HAL
open science

Towards digital image anti-forensics via image restoration

Wei Fan

To cite this version:

Wei Fan. Towards digital image anti-forensics via image restoration. Signal and Image processing. Université Grenoble Alpes; Beihang university (Pékin), 2015. English. NNT : 2015GREAT035 . tel-01216243

HAL Id: tel-01216243

https://theses.hal.science/tel-01216243v1

Submitted on 15 Oct 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

THÈSE

pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ DE GRENOBLE ALPES

préparée dans le cadre d'une cotutelle entre l'*Université Grenoble Alpes* et *Beihang University*

Spécialité : **signal, image, parole, télécommunications (SIPT)**

Arrêté ministériel : le 6 janvier 2005 - 7 août 2006

Présentée par

Wei FAN

Thèse dirigée par **Jean-Marc BROSSIER** et **Zhang XIONG**
et encadrée par **François CAYRE** et **Kai WANG**

préparée au sein du **laboratoire Grenoble, images, parole, signal, automatique (GIPSA-lab)** et **School of Computer Science and Engineering**

dans **l'école doctorale d'électronique, électrotechnique, automatique et traitement du signal (EEATS)** et **Beihang University**

Vers l'anti-criminalistique en images numériques via la restauration d'images

Thèse soutenue publiquement le **30/04/2015**,
devant le jury composé de:

Jean-Marc CHASSERY
Directeur de Recherche CNRS, GIPSA-lab, Examinateur, Président
Jean-Luc DUGELAY
Professeur, EURECOM, Sophia Antipolis, Rapporteur
Stefano TUBARO
Professeur, Politecnico di Milano, Rapporteur
Teddy FURON
Chargé de Recherche INRIA, INRIA Rennes, Examinateur
Jiwu HUANG
Professeur, Shenzhen University, Examinateur
Zhang XIONG
Professeur, Beihang University, Co-directeur
François CAYRE
Maître de Conférences, Grenoble INP, GIPSA-lab, Co-encadrant
Kai WANG
Chargé de Recherche CNRS, GIPSA-lab, Co-encadrant

UNIVERSITY OF GRENOBLE ALPES

Doctoral school EEATS

(Électronique, Électrotechnique, Automatique et Traitement du Signal)

T H E S I S

for obtaining the title of

Doctor of Science

of the University of Grenoble Alpes

Speciality: SIPT
(Signal, Image, Parole, Télécommunications)

Presented by

Wei FAN

Towards Digital Image Anti-Forensics via Image Restoration

Thesis supervised by Jean-Marc BROSSIER and Zhang XIONG
and co-supervised by François CAYRE and Kai WANG

prepared at
Grenoble - Images, Parole, Signal, Automatique Laboratory (GIPSA-lab)
and School of Computer Science and Engineering, Beihang University

presented on 30/04/2015

Jury:

President:	Jean-Marc CHASSERY	-	CNRS, GIPSA-lab
Reviewers:	Jean-Luc DUGELAY	-	EURECOM, Sophia Antipolis
	Stefano TUBARO	-	Politecnico di Milano
Examiners:	Teddy FURON	-	INRIA Rennes
	Jiwu HUANG	-	Shenzhen University
Supervisor:	Zhang XIONG	-	Beihang University
Co-Supervisors:	François CAYRE	-	Grenoble INP, GIPSA-lab
	Kai WANG	-	CNRS, GIPSA-lab

Acknowledgements

First and foremost, I would like to express my deepest gratitude to my four thesis supervisors: Prof. Jean-Marc Brossier, Dr. François Cayre, and Dr. Kai Wang in GIPSA-lab, and Prof. Zhang Xiong in Beihang University. I am grateful to Prof. Brossier and Prof. Xiong, for offering me the wonderful opportunity to conduct a co-supervised Ph.D. thesis between GIPSA-lab/Grenoble INP and Beihang University. Not only did Dr. Cayre give me countless insightful suggestions on the research work, but he also consistently encouraged me when I faced difficulties. He created a motivating, pleasant and relaxing working environment, where I could work effectively and at my pace. I owe my eternal thanks to Dr. Wang for his close guidance as well as constant support, which pushed me to achieve what I never thought possible. He is always patient in answering my questions, always efficient in correcting my mistakes, and always inspiring in our discussions, which are reflected throughout this thesis. Besides, I also would like to sincerely thank Prof. Gang Feng in GIPSA-lab and Prof. Shixiang Qian in Beihang University, who made this thesis opportunity accessible to me.

I would like to express my sincere gratitude to Prof. Jean-Luc Dugelay, and Prof. Stefano Tubaro, for taking their precious time to review my thesis manuscript. I also would like to gratefully thank Prof. Jean-Marc Chassery, Dr. Teddy Furon, and Prof. Jiwu Huang, for being examiners of my Ph.D. defense committee.

I am grateful to Dr. Xiyan He, who not only is a very good friend in the personal life but also provided me with insightful discussions on the total variation and the support vector machine. I would like to thank Dr. Rodrigo Cabral Farias for answering my many mathematical questions and for referring me to many interesting materials. I am grateful to Dr. Zhenyong Chen and Ming Chen for their valuable suggestions and help when I started to work on multimedia security. I also want to thank Dr. Chuantao Yin and Hui Chen for their kind help and support for my coming to GIPSA-lab for study.

