88
50-D	80.00	93.94	86.97	93.33	87.10	90.22
60-D	85.00	96.97	90.98	90.00	96.77	93.39
70-D	60.00	96.97	78.48	93.33	93.55	93.44
80-D	55.00	96.97	75.98	90.00	93.55	91.77
90-D	40.00	96.97	68.48	86.67	87.10	86.88

Figure 5.13 shows the ROC curves for best results of both proposed methods in Red channel. According to these curves, SVD-Log has the best detection accuracy among the others.

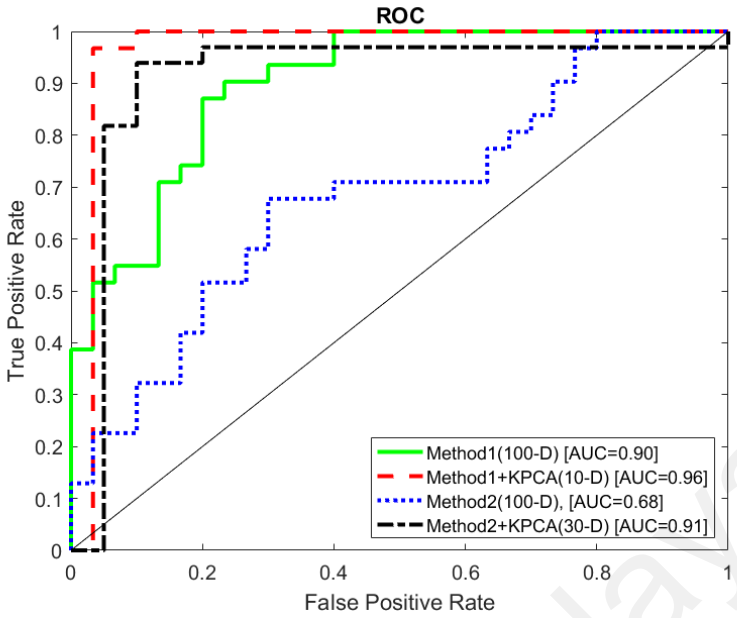


Figure 5.13: Comparison of Detection Performance between features in 100-D, Kernel PCA in 70-D, and Kernel PCA in 10-D using SVD-Log & SVD-CN (Red Channel)

5.4.5 Classification Performance of SVD-Log (Green Channel)

The second channel in the RGB color model is Green channel. Table 5.56 shows that all the accuracy rates for Green channel images in the frequency domain (DCT) after applying SVD-Log are all more than 95.11% and grow to achieve 98.33% with only 20 dimensions, which is a promising result compared to Red channel. However, the detection rate is still less than the one in greyscale features (98.33% vs. 100% respectively).

Table 5.55: Detection accuracy of SVD-Log with the original dimension of 100 in Spatial and Frequency Domains

Original	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
100-D	75.00	78.79	76.89	70.00	75.76	72.88

Table 5.56: Detection accuracy of SVD-Log with Kernel PCA in Frequency Domain

KPCA	Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)
10-D	93.33	100.00	96.67
20-D	**96.67**	**100.00**	**98.33**
30-D	96.67	96.77	96.72
40-D	96.67	96.77	96.72
50-D	96.67	93.55	95.11
60-D	96.67	93.55	95.11
70-D	96.67	93.55	95.11
80-D	96.67	93.55	95.11
90-D	96.67	93.55	95.11

5.4.6 Classification Performance of SVD-CN (Green Channel)

The Green channel for the SVD-CN also shows higher detection rates compared to the ones achieved in Red channel. Table 5.57 indicates an accuracy of 85.91% for 10 dimensions and it increases to reach a maximum amount of 95.45% with 90 dimensions. However, the results are still less than the ones from greyscale features (95.45 vs. 96.72 respectively). Generally, the detection rates show that the Green channel is more informative than the Red one in the DVMM v2 image dataset.

Table 5.57: Detection accuracy of SVD-CN with the original dimension of 100 in Spatial and Frequency Domains

Original	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
100-D	15.00	100.00	57.50	56.67	74.19	65.43

Table 5.58: Detection accuracy of SVD-CN with Kernel PCA in different dimensions in Spatial and Frequency Domains

KPCA	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
10-D	90.00	81.82	85.91	93.33	90.32	91.83
20-D	95.00	90.91	92.95	85.00	93.94	89.47
30-D	95.00	90.91	92.95	90.00	90.91	90.45
40-D	100.00	87.88	93.94	90.00	87.10	88.55
50-D	95.00	87.88	91.44	90.00	90.91	90.45
60-D	85.00	93.94	89.47	90.00	87.88	88.94
70-D	100.00	81.82	90.91	95.00	81.82	88.41
80-D	100.00	87.88	93.94	90.00	87.88	88.94
90-D	**100.00**	**90.91**	**95.45**	90.00	87.88	88.94

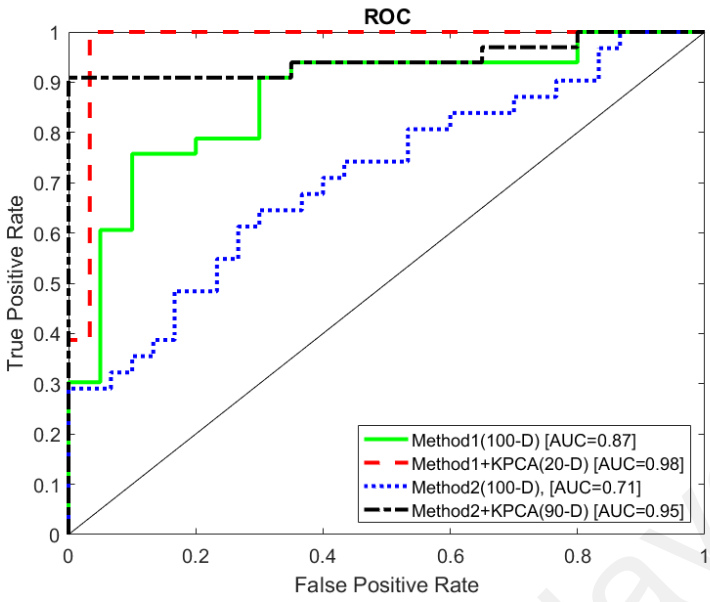


Figure 5.14: Comparison of Detection Performance between features in 100-D, Kernel PCA in 20-D, and Kernel PCA in 90-D using SVD-Log & SVD-CN (Green Channel)

5.4.7 Classification Performance of SVD-Log (Blue Channel)

Table 5.60 shows the accuracy rates for Blue channel of the images in the frequency domain after applying SVD-Log. The results indicate lower detection rates than those from greyscale, Red and Green channels, with a maximum percentage of 91.83 with only 20 dimensions. Moreover, the trend of detection results is similar to the CASIA image dataset, in which the Green channel had the best detection performance over the other two channels. It also verifies that SVD-Log is less sensitive to Blue channel than Red and Green channels.

Table 5.59: Detection accuracy of SVD-Log with the original dimension of 100 in Spatial and Frequency Domains

Original	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
100-D	70	84.85	77.42	80.00	77.42	78.71

Table 5.60: Detection accuracy of SVD-Log with Kernel PCA in Frequency Domain

KPCA	Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)
10-D	86.67	96.77	91.72
20-D	**93.33**	**90.32**	**91.83**
30-D	96.67	80.65	88.66
40-D	90.00	77.42	83.71
50-D	93.33	70.97	82.15
60-D	96.67	70.97	83.82
70-D	93.33	70.97	82.15
80-D	93.33	67.74	80.54
90-D	100.00	67.74	83.87

5.4.8 Classification Performance of SVD-CN (Blue Channel)

Table 5.62 demonstrates the maximum detection accuracy of 94.47% for the Blue channel images in the frequency domain using SVD-CN. Generally, the detection performance of the three R, G, and B channels in SVD-CN varies in a close range (93.55-95.45). The overall results show the Blue channel is also not highly informative in this kind of image dataset, and so far the best results were for greyscale images with both proposed methods.

Table 5.61: Detection accuracy of SVD-CN with the original dimension of 100 in Spatial and Frequency Domains

Original	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
100-D	55.00	69.70	62.35	50.00	75.76	62.88

Table 5.62: Detection accuracy of SVD-CN with Kernel PCA in different dimensions in Spatial and Frequency Domains

KPCA	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
10-D	96.67	77.42	87.04	**95.00**	**93.94**	**94.47**
20-D	93.33	90.32	91.83	90.00	87.88	88.94
30-D	95.00	87.88	91.44	90.00	93.94	91.97
40-D	95.00	75.76	85.38	85.00	90.91	87.95
50-D	100.00	81.82	90.91	**95.00**	**93.94**	**94.47**
60-D	100.00	81.82	90.91	90.00	90.91	90.45
70-D	86.67	90.32	88.49	90.00	96.97	93.48
80-D	90.00	80.65	85.32	80.00	93.94	86.97
90-D	95.00	78.79	86.89	80.00	90.91	85.45

Figure 5.15 shows a comparison between the best accuracy rates of the original, and of other dimensions for both proposed methods, based on ROC curves. As this Figure indicates, the further the curve is from the 45-degree diagonal of ROC space, the higher the detection accuracy is. Thus these curves also verify the best detection rates are for both methods after the application of Kernel PCA.

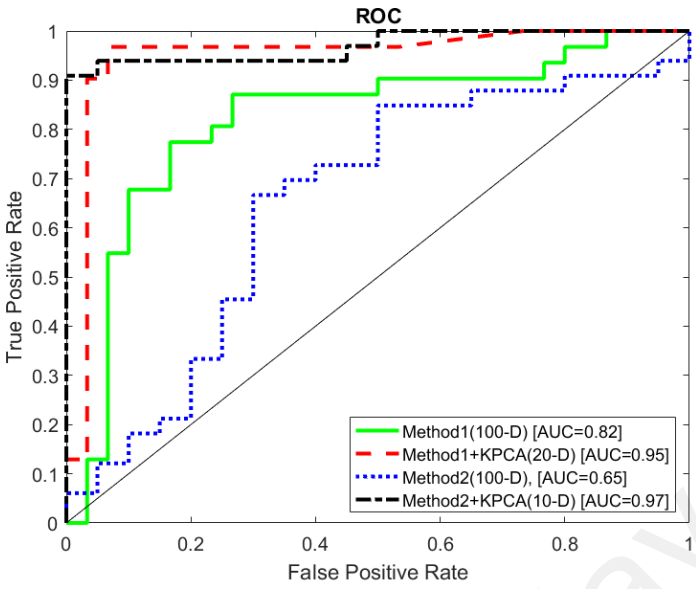


Figure 5.15: Comparison of Detection Performance between features in 100-D, Kernel PCA in 20-D, and Kernel PCA in 10-D using SVD-Log & SVD-CN (Blue Channel)

5.4.9 Classification Performance of SVD-Log (Y Channel)

The second evaluated color model is YCbCr. The detection rates of the luminance channel (Y) are presented in Tables 5.63 and 5.64 for SVD-Log in spatial and frequency (DCT) domains, before and after application of Kernel PCA. According to Table 5.64, the highest detection accuracy of 95% is for features with 10 dimensions, which is very close to the one achieved in Red channel (96.72). After that the detection rate is decreased dramatically to 91.88.

Table 5.63: Detection accuracy of SVD-Log with the original dimension of 100 in Spatial and Frequency Domains

Original	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
100-D	60.00	67.74	63.87	76.67	70.97	73.82

Table 5.64: Detection accuracy of SVD-Log with Kernel PCA in Frequency Domain

KPCA	Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)
10-D	**90.00**	**100.00**	**95.00**
20-D	93.33	90.32	91.83
30-D	96.67	87.10	91.88
40-D	93.33	83.87	88.60
50-D	96.67	80.65	88.66
60-D	93.33	87.10	90.22
70-D	93.33	80.65	86.99
80-D	90.00	90.32	90.16
90-D	90.00	90.32	90.16

5.4.10 Classification Performance of SVD-CN (Y Channel)

Tables 5.65 and 5.66 demonstrate the detection accuracy rates for the features extracted from Y channel of the images in spatial and frequency domains using SVD-CN. Generally, the detection rates in DCT are higher than those in the spatial domain for Y channel, which reaches a maximum of 95.45% for 10 and 30 dimensions as indicated in Table 5.66. However, for SVD-CN, the best result is still for greyscale images, and thus it shows that SVD-CN cannot capture the splicing traces left in the images from the individual channels.

Moreover, the results indicate that the luminance channel, which is a linear combination of R, G, and B channels, shows results in approximately same range of those from the individual R, G, and B channels (93.55, 95.45, and 94.47).

Table 5.65: Detection accuracy of SVD-CN with the original dimension of 100 in Spatial and Frequency Domains

Original	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
100-D	60.00	70.97	65.48	53.33	77.42	65.38

Table 5.66: Detection accuracy of SVD-CN with Kernel PCA in different dimensions in Spatial and Frequency Domains

KPCA	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
10-D	90.00	93.94	91.97	**100.00**	**90.91**	**95.45**
20-D	95.00	93.94	94.47	90.00	84.85	87.42
30-D	95.00	90.91	92.95	**100.00**	**90.91**	**95.45**
40-D	100.00	84.85	92.42	100.00	84.85	92.42
50-D	100.00	75.76	87.88	100.00	87.88	93.94
60-D	100.00	78.79	89.39	100.00	87.88	93.94
70-D	95.00	75.76	85.38	100.00	81.82	90.91
80-D	85.00	84.85	84.92	100.00	81.82	90.91
90-D	90.00	84.85	87.42	100.00	75.76	87.88

Figure 5.16 illustrates the ROC curves for performance classification of original features and after application of Kernel PCA, with the best rates achieved in both methods

in the luma channel. For the features with 10 dimensions extracted from SVD-Log, the true negative rate is 90% and the true positive rate reaches 100%, whereas conversely for features with 10 dimensions extracted from SVD-CN, the true negative rate is 100% and then the true positive rate gets 90% which is clearly depicted in the curves.

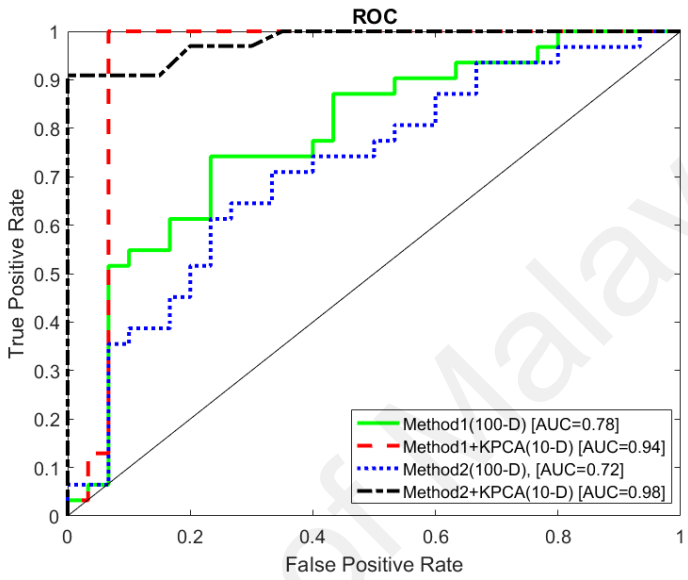


Figure 5.16: Comparison of Detection Performance between features in 100-D, Kernel PCA in 10-D, and Kernel PCA in 10-D using SVD-Log & SVD-CN (Y Channel)

5.4.11 Classification Performance of SVD-Log (Cb Channel)

The detection accuracies of the Chroma channel (Cb) is presented in Tables 5.67 and 5.68 using SVD-Log. The results indicated in Table 5.68 show almost the same trend as the Y channel obtained from Cb channel, with a maximum accuracy of 95.11% obtained for the features with 10 dimensions in DCT.

Table 5.67: Detection accuracy of SVD-Log with the original dimension of 100 in Spatial and Frequency Domains

Original	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
100-D	73.33	83.87	78.60	73.33	70.97	72.15

Table 5.68: Detection accuracy of SVD-Log with Kernel PCA in Frequency Domain

KPCA	Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)
10-D	**96.67**	**93.55**	**95.11**
20-D	93.33	83.87	88.60
30-D	86.67	80.65	83.66
40-D	96.67	80.65	88.66
50-D	90.00	80.65	85.32
60-D	96.67	77.42	87.04
70-D	96.67	80.65	88.66
80-D	90.00	100.00	95.00
90-D	86.67	100.00	93.33

5.4.12 Classification Performance of SVD-CN (Cb Channel)

However, the detection performance of Cb channel for SVD-CN is different from that for SVD-Log. Table 5.70 shows an incredible increase in classification rates of features extracted in Cb channel, which reaches a peak rate of 98.38% with 10 dimensions. The results indicate that Cb channel has much more information than R, G, B, and Y channels able to be captured by SVD-CN.

Table 5.69: Detection accuracy of SVD-CN with the original dimension of 100 in Spatial and Frequency Domains

Original	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
100-D	40.00	75.76	57.88	83.33	67.74	75.54

Table 5.70: Detection accuracy of SVD-CN with Kernel PCA in different dimensions in Spatial and Frequency Domains

KPCA	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
10-D	90.00	100.00	95.00	75.00	90.91	82.95
20-D	**100.00**	**96.97**	**98.48**	80.00	93.94	86.97
30-D	100.00	93.94	96.97	60.00	93.94	76.97
40-D	100.00	93.94	96.97	75.00	81.82	78.41
50-D	100.00	93.94	96.97	80.00	87.88	83.94
60-D	100.00	93.94	96.97	80.00	84.85	82.42
70-D	95.00	96.97	95.98	80.00	90.91	85.45
80-D	100.00	78.79	89.39	90.00	75.76	82.88
90-D	100.00	81.82	90.91	85.00	75.76	80.38

A comparative view of the classification performance for both methods is presented in Figure 5.17, which shows the considerable detection accuracies achieved by both proposed methods.

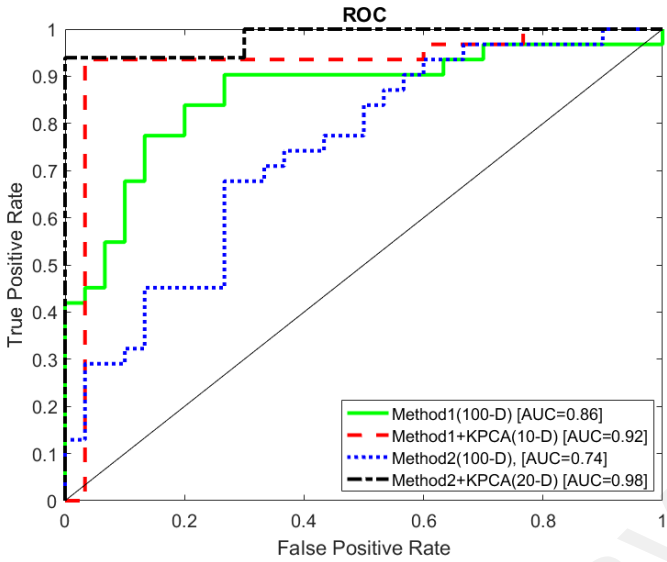


Figure 5.17: Comparison of Detection Performance between features in 100-D, Kernel PCA in 10-D, and Kernel PCA in 20-D using SVD-Log & SVD-CN (Cb Channel)

5.4.13 Classification Performance of SVD-Log (Cr Channel)

The last evaluated channel is Cr channel. Table 5.72 shows a classification performance of 96.67% obtained from Cr channel using SVD-Log with 10 and 20 dimensions in DCT. The results indicate that Cr channel is superior in SVD-Log compared to Y and Cb channels (95% and 95.11%).

Table 5.71: Detection accuracy of SVD-Log with the original dimension of 100 in Spatial and Frequency Domains

Original	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
100-D	86.67	77.42	82.04	76.67	80.65	78.66

Table 5.72: Detection accuracy of SVD-Log with Kernel PCA in Frequency Domain

KPCA	Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)
10-D	**93.33**	**100.00**	**96.67**
20-D	**93.33**	**100.00**	**96.67**
30-D	90.00	100.00	95.00
40-D	93.33	93.55	93.44
50-D	96.67	83.87	90.27
60-D	93.33	87.10	90.22
70-D	93.33	87.10	90.22
80-D	93.33	83.87	88.60
90-D	96.67	77.42	87.04

5.4.14 Classification Performance of SVD-CN (Cr Channel)

The features extracted from Cr channel using SVD-CN are also evaluated. Tables 5.73 and 5.74 demonstrate the classification performance of the features in Cr channel. There is a remarkable increase in detection rates to more than 93.94%, as shown in Table 5.74, which reaches a maximum score of 100% for 90 dimensions. It also verifies that SVD-CN is more sensitive to chroma channels than the other ones investigated, since both Cb and Cr channels have the best accuracies (100% and 98.48%) among the other evaluated channels.