I want to thank my colleagues in the same office, Fatima, Jérémie, Fakhri, and Quentin in GIPSA-lab, and Shuo, Xiaolan, Jiahui, and Jianyuan in Beihang University, for creating such a crazy, funny, and friendly office atmosphere. It was the best I could expect, especially when there was a lot of stress from work. My thanks also go to Aude, Cyrille, Diyin, Gailene, Guanghan, Jonathan, Junshi, Longyu, Matthieu, Robin, Sheng, Vincent, Yang, Ying, and Zhongyang for making my stay in France much easier. They have helped me during my daily life in various forms, supporting me through hard times, integrating me into many interesting outings and activities, introducing me to French culture, kindly helping me on my French, *etc.*

I am indebted to my dearest family, for their endless love and unconditional support. No matter how far away I am from home, it is the source of my strength knowing that they are always there for me.

The work presented in this thesis is supported in part by the China Scholarship Council under Grant 2011602067, in part by the French ANR Estampille under Grant ANR-10-CORD-

019, in part by the French Eiffel Scholarship under Grant 812587B, and in part by the French Rhône-Alpes region through the CMIRA Scholarship program. I would like to express my gratitude to them for the financial support, without which this thesis would not exist.

Abstract

Image forensics enjoys its increasing popularity as a powerful image authentication tool, working in a blind passive way without the aid of any *a priori* embedded information compared to fragile image watermarking. On its opponent side, image anti-forensics attacks forensic algorithms for the future development of more trustworthy forensics. When image coding or processing is involved, we notice that image anti-forensics to some extent shares a similar goal with image restoration. Both of them aim to recover the information lost during the image degradation, yet image anti-forensics has one additional indispensable forensic undetectability requirement. In this thesis, we form a new research line for image anti-forensics, by leveraging on advanced concepts/methods from image restoration meanwhile with integrations of anti-forensic strategies/terms. Under this context, this thesis contributes on the following four aspects for JPEG compression and median filtering anti-forensics: (i) JPEG anti-forensics using Total Variation based deblocking, (ii) improved Total Variation based JPEG anti-forensics with assignment problem based perceptual DCT histogram smoothing, (iii) JPEG anti-forensics using JPEG image quality enhancement based on a sophisticated image prior model and non-parametric DCT histogram smoothing based on calibration, and (iv) median filtered image quality enhancement and anti-forensics via variational deconvolution. Experimental results demonstrate the effectiveness of the proposed anti-forensic methods with a better forensic undetectability against existing forensic detectors as well as a higher visual quality of the processed image, by comparisons with the state-of-the-art methods.

Keywords: Image anti-forensics, image restoration, JPEG compression, median filtering

Contents

Acknowledgements **v**

Abstract **vii**

Contents **ix**

List of Figures **xv**

List of Tables **xix**

Notations **xxi**

Acronyms **xxv**

1 Introduction **1**

1.1 Can You Believe Your Eyes? 1

1.2 Image Anti-Forensics 3

1.3 Objectives and Contributions 4

1.3.1 JPEG Compression and Median Filtering 4

1.3.2 Image Anti-Forensics and Image Restoration 6

1.3.3 Methodology 6

1.4 Outline 7

2 Preliminaries **9**

2.1 Classification of Image (Anti-)Forensics 10

2.1.1 Farid's Classification of Image Forensics 10

2.1.2 Redi *et al.*'s Classification of Image Forensics 10

2.1.3 Piva's Classification of Image Forensics 11

2.1.4 Stamm *et al.*'s Classification of Image Forensics 12

2.1.5 Böhme and Kirchner's Classification of Image Anti-Forensics 13

2.1.6 Classification of Proposed Anti-Forensic Methods 14

2.2 Evaluation Metrics 14

2.2.1 Forensic (Un)detectability 15

2.2.2 Image Quality 21

2.2.3 Histogram Recovery . . . 22
2.3 Natural Image Datasets . . . 22
2.3.1 JPEG Forensic Testing . . . 22
2.3.2 Median Filtering Forensic Testing . . . 24
2.4 Relevant Optimization Algorithms . . . 25
2.4.1 Subgradient Method . . . 25
2.4.2 Hungarian Algorithm . . . 26
2.4.3 Half Quadratic Splitting . . . 27
2.4.4 Split Bregman Method . . . 28

3 Prior Art of JPEG and Median Filtering (Anti-)Forensics 31
3.1 JPEG Forensics and Anti-Forensics . . . 32
3.1.1 Basics of JPEG Compression . . . 32
3.1.2 JPEG Artifacts . . . 33
3.1.3 JPEG Image Quality Enhancement . . . 35
3.1.4 Detecting JPEG Compression . . . 36
3.1.5 Disguising JPEG Artifacts . . . 36
3.1.6 Attacking JPEG Anti-Forensics . . . 37
3.1.7 Other Relevant Methods . . . 38
3.1.8 Summary . . . 38
3.2 Median Filtering Forensics and Anti-Forensics . . . 40
3.2.1 Median Filtering Basics and Artifacts . . . 40
3.2.2 Detecting Median Filtering . . . 41
3.2.3 Disguising Median Filtering Artifacts . . . 44
3.2.4 Summary . . . 45