Table 5.73: Detection accuracy of SVD-CN with the original dimension of 100 in Spatial and Frequency Domains

Original	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
100-D	65.00	75.76	70.38	70.00	83.87	76.94

Table 5.74: Detection accuracy of SVD-CN with Kernel PCA in different dimensions in Spatial and Frequency Domains

KPCA	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
10-D	100.00	93.94	96.97	93.33	96.77	95.05
20-D	100.00	93.94	96.97	93.33	96.77	95.05
30-D	100.00	93.94	96.97	95.00	93.94	94.47
40-D	100.00	90.91	95.45	85.00	87.88	86.44
50-D	100.00	87.88	93.94	80.00	93.94	86.97
60-D	100.00	93.94	96.97	90.00	83.87	86.94
70-D	100.00	93.94	96.97	75.00	96.97	85.98
80-D	100.00	93.94	96.97	86.67	77.42	82.04
90-D	**100.00**	**100.00**	**100.00**	65.00	100.00	82.50

Figure 5.18 indicates the ROC curves for the best detection rates of Cr channel in both proposed methods. Classification performance of features with 90 dimensions, as extracted from SVD-CN, is the best compared to other ROC curves.

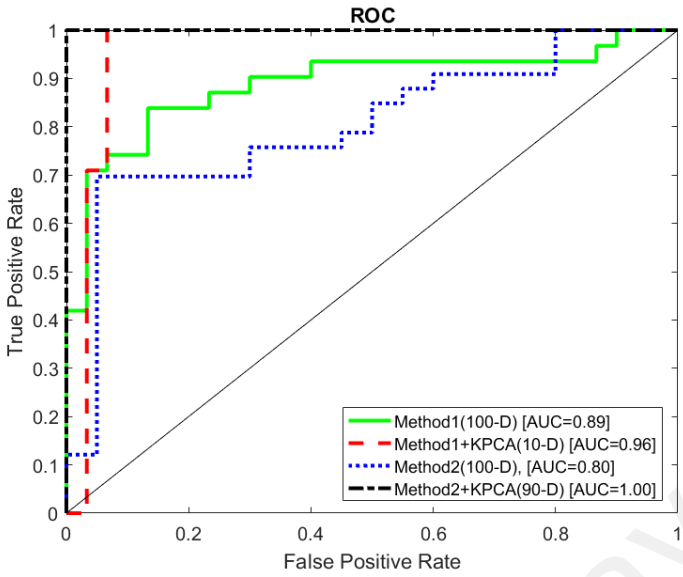


Figure 5.18: Comparison of Detection Performance between features in 100-D, Kernel PCA in 10-D, and Kernel PCA in 90-D using SVD-Log & SVD-CN (Cr Channel)

5.4.15 Classification Performance of SVD-Log (RGB color model)

Similar to the CASIA image dataset, the combination of R, G, and B channels are also evaluated for both proposed methods. Tables 5.75 and 5.76 show the detection rates for SVD-Log in true color images (RGB). According to Table 5.76, the accuracies range between 86.99% and 95.05% for 90 and 30 dimensions respectively. The results from the individual channels and the combined one show that the best detection performance is for Greyscale images, with 100%.

Table 5.75: Detection accuracy of SVD-Log with the original dimension of 100 in Spatial and Frequency Domains

Original	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
100-D	63.33	64.52	63.92	73.33	74.19	73.76

Table 5.76: Detection accuracy of SVD-Log with Kernel PCA in Frequency Domain

KPCA	Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)
10-D	90.00	100.00	95.00
20-D	90.00	96.77	93.39
30-D	**93.33**	**96.77**	**95.05**
40-D	93.33	90.32	91.83
50-D	93.33	90.32	91.83
60-D	93.33	87.10	90.22
70-D	93.33	87.10	90.22
80-D	93.33	80.65	86.99
90-D	93.33	80.65	86.99

5.4.16 Classification Performance of SVD-CN (RGB Color model)

The combined RGB channels are also evaluated for SVD-CN. Table 5.78 shows a considerable accuracy of 96.97% achieved from the features in the frequency domain with 30 dimensions. Unlike SVD-Log, the combination of R, G, and B channels effectively increases the detection performance in SVD-CN. It verifies that in comparison between Greyscale, R, G, B channels and their combination, SVD-CN can better capture the splicing traces in the combination and greyscale forms (96.97% and 96.72%) respectively.

Table 5.77: Detection accuracy of SVD-CN with the original dimension of 100 in Spatial and Frequency Domains

Original	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
100-D	60.00	69.70	64.85	63.33	74.19	68.76

Table 5.78: Detection accuracy of SVD-CN with Kernel PCA in different dimensions in Spatial and Frequency Domains

KPCA	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
10-D	85.00	66.67	75.83	85.00	93.94	89.47
20-D	90.00	83.87	86.94	95.00	93.94	94.47
30-D	93.33	87.10	90.22	**100.00**	**93.94**	**96.97**
40-D	90.00	83.87	86.94	95.00	90.91	92.95
50-D	86.67	80.65	83.66	100.00	87.88	93.94
60-D	90.00	87.10	88.55	95.00	87.88	91.44
70-D	90.00	90.32	90.16	95.00	84.85	89.92
80-D	90.00	93.55	91.77	90.00	87.88	88.94
90-D	83.33	96.77	90.05	90.00	84.85	87.42

Therefore, the best detection performance for the RGB model is for the features with 30 dimensions extracted from SVD-Log, which also is depicted in Figure 5.19 based on the ROC curve.

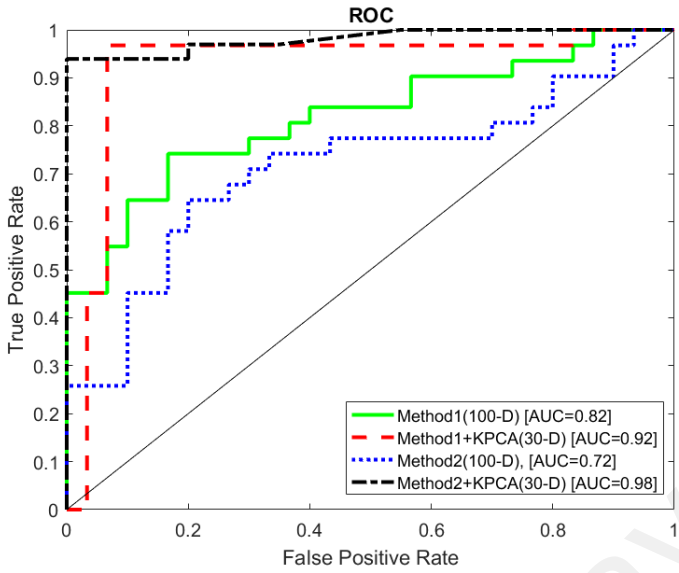


Figure 5.19: Comparison of Detection Performance between features in 100-D, Kernel PCA in 30-D, and Kernel PCA in 30-D using SVD-Log & SVD-CN (RGB Color Model)

5.4.17 Classification Performance of SVD-Log (YCbCr Color Model)

The last channel evaluated is the combination of Y, Cb, and Cr channels. Table 5.80 shows that a considerable detection rate is achieved in 20 dimensions, with 96.77% in SVD-Log; this is almost the same as the Cr channel but higher than the Y and Cb channels individually. Therefore, it can be observed that the combination of the channels can also effectively reflect the splicing artifacts in the third image dataset for SVD-Log.

Table 5.79: Detection accuracy of SVD-Log with the original dimension of 100 in Spatial and Frequency Domains

Original	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
100-D	83.33	77.42	80.38	80.00	70.97	75.48

Table 5.80: Detection accuracy of SVD-Log with Kernel PCA in Frequency Domain

KPCA	Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)
10-D	93.33	100.00	96.67
20-D	**100.00**	**93.55**	**96.77**
30-D	96.67	93.55	95.11
40-D	93.33	93.55	93.44
50-D	**100.00**	**93.55**	**96.77**
60-D	100.00	87.10	93.55
70-D	100.00	87.10	93.55
80-D	100.00	83.87	91.94
90-D	100.00	83.87	91.94

5.4.18 Classification Performance of SVD-CN (YCbCr Color Model)

Similarly, Table 5.81 demonstrates a detection rate of 96.72% with 10 dimensions for the combination of luma and chroma channels utilizing SVD-CN. However, compared to the individual channels in the YCbCr color model, this rate is not the best (100 and 98.48 for chroma channels) but is still an acceptable detection performance. In addition, the overall results show that SVD-CN is more sensitive to chroma channels than the other ones mentioned.

Table 5.81: Detection accuracy of SVD-CN with the original dimension of 100 in Spatial and Frequency Domains

Original	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
100-D	65.00	84.85	74.92	96.67	35.48	66.08

Table 5.82: Detection accuracy of SVD-CN with Kernel PCA in different dimensions in Spatial and Frequency Domains

KPCA	Spatial Domain			Frequency Domain (DCT)		
	TNR(%)	TPR(%)	ACC(%)	TNR(%)	TPR(%)	ACC(%)
10-D	90.00	96.77	93.39	**96.67**	**96.77**	**96.72**
20-D	90.00	93.55	91.77	96.67	90.32	93.49
30-D	90.00	80.65	85.32	96.67	83.87	90.27
40-D	90.00	87.10	88.55	86.67	80.65	83.66
50-D	90.00	93.55	91.77	96.67	77.42	87.04
60-D	96.67	87.10	91.88	93.33	80.65	86.99
70-D	86.67	100.00	93.33	100.00	74.19	87.10
80-D	96.67	90.32	93.49	95.00	72.73	83.86
90-D	93.33	80.65	86.99	95.00	75.76	85.38

Figure 5.29 indicates the ROC curves for best detection performance of both methods in the combination of Y, Cb, and Cr channels. It is observed from the curves that both methods have remarkable detection accuracy in the combination of mentioned channels, since they are very far from the diagonal of ROC space.