4 Total Variation Based JPEG Anti-Forensics 47
4.1 Introduction and Motivation . . . 48
4.2 Performance Analysis of Scalar-Based JPEG Detectors . . . 49
4.2.1 Quantization Table Estimation Based Detector . . . 49
4.2.2 Other Scalar-Based JPEG Forensic Detectors . . . 51
4.3 JPEG Anti-Forensics via TV-Based Deblocking . . . 53
4.3.1 JPEG Deblocking Using Constrained TV-Based Minimization . . . 53
4.3.2 De-Calibration . . . 56
4.4 Experimental Results . . . 56
4.4.1 Parameter Settings . . . 56

4.4.2 Comparison and Analysis . . . 58
4.5 Summary . . . 61

5 JPEG Anti-Forensics with Perceptual DCT Histogram Smoothing 65
5.1 Introduction and Motivation . . . 67
5.2 Proposed JPEG Anti-Forensics . . . 69
5.2.1 First-Round TV-Based Deblocking . . . 69
5.2.2 Perceptual DCT Histogram Smoothing . . . 71
5.2.3 Second-Round TV-Based Deblocking . . . 79
5.2.4 De-Calibration . . . 80
5.3 Experimental Results of JPEG Anti-Forensics . . . 81
5.3.1 Comparing Anti-Forensic Dithering Methods . . . 81
5.3.2 Against JPEG Forensic Detectors . . . 85
5.3.3 Computation Cost . . . 87
5.4 Hiding Traces of Double JPEG Compression Artifacts . . . 88
5.4.1 Hiding Traces of Aligned Double JPEG Compression . . . 90
5.4.2 Hiding Traces of Non-Aligned Double JPEG Compression . . . 92
5.4.3 Fooling JPEG Artifacts Based Image Forgery Localization . . . 93
5.5 Summary . . . 94
5.A Appendix: The p.m.f. of the Dithering Signal Using the Laplacian Model . . . 98
5.B Appendix: The Constraints Used for Modeling the DCT Coefficients . . . 99

6 JPEG Image Quality Enhancement and Anti-Forensics Using a Sophisticated Image Prior Model 105
6.1 Introduction and Motivation . . . 106
6.2 JPEG Image Quality Enhancement . . . 107
6.2.1 Prior Art . . . 107
6.2.2 Proposed Method . . . 108
6.3 Non-Parametric DCT Histogram Smoothing . . . 110
6.4 Proposed JPEG Anti-Forensics . . . 116
6.5 Summary . . . 121

7 Median Filtered Image Quality Enhancement and Anti-Forensics via Variational Deconvolution 123
7.1 Introduction and Motivation . . . 125
7.2 Analysis of Median Filtering and Its Impact on Image Statistics . . . 126

7.2.1 Median Filtering Process . . . 126
7.2.2 Observations of Pixel Value Difference Distribution . . . 127
7.3 Proposed Image Variational Deconvolution Framework . . . 129
7.3.1 Problem Formulation . . . 129
7.3.2 Kernel Selection and Parameter Settings . . . 132
7.3.3 Median Filtered Image Quality Enhancement . . . 133
7.3.4 Anti-Forensics against Median Filtering Detection . . . 135
7.4 Applications: Disguising Footprints of Both Median Filtering and Targeted Image Operation of Median Filtering Processing . . . 146
7.4.1 Hiding Traces of Image Resampling . . . 146
7.4.2 Removing JPEG Blocking Artifacts . . . 149
7.5 Summary . . . 150

8 Conclusions 153
8.1 Summary of Contributions . . . 153
8.2 Perspectives . . . 156

A Résumé en Français 159
A.1 Introduction . . . 160
A.1.1 Pouvez-vous croire vos yeux ? . . . 160
A.1.2 Anti-criminalistique en images numériques . . . 163
A.1.3 Objectifs et contributions . . . 163
A.1.4 Organisation du résume . . . 166
A.2 Préliminaires . . . 167
A.2.1 Classification de la criminalistique et l'anti-criminalistique d'image . . . 167
A.2.2 Métriques d'évaluation . . . 169
A.2.3 Ensembles d'images naturelles . . . 172
A.2.4 Algorithmes pertinents d'optimisation . . . 173
A.3 État de l'art en (anti-)criminalistique de compression JPEG et de filtrage médian 174
A.3.1 (Anti-)Criminalistique de compression JPEG . . . 174
A.3.2 (Anti-)Criminalistique du filtrage médian . . . 176
A.4 Anti-criminalistique de compression JPEG basée sur la TV . . . 177
A.4.1 Introduction et motivation . . . 177
A.4.2 Déblocage JPEG en minimisant un problème contraint basé sur la TV . 178
A.4.3 Décalibrage . . . 179
A.4.4 Quelques résultats expérimentaux . . . 179

A.5 Anti-criminalistique de compression JPEG avec un lissage perceptuel de l'histogramme DCT . . . 181
A.5.1 Introduction et motivation . . . 181
A.5.2 Lissage perceptuel de l'histogramme DCT . . . 183
A.5.3 Quelques résultats expérimentaux . . . 185
A.6 Amélioration de qualité et anti-criminalistique de l'image JPEG basée sur un modèle d'image avancé . . . 187
A.6.1 Introduction et motivation . . . 187
A.6.2 Amélioration de qualité de l'image JPEG . . . 187
A.6.3 Anti-criminalistique de compression JPEG . . . 188
A.7 Amélioration de la qualité et anti-criminalistique de l'image filtrée par le filtre médian à l'aide d'une déconvolution variationnelle d'image . . . 190
A.7.1 Introduction et motivation . . . 190
A.7.2 Déconvolution variationnelle d'image . . . 191
A.7.3 Amélioration de qualité de l'image MF . . . 193
A.7.4 Anti-criminalistique de filtrage médian . . . 195
A.8 Conclusions et perspectives . . . 197
A.8.1 Résumé des contributions . . . 197
A.8.2 Perspectives . . . 201