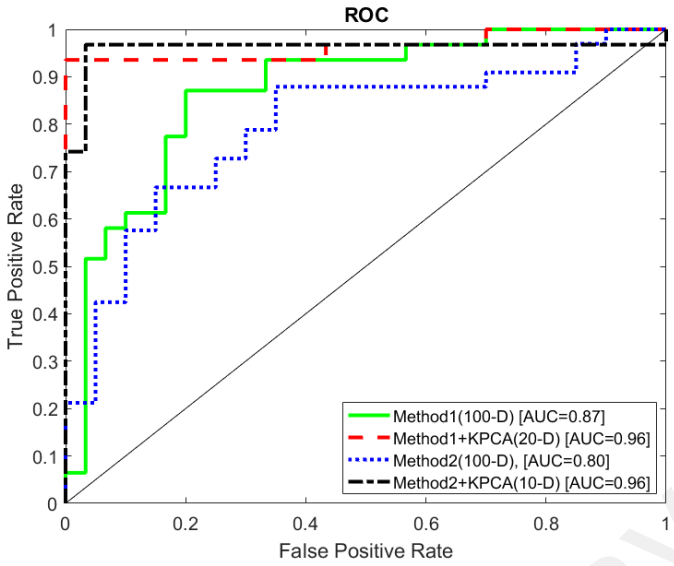


Figure 5.20: Comparison of Detection Performance between features in 100-D, Kernel PCA in 20-D, and Kernel PCA in 10-D using SVD-Log & SVD-CN (YCbCr Color Model)

5.5 Comparison between detection performance of SVD-Log and SVD-CN

In this section, a comparison between the best detection performance of both proposed methods in the examined channels for every image dataset is presented. Figure 5.21 shows the detection accuracy for SVD-Log and SVD-CN in the DVMM v1 image dataset. Accordingly, both SVD-Log and SVD-CN demonstrate considerable detection rates (98.78 and 97.15) in this image dataset, with a superiority for SVD-Log.

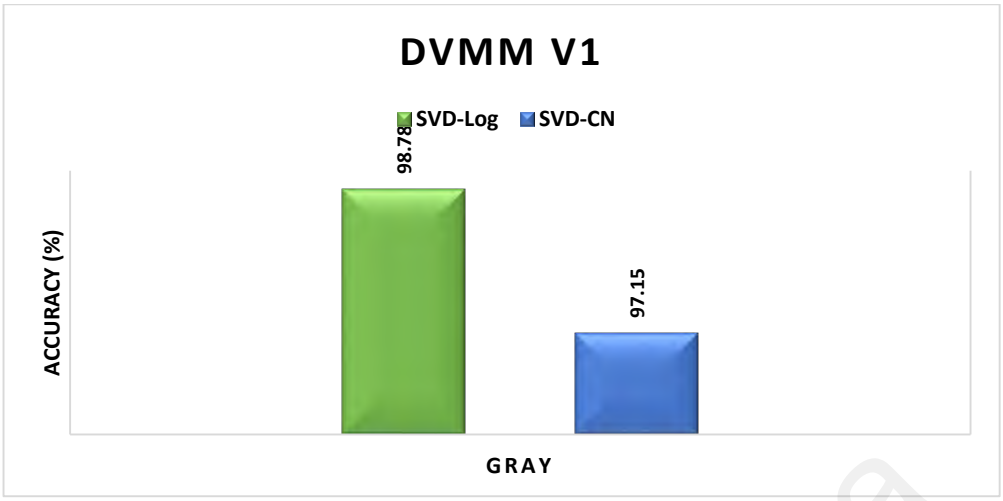


Figure 5.21: Comparison between best detection accuracies of SVD-Log and SVD-CN in DVMM v1 image dataset

Figure 5.22 illustrates the trend of detection rates for SVD-Log and SVD-CN in the CASIA image dataset in the examined channels. Generally, SVD-CN compared to SVD-Log has better performance in all channels except the Luma channels.

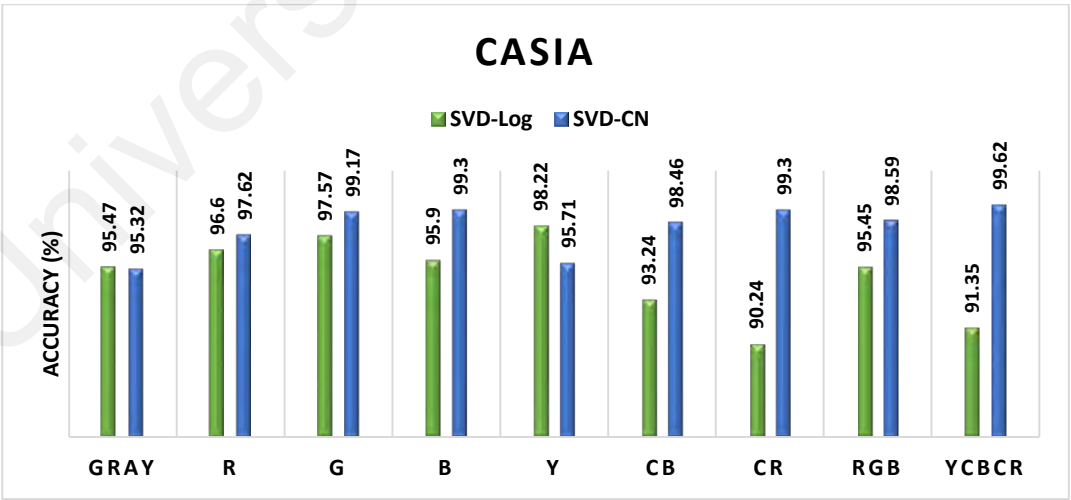


Figure 5.22: Comparison between best detection accuracies of SVD-Log and SVD-CN in CASIA image dataset

However, Figure 5.23 indicates different trends for both proposed methods in DVMM v2 image dataset. SVD-Log shows higher detection accuracy in the Grey, R, and G channels while SVD-CN achieves higher accuracy in YCbCr channels. Generally, SVD-Log has better performance in Grey channels compared to SVD-CN. However, the overall performance of SVD-CN is higher than that of SVD-Log, which verifies that condition number can effectively reflect the splicing artifacts in the tested image datasets.

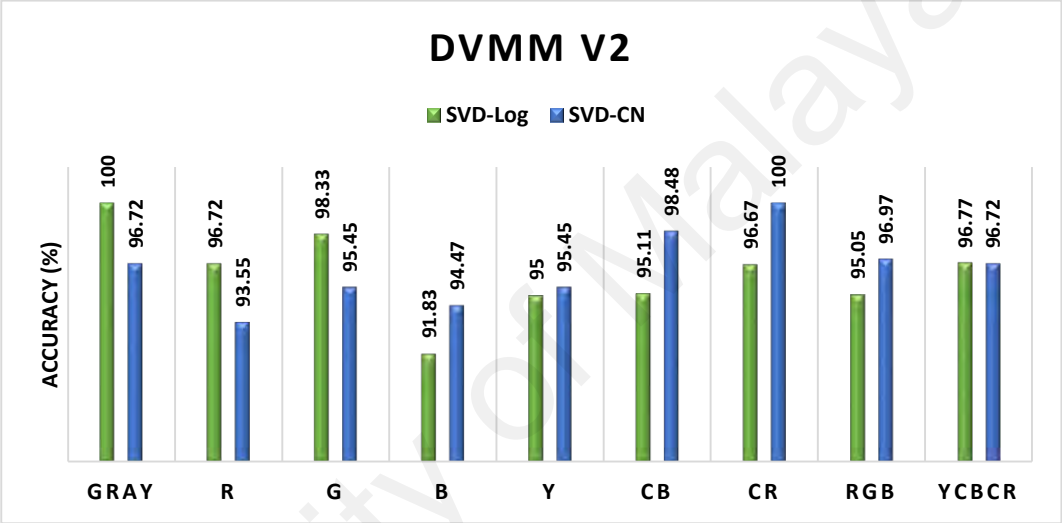


Figure 5.23: Comparison between best detection accuracies of SVD-Log and SVD-CN in DVMM v2 image dataset

5.6 Comparison with Conventional Image Splicing Methods

A comparison of some state-of-the-art image splicing detection methods was conducted for a comprehensive evaluation of the entire systems. Tables 5.83-5.85 show a comparison between the existing image splicing detection methods and our proposed schemes based on dimensionality, TPR, TNR and accuracy in different image datasets including DVMM v1, DVMM v2 and CASIA. It is observed from the tables that the best detection performance in DVMM v1 is for SVD-Log and SVD-CN with 98.78% in 50

dimensions and 97.15% in 90 dimensions respectively. Thus SVD-Log achieved the best detection accuracy among all of the recently developed techniques.

The third and the last image dataset is DVMM v2. The results indicate some valuable detection rates (95.13 and 96.39) recorded for DVMM v2, which reaches the maximum amount of 100% for proposed methods 1 & 2 in 10 and 90 dimensions, respectively.

Generally, it can be observed from the comparison tables that the SVD-based proposed methods achieved the best detection performances with low enough dimensionality, compared to other recently developed image splicing detection techniques. Thus, SVD can be effectively applied in the detection of spliced images.

Table 5.83: Comparison of the proposed methods with the existing image splicing detection methods based on DVMM v1

Methods	Image Dataset	Dimensionality	TPR (%)	TNR (%)	Accuracy (%)
Shi et al. (Shi et al., 2007b)	DVMM v1	266	92.76	91.00	91.87
Dong et al. (Dong et al., 2009)	DVMM v1	163	83.23	85.53	84.36
Anusudha et al. (Anusudha et al., 2010)	DVMM v1	198	91.78	91.64	91.7
He et al. (He et al., 2012)	DVMM v1	100	93.28	93.83	93.55
Rabha et al. (Ibrahim et al., 2015)	DVMM v1	40	92.31	91.45	91.88
SVD-Log	DVMM v1	50	97.56	100	98.78
SVD-CN	DVMM v1	90	97.53	96.75	97.15

Table 5.84: Continued, comparison of the proposed methods with the existing image splicing detection methods based on CASIA

Methods	Image Dataset	Dimensionality	TPR (%)	TNR (%)	Accuracy (%)
Zhao et al. (Zhao et al., 2011)	CASIA	60	91.80	97.10	94.7
Muhammad. G et al. (Muhammad et al., 2014)	CASIA	475	95.15	93.91	94.89
Hakimi. F et al. (Hakimi et al., 2015)	CASIA				97.21
SVD-Log	CASIA	40	97.74	98.70	98.22
SVD-CN	CASIA	10	99.25	100	99.62

Table 5.85: Continued, comparison of the proposed methods with the existing image splicing detection methods based on DVMM v2

Methods	Image Dataset	Dimensionality	TPR (%)	TNR (%)	Accuracy (%)
Zhao et al. (Zhao et al., 2011)	DVMM v2	60	80.20	89.80	85
Muhammad. G et al. (Muhammad et al., 2014)	DVMM v2	359	97.92	95.53	96.39
Hakimi. F et al. (Hakimi et al., 2015)	DVMM v2				95.13
SVD-Log	DVMM v2	10	100	100	100
SVD-CN	DVMM v2	90	100	100	100

5.7 Summary

Throughout this chapter, the two proposed SVD-based detection methods evaluated by applying different available image datasets (DVMM v1, DVMM v2, and CASIA) in spatial and frequency i.e. DCT domains. For the colored image datasets, Gray-scale, R, G, B, Y, Cb, and Cr channels are considered separately. These conditions have been provided (different domains and color channels) to investigate their effectiveness on the detection performance of the proposed detection schemes.