Bibliography **205**

Author's Publications **215**

List of Figures

1.1 An example of image forgery. 1
1.2 Annual number of IEEE publications involving forensics. 3
1.3 Illustration of creating a composite JPEG image. 5

2.1 The ROC space and an example ROC curve. 16
2.2 Illustration of creating a composite image forgery for training/testing the SVM-based detector. 20

3.1 Examples of JPEG artifacts. 34
3.2 Illustration of DCT basis functions. 35
3.3 Examples of first-order horizontal pixel value difference histograms. 41

4.1 Example DCT histograms from which detector K_F^Q [FD03] fails to detect the correct quantization step. 51
4.2 ROC curves achieved by different (state-of-the-art anti-forensic) JPEG images. 52
4.3 Pixel classification according to its position in the 8×8 pixel value block. . . . 54
4.4 TV-based blocking measurement test. 55
4.5 Performance variation trend of $\hat{\mathcal{F}}_0^J$ under different settings of α. 58
4.6 ROC curves achieved by different (anti-forensic/post-processed) JPEG images. 59
4.7 Example results of $\mathcal{I}$ and $\mathcal{J}$. 61
4.8 Example results of $\mathcal{J}_A$ [ADF05], $\mathcal{F}_V^J$ [VTT11] and $\mathcal{F}_{Su}^J$ [SS11]. 62
4.9 Example results of $\mathcal{F}_{S_q}^J$ [Sta+10a, SL11], $\mathcal{F}_{S_qS_b}^J$ [Sta+10b, SL11] and $\mathcal{F}_0^J$. . . . 63
4.10 Example DCT histograms of $\mathcal{F}_{S_qS_b}^J$ [Sta+10a, Sta+10b, SL11] and $\mathcal{F}_0^J$ 64

5.1 The proposed anti-forensic JPEG image creation process for $\mathcal{F}^J$. 68
5.2 Performance variation trend of $\hat{\mathcal{F}}_b^J$ under different settings of α. 70
5.3 Example DCT histograms of $\mathcal{I}$, $\mathcal{J}$, $\hat{\mathcal{F}}_b^J$, and $\hat{\mathcal{F}}_b^J$ after the adaptive local dithering signal injection. 72
5.4 Illustration for the constraint used for the searching of λ_b^+ when $\mathbf{Q}_{r,c}$ is an odd number, in the quantization bin $b = 0$. 75
5.5 Comparison of SSIM values achieved by $\mathcal{J}$, $\mathcal{F}_{S_q}^J$ [Sta+10a, SL11], $\mathcal{F}_V^J$ [VTT11], $\hat{\mathcal{F}}_{bq}^J$, and $\hat{\mathcal{F}}_{bq}'$, with $\mathcal{I}$ as the reference. 82
5.6 Time taken to create $\hat{\mathcal{F}}_{bq}^J$, and $\hat{\mathcal{F}}_{bq}'$ from the JPEG compressed "Lena" image with different quality factors. 82
5.7 Example results of $\hat{\mathcal{F}}_{bq}^J$ compared with $\mathcal{J}$, $\mathcal{F}_{S_q}^J$ [Sta+10a, SL11], and $\mathcal{F}_V^J$ [VTT11]. 83

5.8 ROC curves achieved by $\mathcal{F}^J$ and $\mathcal{F}^J_{S_qS_b}$ [Sta+10a, Sta+10b, SL11]. 85
5.9 AUC values achieved at different image replacement rates for different (anti-forensic) JPEG images against SVM-based detectors. 87
5.10 Example results of $\mathcal{F}^J$ compared with $\mathcal{I}$, $\mathcal{J}$, and $\mathcal{F}^J_{S_qS_b}$ [Sta+10a, Sta+10b, SL11]. 88
5.11 Example DCT histograms of $\mathcal{F}^J$. 89
5.12 ROC curves achieved on **A-DJPG-*R*** against the SVM-based A-DJPG compression detector [PF08]. 91
5.13 Average AUC value over QF_1 as a function of QF_2 achieved on **NA-DJPG-*R*** against the NA-DJPG compression detector [BP12a]. 92
5.14 Average AUC value over QF_1 as a function of QF_2 achieved on **LOC-*E*-DJPG-*K*/*L*-*R*** against the forgery localization detector [BP12b]. 95
5.15 Illustration for the constraint used for the searching of λ_b^+ when $\mathbf{Q}_{r,c}$ is an even number, in the quantization bin $b = 0$. 99
5.16 Illustration for the constraints used for the searching of λ_b^- and λ_b^+, in the quantization bin $b > 0$. 101

6.1 Comparison of DCT-domain quantization noise stacked by the spatial-domain location and by the 64 DCT frequencies. 111
6.2 Example DCT histograms of $\hat{\mathcal{F}}^J_b$, $\hat{\mathcal{I}}^J$, $\hat{\mathcal{F}}^J_{bq}$ and $\hat{\mathcal{F}}^J_c$. 114
6.3 ROC curves achieved by $\mathcal{F}^J_1$. 117
6.4 Example results of $\mathcal{F}^J_1$ compared with $\mathcal{I}$, $\mathcal{J}$, and $\mathcal{F}_{S_qS_b}$. 119
6.5 Example DCT histograms of $\mathcal{F}^J_1$. 120

7.1 Examples of first-order horizontal pixel value difference histograms. 128
7.2 Image quality variation trend for different convolution kernels and ω values. . 134
7.3 Image quality variation trend of the quality enhanced MF image using the AVE kernel with $\omega = 0.4$, for different parameter combinations (λ, γ). 134
7.4 Example results of the proposed MF image quality enhancement method. . . . 136
7.5 Image quality variation trend of $\mathcal{F}^M$ generated using the AVEE kernel with $\omega = 0.1$, for different parameter combinations (λ, γ). 139
7.6 Anti-forensic performance variation trend of $\mathcal{F}^M$ generated using the AVEE kernel with $\omega = 0.1$, for different parameter combinations (λ, γ). 140
7.7 Example results of $\mathcal{F}^M$, compared with $\mathcal{M}$. 141
7.8 Anti-forensic performance variation trend of $\mathcal{F}^M_D$ [DN+13] with different parameter settings. 142
7.9 ROC curves achieved by $\mathcal{M}$, $\mathcal{F}^M_W$ [WSL13], $\mathcal{F}^M_W$ [WSL13], and $\mathcal{F}^M$ against scalar-based median filtering forensic detectors. 143
7.10 AUC values achieved at different image replacement rates for different (anti-forensic) MF images against SVM-based detectors. 145

7.11 AUC values achieved at different image replacement rates for different (anti-forensic/median filtered) resampled images against SVM-based detectors. . . . 148
7.12 AUC values achieved at different image replacement rates for different (anti-forensic/median filtered) JPEG images against SVM-based detectors. 151

A.1 Un exemple de l'image fausse. 160
A.2 Nombre annuel de publications de l'IEEE sur la criminalistique. 162
A.3 Illustration de créer une image composite en JPEG. 164
A.4 Un exemple de la courbe ROC. 170
A.5 Illustration de créer une image composite fausse afin de former/tester le détecteur à base du SVM. 171
A.6 Exemples de les artefacts de la compression JPEG. 174
A.7 Exemples de l'histogramme de différence de valeurs de pixels au premier ordre. 176
A.8 Classification des pixels en fonction de leur position dans un bloc de taille 8×8. 179
A.9 Les courbes ROC obtenues pour différentes images contre différents détecteurs criminalistiques. 180
A.10 Exemples d'histogrammes DCT de $\mathcal{F}^J_{S_q S_b}$ [Sta+10a, Sta+10b, SL11] et de $\mathcal{F}^J_0$ 181
A.11 Le processus proposé pour créer l'image anti-criminalistique $\mathcal{F}^J$. 182
A.12 Exemples d'histogrammes DCT des images $\mathcal{I}$, $\mathcal{J}$, $\hat{\mathcal{F}}^J_b$, et $\hat{\mathcal{F}}^J_b$ après l'injection du signal de tramage local adaptatif. 184
A.13 Les courbes ROC obtenues par $\mathcal{F}^J$ contre des détecteurs criminalistiques. . . . 186
A.14 Les valeurs AUC obtenues à différents taux de remplacement d'image pour différents types d'images contre des détecteurs à base de SVM. 187
A.15 Courbes ROC obtenues par $\mathcal{F}^J_1$ contre des détecteurs criminalistiques. 190
A.16 Courbes ROC obtenues pour $\mathcal{M}$, $\mathcal{F}^M_W$ [WSL13], $\mathcal{F}^M_W$ [WSL13], et $\mathcal{F}^M$ contre des détecteurs scalaires. 195
A.17 Les valeurs AUC obtenues à différents taux de remplacement d'image pour différents types d'images contre des détecteurs à base de SVM. 196

List of Tables

3.1 JPEG forensic detectors. 39
3.2 Notations for original, JPEG, and state-of-the-art anti-forensic JPEG images. 40
3.3 Median filtering forensic detectors. 45
3.4 Notations for original, MF, and state-of-the-art anti-forensic MF images. . . . 46

4.1 Results obtained by detector K_F^Q [FD03] on BOSSBase dataset [BFP11]. . . . 50
4.2 Image quality comparison of different (anti-forensic/post-processed) JPEG images. 59