However, the observed detection results indicate that the original extracted features are not high enough. For satisfactory results, two dimension reduction methods namely, PCA and Kernel PCA have been applied to reduce feature dimensionality by eliminating redundant features and maintaining important dimensions in the feature vector. To evaluate the efficiency of the mentioned dimension reduction methods on the detection performance, the dimensionality of the reduced feature vector was set to different values (10, 20, 30 … 90-D) to find out which dimensionality is the best in that feature set.

The detection results show that after application of Kernel PCA, remarkable accuracies have been obtained from both proposed methods in different color channels and dimensionality. Furthermore, the overall results demonstrate that the proposed image splicing detection schemes outperform the recently developed techniques in low dimensionality and high detection rate.

CHAPTER 6: CONCLUSION AND FUTURE WORK

6.1 Introduction

The last chapter concludes and summarizes the previous chapters and reviews the entire study. This chapter is organized in three different sections. The first section refers to the research findings and achievements of this research and explains how the mentioned objectives were met throughout the research. The second section is about the contributions of the research, what this study has achieved. Lastly, the final section covers future works of this study for those who are interested in the topic and want to do further research in this field.

6.2 Research Findings and Achievements

Digital image technology has been rapidly developing in recent decades due to its special place in people's lives. However, various digital image processing tools have been created to easily modify and improve captured digital images. However, these tools are not only applied for ethical purposes. There are many instances that show how digital images are forged and fake ones created. Therefore, the authentication of digital images has become a substantial concern for different organizations.

There are several approaches (image copy-move, image splicing, etc.) that can be used to modify a digital image and create a tampered one. All these approaches leave different kind of traces, such as statistical ones, in the forged images. It is important for researchers to find a way that captures such traces in the tampered images efficiently and detects them as accurately as possible. Various aspects should be considered when developing new image forgery detection methods. These aspects vary for different image forgery techniques. For example, for image splicing detection techniques, it is essential to detect

the spliced images accurately and within a reasonable computational complexity. It is also important to detect the images in different formats, sizes and color models.

The main goal of this study is to develop image splicing detection methods to accurately detect and distinguish authentic images from spliced ones. In this research, two SVD-based image splicing detection methods (SVD-Log and SVD-CN) are proposed, designed, and implemented with remarkably high detection performance. The study was conducted in line with its objectives stated in section 1.4. The following is the explanation how those objectives are achieved throughout the research:

i. **To investigate different image splicing detection methods:** The first objective of this research is in evaluating and comparing recently developed image splicing detection algorithms. This was carried out in chapter 2 with the "Literature Review". In that chapter, various concepts were presented and several works related to this research were investigated and analyzed. Afterwards, all of the methods were summarized in a table to give the reader a comprehensive understanding of different related techniques.

ii. **To select appropriate image preprocessing methods to prepare the image for feature extraction:** One important phase in designing image splicing detection algorithms is to prepare the image to better reflect the splicing traces left in the image. The image preprocessing methods are selected in such a way as to be compatible with the feature extraction technique. Section 4.2 presents and discusses the image preprocessing techniques applied in this research. Accordingly, the techniques applied include multi-blocking, image domain transformation and color models.

iii. **To propose two splicing detection methods based on Singular Value Decomposition (SVD) with low dimensionality:** The most important part of this research is the design and development of efficient image splicing detection frameworks with the capability to capture the splicing artifacts inside the images with higher detection accuracy. To do this, the Singular Value Decomposition concept has been applied to develop the splicing detection methods. Thus, two different splicing detection methods (namely SVD-Log and SVD-CN) were proposed to detect the spliced images from the original ones with a high enough detection rate and with reasonable computation complexity. All the steps, flowcharts and pseudo-codes for the proposed techniques are presented in Section 4.3.

iv. **To select an appropriate feature selector to improve the classification performance:** Since the last phase of the proposed methods is classification, the extracted features from the proposed schemes should be prepared for this phase. Thus, one more phase is required, which is called the feature selection phase in this paper. In this phase, the features are prepared for classification by removing some redundancy from the feature set. The issue in this phase is to select an appropriate feature selector to efficiently improve classification performance. The best dimension reduction method has been selected as Kernel PCA, which achieves the best results in the proposed methods (see Section 4.4).

v. **To test and evaluate the two proposed methods by measuring the detection rate using three standard image datasets DVMM v1 & v2, and CASIA:** All the experimental results and their analyses have been presented in chapter 5. The proposed methods were tested on three available datasets in different domains and color models. The obtained results were presented based on ROC curves, AUC,

TPR, TNR, and their average (i.e. detection accuracy). At the end, a comparison with other state-of-the-art splicing techniques has been provided to give a comprehensive evaluation of both entire systems. The comparison states that our SVD-based proposed schemes outperform the recently developed methods over all three datasets.

Images, in comparison with other media, offer a natural and influential communication approach for people, since often people require no specific training to realize the contents of an image. However, the ability to distinguish original images from tampered ones is of high importance. Currently, digital images already have been widely employed for reporting the news, studies on insurance claims, criminal or forensic studies, national intelligence investigations and legal proceedings. Thus, this kind of image forgery detection can strongly influence the application domain mentioned above.

A good image splicing detection algorithm should be able to detect spliced images with a high enough detection rate and within an acceptable computational complexity. These two aspects are essential in each image splicing detection method. Since our proposed methods provide both of these aspects, it can be effectively applied in many commercial and governmental organizations for security purposes. The organizations are only required to provide a complete training model and then the digital images can be tested, based on the training model, for authenticity.

6.3 Conclusion

Image splicing is a common process that can cause inconsistencies in many features, such as an abnormally sharp transient at the splicing edges. These inconsistencies are used to detect the forged image. Several splicing detection techniques have been proposed that

suffer from various problems, such as high-dimensionality, low detection rate and high computational complexity. In order to overcome the aforementioned problems, it is necessary to have an efficient image splicing detection method with the ability to detect spliced images accurately and within reasonable computation complexity. Thus, two splicing detection methods (SVD-Log and SVD-CN) are proposed and developed to help solve the above-mentioned problems and achieve the ultimate goal. Based on the existing splicing detection problems and the experimental results conducted in chapter 5, the following statements make up the study's conclusions:

- In this study, various challenges encountered by current splicing detection methods are discussed. It is difficult for recently proposed methods to overcome all the issues in splicing detection methods, since there are several factors which affect the detection rate, such as color channels, transformation domains, etc.

- The performance of the two proposed methods has been widely analyzed (based on detection accuracy) in the experimental results, and their advantages and drawbacks discussed. The proposed methods showed different behaviors in different image datasets, color channels, and transformation domains. In the DVMM v1 image dataset, SVD-Log had a better detection result, while in the CASIA image dataset, the best detection rate was for SVD-CN. For the DVMM v2, both methods show remarkable detection rates, but in different color channels. In terms of feature extraction time, SVD-CN takes less time to extract the features, since it is based on the condition number instead of the Logarithm function. This statement answers research question 5.

- To develop a new image splicing detection method with low-dimensionality and high detection rates, several factors should be considered. In this research different properties of SVD in the images have been presented and discussed (see section 2.6). Based on these properties, SVD can effectively capture the splicing artifacts left in the images. Thus, the SVD-based splicing detection methods have been designed, developed and tested to meet the ultimate goal of this research. This covers the answer to research questions 1-3.

- According to our previous research (Moghaddasi, Jalab, Md Noor, et al., 2014) on improving classification performance by removing redundancies from the extracted features, many of the extracted features are not in an appropriate format for classification and thus it is essential to prepare the features before the classification is performed. Another point is that, since the features have different natures such as linearity and nonlinearity, the applied feature selector must be suitable for the extracted features. To do this, different dimension reduction methods are tested on the extracted features and Kernel PCA was the best, as illustrated in the experimental results. This item answers research question 4.

- Generally, it can be concluded that the proposed methods have the following properties: 1) In the first image dataset (DVMM v1, which is a grayscale image dataset), both methods have good detection rates (98.78% and 97.15%) with SVD-Log showing superior rates. 2) In the CASIA image dataset, both methods display their best detection rates in YCbCr channels (98.22% and 99.62%), which indicates that the proposed methods are more sensitive to luma and chroma channels than R, G, and B channels. 3) In the DVMM v2 dataset, SVD-Log has the best possible detection rate of 100% in grayscale images and

SVD-CN achieves this accuracy in Cr channel, which shows the sensitivity of this method to chroma channels.

6.4 Future Work and Directions

According to the conclusions and contributions provided in this chapter, there are some recommendations for researchers who are interested in image forgery detection schema. The future directions of this research can be conducted as follows:

- Enhancing the proposed methods to locate the spliced regions inside the spliced images.
- Modifying the methods to detect other types of forgeries, such as copy-move, retouching, etc.
- Improving the overall performance of the methods to be independent of the separate channels and feature selectors.

REFERENCES

Al-haj, A., & Abu-errub, A. (2008). Performance Optimization of Discrete Wavelets Transform Based Image Watermarking Using Genetic Algorithms 1. *Journal of Computer Science and Technology, 4*(10), 7.