5.1 Performance comparison of $\hat{\mathcal{F}}_{bq}^J$ and $\hat{\mathcal{F}}_q^J$. 79
5.2 DCT histogram recovery comparison between $\mathcal{F}_{S_q}^J$ [Sta+10a, SL11] and $\hat{\mathcal{F}}_{bq}^J$. . 84
5.3 DCT histogram recovery comparison between $\mathcal{F}_0^J$ and $\mathcal{F}^J$. 84
5.4 Performance comparison of different (anti-forensic/post-processed) JPEG images. 85
5.5 Comparison of computation cost for generating different anti-forensic JPEG images. 87
5.6 Image quality comparison of different (anti-forensic) double JPEG compressed images on **A-DJPG-*R***. 91
5.7 Image quality comparison of different (anti-forensic) double JPEG compressed images on **NA-DJPG-*R***. 93
5.8 Image quality comparison of different (anti-forensic) double JPEG compressed images on **LOC-A-DJPG-15/16-*R***. 94
5.9 Image quality comparison of different (anti-forensic) double JPEG compressed images on **LOC-NA-DJPG-15/16-*R***. 94
5.10 Image quality comparison of different (anti-forensic) double JPEG compressed images on **LOC-A-DJPG-1/2-*R***. 94
5.11 Image quality comparison of different (anti-forensic) double JPEG compressed images on **LOC-NA-DJPG-1/2-*R***. 96
5.12 Image quality comparison of different (anti-forensic) double JPEG compressed images on **LOC-A-DJPG-1/16-*R***. 96
5.13 Image quality comparison of different (anti-forensic) double JPEG compressed images on **LOC-NA-DJPG-1/16-*R***. 96

6.1 Image quality comparison of the post-processed JPEG images. 110
6.2 DCT histogram recovery comparison between $\mathcal{F}_{S_q}^J$ [Sta+10a, SL11] and $\hat{\mathcal{F}}_c^J$. . 113

6.3 Pair-wise DCT histogram recovery and image quality comparison of different images. 114
6.4 DCT histogram recovery comparison between $\hat{\mathcal{F}}^J_{bq}$ and $\hat{\mathcal{F}}^J_c$. 115
6.5 Performance comparison of different (anti-forensic/post-processed) JPEG images. 118

7.1 Image quality comparison of the "salt & pepper" noised, median filtered, and quality enhanced images 135
7.2 Performance comparison of different (anti-forensic/processed) MF images. . . . 137
7.3 Performance comparison of different (anti-forensic/median filtered) resampled images. 147
7.4 Performance comparison of different (anti-forensic/median filtered) JPEG images. 150

A.1 Détecteurs criminalistiques de compression JPEG. 175
A.2 Notations pour l'image originale, compressée JPEG, et anti-criminalistique dans l'état de l'art. 175
A.3 Détecteurs criminalistiques du filtrage médian. 177
A.4 Notations pour l'image originale, filtrée médian, et anti-criminalistique dans l'état de l'art. 177
A.5 La comparaison de la qualité d'image. 181
A.6 Comparaison de la récupération d'histogramme DCT entre $\mathcal{F}^J_{S_q}$ [Sta+10a, SL11] et $\hat{\mathcal{F}}^J_{bq}$. 185
A.7 Comparaison de l'indétectabilité criminalistique et de la qualité d'image. . . . 186
A.8 Comparaison de qualité d'image des images JPEG. 188
A.9 Comparaison de la restauration d'histogramme DCT entre $\hat{\mathcal{F}}^J_{bq}$ et $\hat{\mathcal{F}}^J_c$. 189
A.10 Comparaison de l'indétectabilité criminalistique et la qualité d'image de différentes images. 191
A.11 Comparaison de la qualité de l'image bruitée par le bruit sel et poivre, celle puis filtrée par le filtre médian, et enfin celle traitée par la méthode proposée. 194
A.12 Comparaison de l'indétectabilité criminalistique et de la qualité de différentes images. 194

Notations

Variables

$\mathbf{U}$	Generic image pixel value matrix
$\mathbf{u}$	Vectorized form of matrix $\mathbf{U}$
$\mathbf{X}$	Pixel value matrix of the *original* image
$\mathbf{x}$	Vectorized form of matrix $\mathbf{x}$
$\mathbf{Y}$	Generic pixel value matrix of the JPEG compressed or median filtered image from the original image $\mathbf{X}$
$\mathbf{y}$	Vectorized form of matrix $\mathbf{Y}$

Indices

$(\cdot)_i$	The i-th entry of a vector
$(\cdot)_{i,j}$	The (i,j)-th entry of a matrix
$(\cdot)_{r,c}$	The (r,c)-th entry of an 8×8 matrix
$(\cdot)^l_{r,c}$	The (r,c)-th entry of the l-th 8×8 non-overlapping submatrix of a matrix

Images

$\mathcal{I}$	Original image
$\mathcal{J}$	JPEG image compressed from $\mathcal{I}$ with a certain quality factor
$\mathcal{M}$	Median filtered image obtained from $\mathcal{I}$ with filter window of size 3×3
$\mathcal{R}$	Resampled image using bicubic interpolation with a certain factor

$\mathcal{F}^J_{S_q}$	Stamm *et al.*'s [Sta+10a, SL11] anti-forensic JPEG image created from $\mathcal{J}$ using the dithering operation
$\mathcal{F}^J_{S_q S_b}$	Stamm *et al.*'s [Sta+10b, SL11] anti-forensic JPEG image created from $\mathcal{F}^J_{S_q}$ using the median filtering based deblocking method
$\mathcal{F}^J_V$	Valenzise *et al.*'s [VTT11] anti-forensic JPEG image created from $\mathcal{J}$ using the perceptual anti-forensic dithering operation
$\mathcal{F}^J_{Su}$	Sutthiwan and Shi's [SS11] anti-forensic JPEG image created from $\mathcal{J}$ using the Shrink-and-Zoom attack
$\mathcal{J}_A$	Alter *et al.*'s [ADF05] processed JPEG image created from $\mathcal{J}$ using the Total Variation based deblocking method
$\hat{\mathcal{F}}^J_0$	Intermediate anti-forensic JPEG image created from $\mathcal{J}$ using the proposed Total Variation based JPEG anti-forensic deblocking method described in Section 4.3.1
$\mathcal{F}^J_0$	Anti-forensic JPEG image created from $\hat{\mathcal{F}}^J_0$ using the proposed de-calibration operation described in Section 4.3.2
$\hat{\mathcal{F}}^J_b$	Intermediate anti-forensic JPEG image created from $\mathcal{J}$ using the proposed Total Variation based JPEG anti-forensic deblocking method described in Section 5.2.1
$\hat{\mathcal{F}}^J_{bq}$	Intermediate anti-forensic JPEG image created from $\hat{\mathcal{F}}^J_b$ using the proposed perceptual DCT histogram smoothing method described in Section 5.2.2, with the maximum dimensionality of the assignment problem set to 200
$\hat{\mathcal{F}}'_{bq}$	Intermediate anti-forensic JPEG image created from $\hat{\mathcal{F}}^J_b$ using the proposed perceptual DCT histogram smoothing method described in Section 5.2.2, without the maximum dimensionality limit of the assignment problem
$\hat{\mathcal{F}}^J_q$	Intermediate anti-forensic JPEG image created from $\mathcal{J}$ using the proposed perceptual DCT histogram smoothing method described in Section 5.2.2
$\hat{\mathcal{F}}^J_{bqb}$	Intermediate anti-forensic JPEG image created from $\hat{\mathcal{F}}^J_{bq}$ using the proposed second-round Total Variation based JPEG anti-forensic deblocking method described in Section 5.2.3
$\mathcal{F}^J$	Anti-forensic JPEG image created from $\hat{\mathcal{F}}^J_{bqb}$ using the proposed de-calibration operation described in Section 5.2.4
$\hat{\mathcal{I}}^J$	Quality enhanced JPEG image created from $\mathcal{J}$ using the proposed JPEG post-processing method described in Section 6.2.2
$\hat{\mathcal{F}}^J_c$	Processed JPEG image created from $\hat{\mathcal{I}}^J$ using the proposed calibration based non-parametric DCT histogram smoothing method described in Section 6.3
$\mathcal{F}^J_1$	Anti-forensic JPEG image created from $\hat{\mathcal{F}}^J_c$ using the proposed JPEG anti-forensic method described in Section 6.4

$\mathcal{F}_W^M$	Wu *et al.*'s [WSL13] anti-forensic median filtered image created from $\mathcal{M}$ using the dithering operation
$\mathcal{F}_D^M$	Dang-Nguyen *et al.*'s [DN+13] anti-forensic median filtered image created from $\mathcal{M}$ using the noise injection based method
$\mathcal{M}^p$	Processed median filtered image created from $\mathcal{M}$ using the quality enhancement method proposed in Section 7.3.3
$\bar{\mathcal{M}}$	Processed median filtered image created from $\mathcal{M}$ using the pixel value perturbation proposed in Section 7.3.4.1
$\mathcal{F}'^M$	Processed median filtered image created from $\mathcal{M}$ using the median filtering anti-forensic method proposed in Section 7.3.4
$\mathcal{F}^M$	Anti-forensic median filtered image created from $\bar{\mathcal{M}}$ using the median filtering anti-forensic method proposed in Section 7.3.4
$\mathcal{R}^m$	Median filtered resampled image created from $\mathcal{R}$ with filter window of size 3×3
$\mathcal{R}^w$	Wu *et al.*'s [WSL13] anti-forensic median filtered resampled image created from $\mathcal{R}^m$ using the dithering operation
$\mathcal{R}^d$	Dang-Nguyen *et al.*'s [DN+13] anti-forensic median filtered resampled image created from $\mathcal{R}^m$ using the noise injection based method
$\mathcal{R}^f$	Anti-forensic median filtered resampled image created from $\mathcal{R}^m$ using the median filtering anti-forensic method proposed in Section 7.3.4
$\mathcal{J}^m$	Median filtered JPEG image created from $\mathcal{J}$ with filter window of size 3×3
$\mathcal{J}^w$	Wu *et al.*'s [WSL13] anti-forensic median filtered JPEG image created from $\mathcal{J}^m$ using the dithering operation
$\mathcal{J}^d$	Dang-Nguyen *et al.*'s [DN+13] anti-forensic median filtered JPEG image created from $\mathcal{J}^m$ using the noise injection based method
$\mathcal{J}^f$	Anti-forensic median filtered JPEG image created from $\mathcal{J}^m$ using the median filtering anti-forensic method proposed in Section 7.3.4