Alahmadi, A. A., Hussain, M., Aboalsamh, H., Muhammad, G., & Bebis, G. (2013). *Splicing image forgery detection based on DCT and Local Binary Pattern.* Paper presented at the Global Conference on Signal and Information Processing (GlobalSIP), 2013 IEEE.

Albregtsen, F. (2008). Statistical texture measures computed from gray level coocurrence matrices. *Image processing laboratory, department of informatics, university of oslo, 5*.

Amsberry, C. (1989). Alterations of photos raise host of legal, ethical issues. *The Wall Street Journal, 1*, 26-89.

Andrews, H., & Patterson, C. (1976). Singular value decompositions and digital image processing. *IEEE Transactions on Acoustics, Speech, and Signal Processing, 24*(1), 26-53.

Anthony, G., Greg, H., & Tshilidzi, M. (2007). Classification of images using support vector machines. *ArXiv e-print*.

Anusudha, K., Koshie, S. A., Ganesh, S. S., & Mohanaprasad, K. (2010). Image Splicing Detection involving Moment-based Feature Extraction and Classification using Artificial Neural Networks. *Aceee International Journal on signal & Image processing, 1*(3).

Baker, K. (2005). Singular value decomposition tutorial. *The Ohio State University, 24*.

Banham, M. R., & Katsaggelos, A. K. (1996). Spatially adaptive wavelet-based multiscale image restoration. *Image Processing, IEEE Transactions on, 5*(4), 619-634.

Bengio, Y., Delalleau, O., & Le Roux, N. (2005). The curse of dimensionality for local kernel machines. *Techn. Rep, 1258*.

Bigun, J., Granlund, G. H., & Wiklund, J. (1991). Multidimensional orientation estimation with applications to texture analysis and optical flow. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 13*(8), 775-790.

Birajdar, G. K., & Mankar, V. H. (2013). Digital image forgery detection using passive techniques: A survey. *Digital Investigation, 10*(3), 226-245.

Böhme, R., Freiling, F. C., Gloe, T., & Kirchner, M. (2009). Multimedia forensics is not computer forensics *Computational Forensics* (pp. 90-103): Springer.

Bregler, C., Covell, M., & Slaney, M. (1997). *Video rewrite: Driving visual speech with audio.* Paper presented at the Proceedings of the 24th annual conference on Computer graphics and interactive techniques.

Caldelli, R., Amerini, I., Picchioni, F., De Rosa, A., & Uccheddu, F. (2009). Multimedia forensic techniques for acquisition device identification and digital image authentication. *Handbook of research on computational forensics, digital crime and investigation: methods and solutions'(Information Science Reference, IGI Global, 2009)*, 130-154.

Campbell, C. (2001). An introduction to kernel methods, Radial basis function networks 1: recent developments in theory and applications: Physica Verlag Rudolf Liebing KG, Vienna, Austria.

Cao, L. (2006). Singular value decomposition applied to digital image processing. *Division of Computing Studies, Arizona State University Polytechnic Campus, Mesa, Arizona State University polytechnic Campus*.

Casey, E., Blitz, A., & Steuart, C. (2014). Digital Evidence and Computer Crime: Academic Press.

Chamlawi, R., Khan, A., & Usman, I. (2010). Authentication and recovery of images using multiple watermarks. *Computers & Electrical Engineering, 36*(3), 578-584.

Chang, C. C., & Lin, C. J. (2011). LIBSVM: A Library for Support Vector Machines. *Acm Transactions on Intelligent Systems and Technology, 2*(3), 1-27.

Chen, C., & Shi, Y. Q. (2008). *JPEG image steganalysis utilizing both intrablock and interblock correlations.* Paper presented at the IEEE International Symposium on Circuits and Systems.

Chen, W., Shi, Y. Q., & Su, W. (2007). *Image splicing detection using 2-D phase congruency and statistical moments of characteristic function.* Paper presented at the Society of photo-optical instrumentation engineers (SPIE) conference series.

Chen, Y.-S., & Wang, R.-Z. (2011). Reversible authentication and cross-recovery of images using (t, n)-threshold and modified-RCM watermarking. *Optics Communications, 284*(12), 2711-2719.

Cline, A. K., & Dhillon, I. S. (2006). Computation of the singular value decomposition. *Handbook of linear algebra*, 45.41-45.13.

Cox, I. J., Miller, M. L., Bloom, J. A., & Honsinger, C. (2002). *Digital watermarking* (Vol. 1558607145): Springer.

Craver, S., Wu, M., Liu, B., Stubblefield, A., Swartzlander, B., Wallach, D. S., . . . Felten, E. W. (2001). *Reading Between the Lines: Lessons from the SDMI Challenge.* Paper presented at the USENIX Security Symposium.

Cristianini, N., & Shawe-Taylor, J. (2000). *An introduction to support vector machines and other kernel-based learning methods*: Cambridge university press.

Curtis, C. (1994). Sirius Video Programming and Configuration Guide (pp. 211): Silicon Graphics, Inc.

Daubechies, I. (1990). The wavelet transform, time-frequency localization and signal analysis. *IEEE Transactions on Information Theory, 36*(5), 961-1005.

Daugman, J. G. (1993). High confidence visual recognition of persons by a test of statistical independence. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 15*(11), 1148-1161.

Dictionary, O. E. (2004). Oxford English dictionary online. *Mount Royal College Lib., Calgary, 14*.

Dong, J., & Tan, T. (2008). *Blind image steganalysis based on run-length histogram analysis.* Paper presented at the ICIP.

Dong, J., & Wang, W. (2011). *CASIA tampered image detection evaluation database* [Image]. Retrieved from: http://forensics.idealtest.org/

Dong, J., Wang, W., Tan, T., & Shi, Y. Q. (2008). *Run-length and edge statistics based approach for image splicing detection.* Paper presented at the International Workshop on Digital Watermarking.

Dong, J., Wang, W., Tan, T., & Shi, Y. Q. (2009). Run-length and edge statistics based approach for image splicing detection *Digital Watermarking* (pp. 76-87): Springer.

Fang, Z., Wang, S., & Zhang, X. (2010). *Image splicing detection using color edge inconsistency.* Paper presented at the 2010 International Conference on Multimedia Information Networking and Security.

Farid, H. (2006a). Digital doctoring: how to tell the real from the fake. *Significance, 3*(4), 162-166.

Farid, H. (2006b). *Exposing digital forgeries in scientific images.* Paper presented at the Proceedings of the 8th workshop on Multimedia and security.

Farid, H. (2008). Digital image forensics. *Scientific American, 298*(6), 66-71.

Farid, H. (2009). Image forgery detection--A survey.

Fodor, I. K. (2002). A survey of dimension reduction techniques: Technical Report UCRL-ID-148494, Lawrence Livermore National Laboratory.

Foody, G. M., & Mathur, A. (2004a). A relative evaluation of multiclass image classification by support vector machines. *IEEE Transactions on Geoscience and Remote Sensing, 42*(6), 1335-1343.

Foody, G. M., & Mathur, A. (2004b). Toward intelligent training of supervised image classifications: directing training data acquisition for SVM classification. *Remote Sensing of Environment, 93*(1), 107-117.

Friedman, G. L. (1993). The trustworthy digital camera: Restoring credibility to the photographic image. *IEEE Transactions on Consumer Electronics, 39*(4), 905-910.

Fu-bing, W. W.-s. C., & Jing-yu, Y. (2005). A Method of Feature Extraction Based on SVD. *Journal of Electronics and Information Technology, 2*, 032.

Fu, D., Shi, Y. Q., & Su, W. (2006). Detection of image splicing based on hilbert-huang transform and moments of characteristic functions with wavelet decomposition *Digital Watermarking* (pp. 177-187): Springer.

Galloway, M. M. (1975). Texture analysis using gray level run lengths. *Computer graphics and image processing, 4*(2), 172-179.

Ganic, E., Zubair, N., & Eskicioglu, A. M. (2003). *An optimal watermarking scheme based on singular value decomposition.* Paper presented at the Proceedings of the IASTED International Conference on Communication, Network, and Information Security.

Ghodsi, A. (2006). Dimensionality reduction a short tutorial. *Department of Statistics and Actuarial Science, Univ. of Waterloo, Ontario, Canada*.

Gonzalez, R. C. (2010). Digital image processing using MATLAB® Rafael C. *Gonzalez, Richard E. Woods, Steven L. Eddins. 2nd ed: New Delhi: McGraw-Hill*.

Gorodetski, V. I., Popyack, L. J., Samoilov, V., & Skormin, V. A. (2001). *SVD-based approach to transparent embedding data into digital images.* Paper presented at the International Workshop on Mathematical Methods, Models, and Architectures for Network Security.

Granty, R. E. J., Aditya, T., & Madhu, S. S. (2010). *Survey on passive methods of image tampering detection.* Paper presented at the Proceedings of the International Conference on Communication and Computational Intelligence India.

Group, C. D. E. S. F. W. (2006). Standardizing digital evidence storage. *Communications of the ACM, 49*(2), 67-68.

Gu, M., & Eisenstat, S. C. (1995). A divide-and-conquer algorithm for the bidiagonal SVD. *SIAM Journal on Matrix Analysis and Applications, 16*(1), 79-92.

Gu, Q., Zhu, L., & Cai, Z. (2009). *Evaluation measures of the classification performance of imbalanced data sets*. Paper presented at the International Symposium on Intelligence Computation and Applications.

Gul, G., & Kurugollu, F. (2010). SVD-based universal spatial domain image steganalysis. *IEEE Transactions on Information Forensics and Security, 5*(2), 349-353.

Günter, S., Schraudolph, N. N., & Vishwanathan, S. (2007). Fast iterative kernel principal component analysis. *Journal of Machine Learning Research, 8*(Aug), 1893-1918.

Hakimi, F., Hariri, M., & GharehBaghi, F. (2015). *Image splicing forgery detection using local binary pattern and discrete wavelet transform.* Paper presented at the 2015

2nd International Conference on Knowledge-Based Engineering and Innovation (KBEI).

Haralick, R. M., & Shanmugam, K. (1973). Textural features for image classification. *IEEE Transactions on systems, man, and cybernetics*(6), 610-621.

Hartung, F., & Kutter, M. (1999). Multimedia watermarking techniques. *Proceedings of IEEE, 87*(7), 1079-1107.

He, Z., Lu, W., Sun, W., & Huang, J. (2012). Digital image splicing detection based on Markov features in DCT and DWT domain. *Pattern Recognition, 45*(12), 4292-4299.

He, Z., Sun, W., Lu, W., & Lu, H. (2011). Digital image splicing detection based on approximate run length. *Pattern Recognition Letters, 32*(12), 1591-1597.

Holmes, M., Gray, A., & Isbell, C. (2007). *Fast SVD for large-scale matrices.* Paper presented at the Workshop on Efficient Machine Learning at NIPS.

Hsu, C.-W., Chang, C.-C., & Lin, C.-J. (2010). A practical guide to support vector classification. *citeseerx*, 1-12. doi: 10.1.1.224.4115

Huang, C., Davis, L., & Townshend, J. (2002). An assessment of support vector machines for land cover classification. *International Journal of remote sensing, 23*(4), 725-749.

Huang, N. E., Shen, Z., Long, S. R., Wu, M. C., Shih, H. H., Zheng, Q., . . . Liu, H. H. (1998). The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis. *Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences, 454*(1971), 903-995.

Ibrahim, R. W. (2011). On generalized Srivastava-Owa fractional operators in the unit disk. *Advances in Difference Equations, 2011*(1), 1.

Ibrahim, R. W., Moghaddasi, Z., Jalab, H. A., & Noor, R. M. (2015). Fractional differential texture descriptors based on the machado entropy for image splicing detection. *Entropy, 17*(7), 4775-4785.

Inman, K., & Rudin, N. (2000). *Principles and practice of criminalistics: the profession of forensic science*: CRC Press.

Inman, K., & Rudin, N. (2002). The origin of evidence. *Forensic science international, 126*(1), 11-16.

Jabid, T., Kabir, M. H., & Chae, O. (2010). *Gender classification using local directional pattern (LDP).* Paper presented at the 20th International Conference on Pattern Recognition (ICPR)

Jackson, J. E. A User's Guide to Principal Components. 1991. *John Willey &Sons, New York*.

Jähne, B. (2005). Digital Image Processing: Springer, Berlin.

Jalab, H. A., & Ibrahim, R. W. (2012). *Texture feature extraction based on fractional mask convolution with cesáro means for content-based image retrieval.* Paper presented at the Pacific Rim International Conference on Artificial Intelligence.

Jalab, H. A., & Ibrahim, R. W. (2013). Texture enhancement based on the Savitzky-Golay fractional differential operator. *Mathematical Problems in Engineering, 2013*.

Jalab, H. A., & Ibrahim, R. W. (2015). Fractional conway polynomials for image denoising with regularized fractional power parameters. *Journal of mathematical imaging and vision, 51*(3), 442-450.

Jin, R., Dong, Z., & Xi, L. (2009). *A Feature Extraction Method of Image Based on Dimentional Transformation and SVD Algorithm.* Paper presented at the Proceedings of the 3rd WSEAS International Conference on COMPUTER ENGINEERING and APPLICATIONS (CEA'09).

JING, X., GUO, Y., & YANG, J. (1999). Image preprocessing based on singular value and face recognition. *Information and Control*(2), 120-124.

Joachims, T. (1998). *Text categorization with support vector machines: Learning with many relevant features*: Springer.

Jolliffe, I. T. (1986). *Principal component analysis* (Vol. 487): Springer-Verlag New York.

Kakarala, R., & Ogunbona, P. O. (2001). Signal analysis using a multiresolution form of the singular value decomposition. *IEEE Transactions on Image Processing, 10*(5), 724-735.

Kamm, J. L. (1998). *SVD-based methods for signal and image restoration.* PhD thesis.

Katzenbeisser, S., & Petitcolas, F. (2000). *Information hiding techniques for steganography and digital watermarking*: Artech house.

Khayam, S. A. (2003). The discrete cosine transform (DCT): theory and application. *Michigan State University*.

Kirk, P. L. (1974). *Crime investigation* (J. I. Thornton Ed.): Wiley.

Kohl, M. (2012). Performance measures in binary classification. *International Journal of Statistics in Medical Research, 1*(1), 79-81.

Konda, T., & Nakamura, Y. (2009). A new algorithm for singular value decomposition and its parallelization. *Parallel Computing, 35*(6), 331-344.

Konstantinides, K., Natarajan, B., & Yovanof, G. S. (1996). Noise estimation and filtering using block-based singular value decomposition. *IEEE transactions on image processing: a publication of the IEEE Signal Processing Society, 6*(3), 479-483.

Kovesi, P. (1999). Image Features from Phase Congruency. *Journal of Computer Vision Research, 1*(3), 26.

Kruse II, W. G., & Heiser, J. G. (2001). *Computer forensics: incident response essentials*: Pearson Education.

Kuhn, M. G. (2002). *Compromising emanations: eavesdropping risks of computer displays.* University of Cambridge.

Lades, M., Vorbrüggen, J. C., Buhmann, J., Lange, J., d Malsburg, C. V., Wurtz, R., & Konen, W. (1993). Distortion invariant object recognition in the dynamic link architecture. *IEEE Transactions on Computers, 42*(3), 300-311.

Lai, J. H., Yuen, P. C., & Feng, G. C. (2001). Face recognition using holistic Fourier invariant features. *Pattern Recognition, 34*(1), 95-109.

Laine, A., & Fan, J. (1993). Texture classification by wavelet packet signatures. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 15*(11), 1186-1191.

Leach, S. (1995). Singular value decomposition-a primer. *Unpublished Manuscript, Department of Computer Science, Brown University, Providence, RI, USA*.

Li, J., Wang, Y., Tan, T., & Jain, A. K. (2004). *Live face detection based on the analysis of fourier spectra.* Paper presented at the Defense and Security.

Li, X., Jing, T., & Li, X. (2010). *Image splicing detection based on moment features and Hilbert-Huang Transform.* Paper presented at the IEEE International Conference on Information Theory and Information Security (ICITIS).

Lin, C.-Y., & Chang, S.-F. (1998). *Generating robust digital signature for image/video authentication.* Paper presented at the Multimedia and Security Workshop at ACM Multimedia.

Lu, C.-S., & Liao, H.-Y. M. (2003). Structural digital signature for image authentication: an incidental distortion resistant scheme. *Multimedia, IEEE Transactions on, 5*(2), 161-173.

Lu, J., & Wang, H. (2005). Comparison Between SVD and DCT Feature Extraction Methods in Face Recognition. *Editorial Board of Geomatics and Information Science of Wuhan University, 30*(02), 18-19.

Luo, W., Qu, Z., Pan, F., & Huang, J. (2007). A survey of passive technology for digital image forensics. *Frontiers of Computer Science in China, 1*(2), 166-179.

Machado, J. T. (2014). Fractional order generalized information. *Entropy, 16*(4), 2350-2361.

Mashao, D. J. (2003). *Comparing SVM and GMM on parametric feature-sets.* Paper presented at the Proceedings of the 14th Annual Symposium of the Pattern Recognition Association of South Africa.

Mather, P. M., & Koch, M. (2011). *Computer processing of remotely-sensed images: an introduction*: John Wiley & Sons.

Mitra, S., & Acharya, T. (2005). *Data mining: multimedia, soft computing, and bioinformatics*. New York: John Wiley & Sons.

Moghaddasi, Z., Jalab, H. A., Md Noor, R., & Aghabozorgi, S. (2014). Improving RLRN image splicing detection with the use of PCA and Kernel PCA. *The Scientific World Journal, 2014*.

Moghaddasi, Z., Jalab, H. A., & Noor, R. M. (2014). *SVD-based image splicing detection.* Paper presented at the International Conference on Information Technology and Multimedia (ICIMU), Malaysia.

Moghaddasi, Z., Jalab, H. A., & Noor, R. M. (2015). *A comparison study on SVD-based features in different transforms for image splicing detection.* Paper presented at the IEEE International Conference on Consumer Electronics-Taiwan (ICCE-TW), Taiwan.

Moonen, M., Van Dooren, P., & Vandewalle, J. (1992). A singular value decomposition updating algorithm for subspace tracking. *SIAM Journal on Matrix Analysis and Applications, 13*(4), 1015-1038.

Morrone, M. C., & Burr, D. (1988). Feature detection in human vision: A phase-dependent energy model. *Proceedings of the Royal Society of London. Series B, biological sciences*, 221-245.

Morrone, M. C., Ross, J., Burr, D. C., & Owens, R. (1986). Mach bands are phase dependent. *Nature, 324*(6094), 250-253.

Muhammad, G., Al-Hammadi, M. H., Hussain, M., & Bebis, G. (2014). Image forgery detection using steerable pyramid transform and local binary pattern. *Machine Vision and Applications, 25*(4), 985-995.

Mushtaq, S., & Mir, A. H. (2014). Digital Image Forgeries and Passive Image Authentication Techniques: A Survey. *International Journal of Advanced Science and Technology, 73*, 15-32.

Ng, T.-T., & Chang, S.-F. (2004). *A model for image splicing.* Paper presented at the International Conference on Image Processing, 2004. ICIP'04. .

Ng, T.-T., Chang, S.-F., Lin, C.-Y., & Sun, Q. (2006). Passive-blind image forensics. *Multimedia Security Technologies for Digital Rights, 15*, 383-412.

Ng, T.-T., Chang, S.-F., & Sun, Q. (2004a). *Blind detection of photomontage using higher order statistics.* Paper presented at the International Symposium on Circuits and Systems. ISCAS'04. .

Ng, T.-T., Chang, S.-F., & Sun, Q. (2004b). *Blind detection of photomontage using higher order statistics.* Paper presented at the International Symposium on Circuits and Systems. ISCAS'04.

Ng, T.-T., Chang, S.-F., & Sun, Q. (2004c). A data set of authentic and spliced image blocks. *Columbia University, ADVENT Technical Report.* from http://www.ee.columbia.edu/ln/dvmm/

Ng, T.-T., Hsu, J., & Chang, S.-F. (2009). *Columbia image splicing detection evaluation dataset* [Image]. Retrieved from: http://www.ee.columbia.edu/ln/dvmm/

Nixon, M. (2008). *Feature extraction & image processing*: Academic Press.

Parekh, N. (2016, January 19). The Waterfall Model Explained. Retrieved August, 2016, from http://www.buzzle.com/editorials/1-5-2005-63768.asp

Pearson, H. (2005). Image manipulation: CSI: cell biology. *Nature, 434*(7036), 952-953.

Popejoy, A. L. (2015). *Digital and multimedia forensics justified: An appraisal on professional policy and legislation.* University of Colorado Denver.

Pratt, W. K. (1991). Image segmentation. *Digital Image Processing: PIKS Inside, Third Edition*, 551-587.

Redi, J. A., Taktak, W., & Dugelay, J.-L. (2011). Digital image forensics: a booklet for beginners. *Multimedia Tools and Applications, 51*(1), 133-162.

Renkjumnong, W. (2007). *SVD and PCA in Image Processing.* (Master of Science (MS) Thesis), Georgia State University. Retrieved from http://scholarworks.gsu.edu/math_theses/31

Rocha, A., Scheirer, W., Boult, T., & Goldenstein, S. (2011). Vision of the unseen: Current trends and challenges in digital image and video forensics. *ACM Computing Surveys (CSUR), 43*(4), 26.

Rosales-Roldan, L., Cedillo-Hernandez, M., Nakano-Miyatake, M., Perez-Meana, H., & Kurkoski, B. (2013). Watermarking-based image authentication with recovery capability using halftoning technique. *Signal Processing: Image Communication, 28*(1), 69-83.

Rosipal, R., Girolami, M., Trejo, L. J., & Cichocki, A. (2001). Kernel PCA for feature extraction and de-noising in nonlinear regression. *Neural Computing & Applications, 10*(3), 231-243.

Sadek, R. A. (2008). *Blind synthesis attack on SVD based watermarking techniques.* Paper presented at the International Conference on Computational Intelligence for Modelling Control & Automation.

Sadek, R. A. (2012). SVD based image processing applications: state of the art, contributions and research challenges. *International Journal of Advanced Computer Science and Applications (IJACSA), 3*(7), 9.

Saferstein, R. (2001). An Introduction to Forensic Science. *Criminalistics. Prentice Hall, Upper Saddle River, New Jersy*, 108-112.

Schneier, B., & Kelsey, J. (1998). *Cryptographic Support for Secure Logs on Untrusted Machines.* Paper presented at the USENIX Security.

Schölkopf, B., Smola, A., & Müller, K.-R. (1997). Kernel principal component analysis *Artificial Neural Networks—ICANN'97* (pp. 583-588): Springer.

Sengupta, M., & Mandal, J. (2013). Authentication Through Hough Transformation Generated Signature on G-Let D3 Domain (AHSG). *Procedia Technology, 10*, 121-130.

Shi, Y. Q., Chen, C., & Chen, W. (2007a). *A Markov process based approach to effective attacking JPEG steganography.* Paper presented at the Information Hiding.

Shi, Y. Q., Chen, C., & Chen, W. (2007b). *A natural image model approach to splicing detection.* Paper presented at the Proceedings of the 9th workshop on Multimedia & security.

Shieh, J.-M., Lou, D.-C., & Chang, M.-C. (2006). A semi-blind digital watermarking scheme based on singular value decomposition. *Computer Standards & Interfaces, 28*(4), 428-440.

Silva, E., & Ghanbari, M. (1996). On the performance of linear phase wavelet transforms in low bit-rate image coding. *IEEE Transactions on Image Processing, 5*(5), 689-704.

Sokolova, M., Japkowicz, N., & Szpakowicz, S. (2006). *Beyond accuracy, F-score and ROC: a family of discriminant measures for performance evaluation.* Paper presented at the Australasian Joint Conference on Artificial Intelligence.

Spagnolo, G. S., & De Santis, M. (2011). Holographic watermarking for authentication of cut images. *Optics and Lasers in Engineering, 49*(12), 1447-1455.

Srivastava, A., Lee, A. B., Simoncelli, E. P., & Zhu, S.-C. (2003). On advances in statistical modeling of natural images. *Journal of mathematical imaging and vision, 18*(1), 17-33.

Stokman, H., & Gevers, T. (2007). Selection and fusion of color models for image feature detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 29*(3), 371-381.

Sutthiwan, P., Cai, X., Shi, Y. Q., & Zhang, H. (2009). *Computer graphics classification based on Markov process model and boosting feature selection technique.* Paper presented at the 16th IEEE International Conference on Image Processing (ICIP).

Sutthiwan, P., Shi, Y.-Q., Dong, J., Tan, T., & Ng, T.-T. (2010). *New developments in color image tampering detection.* Paper presented at the Proceedings of 2010 IEEE International Symposium on Circuits and Systems (ISCAS).

Sverdlov, A., Dexter, S., & Eskicioglu, A. M. (2006). Secure DCT-SVD domain image watermarking: embedding data in all frequencies. *Dept. Computer and Information Science, Brooklyn College, City Univ. of New York*.

Tang, X. (1998). Texture information in run-length matrices. *IEEE Transactions on Image Processing, 7*(11), 1602-1609.

The MathWorks, I. (2012). *MATLAB, The Language of Technical Computing* (pp. 25). Retrieved from http://www.mathworks.com/

Thompson, C. M., & Shure, L. (1995). Image processing toolbox [for use with Matlab].

van der Maaten, L. J., Postma, E. O., & van den Herik, H. J. (2009). Dimensionality reduction: A comparative review. *Journal of Machine Learning Research, 10*(1-41), 66-71.

Viniotis, Y. (1998). *Probability and random processes for electrical engineers*: McGraw-Hill Companies.

Vladimir, V. N., & Vapnik, V. (1995). The nature of statistical learning theory: Springer Heidelberg.

Wang, W., Dong, J., & Tan, T. (2009). *Effective image splicing detection based on image chroma.* Paper presented at the 16th IEEE International Conference on Image Processing (ICIP).

WANG, W., YANG, J., & CHEN, F. (2006). Method of Image Feature Extraction Based on SVD. *Computer Engineering, 8*, 011.

Wang, X., Xue, J., Zheng, Z., Liu, Z., & Li, N. (2012). Image forensic signature for content authenticity analysis. *Journal of Visual Communication and Image Representation, 23*(5), 782-797.

Wright, C., & Kleiman, D. (2008). Overwriting hard drive data: The great wiping controversy *Information systems security* (pp. 243-257): Springer.

Xu, X., Dexter, S. D., & Eskicioglu, A. M. (2004). *A hybrid scheme for encryption and watermarking.* Paper presented at the Electronic Imaging 2004.

Yang, J.-F., & Lu, C.-L. (1995). Combined techniques of singular value decomposition and vector quantization for image coding. *IEEE Transactions on Image Processing, 4*(8), 1141-1146.

Zander, S., & Murdoch, S. J. (2008). *An Improved Clock-skew Measurement Technique for Revealing Hidden Services.* Paper presented at the USENIX Security Symposium.

Zhang, G., Huang, X., Li, S. Z., Wang, Y., & Wu, X. (2004). Boosting local binary pattern (LBP)-based face recognition *Advances in biometric person authentication* (pp. 179-186): Springer.

Zhang, H.-B., Yang, C., & Quan, X.-M. (2004). Image authentication based on digital signature and semi-fragile watermarking. *Journal of Computer Science and Technology, 19*(6), 752-759.

Zhang, J., Zhao, Y., & Su, Y. (2009). *A new approach merging Markov and DCT features for image splicing detection.* Paper presented at the IEEE International Conference on Intelligent Computing and Intelligent Systems. ICIS 2009. .

Zhang, Y., Zhao, C., Pi, Y., & Li, S. (2012). Revealing image splicing forgery using local binary patterns of DCT coefficients *Communications, Signal Processing, and Systems* (pp. 181-189): Springer.

Zhang, Z., Ren, Y., Ping, X.-J., He, Z.-Y., & Zhang, S.-Z. (2008). *A survey on passive-blind image forgery by doctor method detection.* Paper presented at the International Conference on Machine Learning and Cybernetics. .

Zhang, Z., Zhou, Y., Kang, J., & Ren, Y. (2008). Study of image splicing detection *Advanced Intelligent Computing Theories and Applications. With Aspects of Theoretical and Methodological Issues* (pp. 1103-1110): Springer.

Zhao, X., Li, J., Li, S., & Wang, S. (2011). Detecting digital image splicing in chroma spaces *Digital Watermarking* (pp. 12-22): Springer.

Zhao, X., Wang, S., Li, S., & Li, J. (2012). A comprehensive study on third order statistical features for image splicing detection *Digital Forensics and Watermarking* (pp. 243-256): Springer.

Zhou, Z.-p., & Zhang, X.-x. (2010). *Image splicing detection based on image quality and analysis of variance.* Paper presented at the 2nd International Conference on Education Technology and Computer (ICETC).

LIST OF PUBLICATIONS AND PAPERS PRESENTED

Journal Papers:

- Zahra Moghaddasi, Hamid A. Jalab, Rafidah Md Noor, and Saeed Aghabozorgi. 2014. "Improving RLRN Image Splicing Detection with the Use of PCA and Kernel PCA". The Scientific World Journal (ISI- Q1). Special Issue: Security of Information and Networks. (*ISI/SCOPUS Cited Publication*).

- Rabha Ibrahim, Zahra Moghaddasi, Hamid Jalab, Rafidah Md Noor. 2015. "Fractional differential texture descriptors based on the Machado entropy for image splicing detection". Entropy, 17, 4775-4785. special issue "Complex and Fractional Dynamics". doi: 10.3390/e17074775 (ISI-Q2). For 2014, the journal Entropy has an Impact Factor of 1.502. (*ISI/SCOPUS Cited Publication*).

Conference Papers:

- Zahra Moghaddasi, Hamid A. Jalab, Rafidah Md Noor. "SVD-Based Image Splicing Detection". International conference on information technology and multimedia 2014(ICIMu2014). (*SCOPUS-Indexed*). *(Best Paper Award)*

- Zahra Moghaddasi, Hamid A. Jalab, Rafidah Md Noor ,2015. "A Comparison Study on SVD-based Features in Different Transforms for Image Splicing Detection". Special session in "Intelligent service technology for wearable consumer electronics"- "Multimedia Security and its Applications for Consumer Electronics". IEEE International Conference on Consumer Electronics- Taiwan 2015(IEEE 2015 ICCE TW). (*SCOPUS-Indexed*).

- Hamid A. Jalab, Zahra Moghaddasi, Ali M. Hasan, and Zouhir Wakaf. “Image Splicing Detection using Electromagnetism-like Based Descriptor”. SAI Intelligent Systems Conference 2016, September 21-22, 2016, London, UK (*SCOPUS-Indexed*).