Acronyms

AC	Alternating Current (defined on page 32)
A-DJPG	Aligned Double JPEG (defined on page 90)
AUC	Area Under Curve (defined on page 17)
bpp	bits per pixel (defined on page 19)
CFA	Color Filter Array (defined on page 10)
dB	deciBel (defined on page 21)
DC	Direct Current (defined on page 32)
DCT	Discrete Cosine Transform (defined on page 6)
EBPM	Edge Based Prediction Matrix (defined on page 43)
EM	Expectation Maximization (defined on page 108)
EPLL	Expected Patch Log Likelihood (defined on page 105)
FoE	Fields of Experts (defined on page 108)
FPN	Fixed Pattern Noise (defined on page 11)
GLF	Global and Local Feature (defined on page 43)
GMM	Gaussian Mixture Model (defined on page 7)
HMRF	Huber-Markov Random Field (defined on page 108)
HVS	Human Visual System (defined on page 21)
IDCT	Inverse Discrete Cosine Transform (defined on page 33)
IJG	Independent JPEG Group (defined on page 32)
JND	Just-Noticeable Distortion (defined on page 37)
JPEG	Joint Photographic Experts Group (defined on page 4)
KL	Kullback-Leibler (defined on page 15)
MAP	Maximum *a posteriori* (defined on page 6)
MF image	Median Filtered image (defined on page 40)
MFF	Median Filtering Forensics (defined on page 42)

MFLTP Median Filter Local Ternary Patterns (defined on page 44)

MFRAR Median Filter Residual AutoRegressive (defined on page 43)

MFRTP Median Filter Residual Transition Probabilities (defined on page 43)

MLE Maximum-Likelihood Estimation (defined on page 36)

MSE Mean Squared Error (defined on page 21)

NA-DJPG Non-Aligned Double JPEG (defined on page 92)

PCA Principal Component Analysis (defined on page 11)

p.d.f. probability density function (defined on page 98)

PGM Portable GrayMap (defined on page 22)

p.m.f. probability mass function (defined on page 74)

PRNU Photo Response Non-Uniformity (defined on page 10)

PSNR Peak Signal-to-Noise Ratio (defined on page 14)

QCS Quantization Constraint Set (defined on page 6)

RBF Radial Basis Function (defined on page 18)

RGB Red, Green, Blue (defined on page 22)

ROC Receiver Operating Characteristic (defined on page 14)

SAZ Shrink-and-Zoom (defined on page 38)

SIFT Scale-Invariant Feature Transform (defined on page 11)

SPAM Subtractive Pixel Adjacency Matrix (defined on page 38)

SPIHT Set Partitioning In Hierarchical Trees (defined on page 10)

SSIM Structural SIMilarity (defined on page 14)

SVM Support Vector Machine (defined on page 17)

TIFF Tagged Image File Format (defined on page 22)

TV Total Variation (defined on page 6)

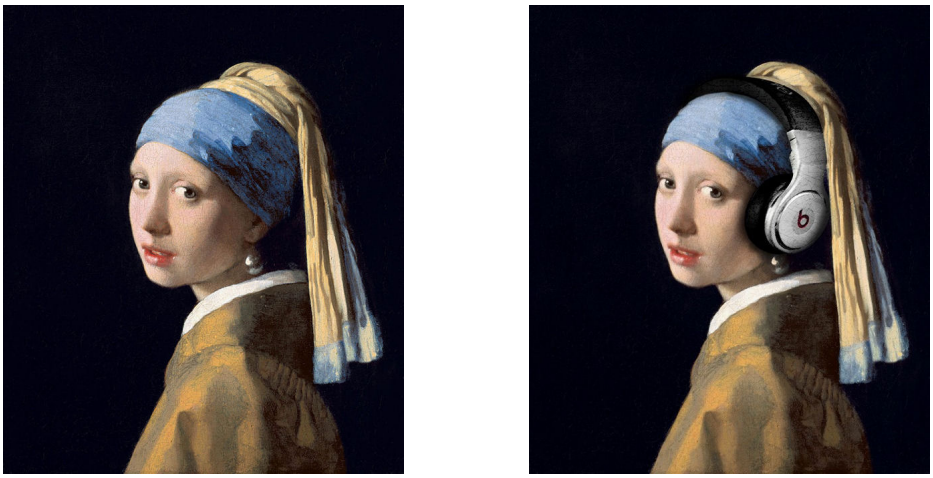
Chapter 1

Introduction

1.1 Can You Believe Your Eyes?

Seeing is believing.
A picture is worth a thousand words.

Seeing is believing, or *a picture is worth a thousand words*, they always say. That is probably why the digital image is one of the most commonly used multimedia types on Internet. According to Mary Meeker's annual Internet trends report [Mar], over 1.8 billion photos are uploaded and shared on Internet per day! Digital images are literally ubiquitous, in which we historically have trust. However, this confidence is being constantly shaken due to the widespread availability of high-quality cameras and powerful photo-editing tools. In Figure 1.1-(b), the Girl with A Pearl Earring, portrayed by the 17-century painter Johannes Vermeer, appears to appreciate the music coming from the beats. Can you believe your eyes?



(a) Original (b) Edited[1]

Figure 1.1: (a) is the Girl with A Pearl Earring painted by Johannes Vermeer around 1665. The beats, a product of modern technologies, were integrated into the digital version of this classic painting by the *Worth1000* website user *bigchopper* in (b).

[1]Downloaded from: http://www.worth1000.com/entries/740270/girl-with-the-beats.

It can be seen that nowadays creating visually plausible fake images has become a less and less challenging task. People's trust in digital images is being gradually eroded by the development of modern information technologies. Doctored images, which are able to fool human naked eyes, are appearing with a growing frequency. Unfortunately, not every doctored image is as "innocent" as the one shown in Figure 1.1-(b), which may be mainly for amusement. The image editing can be malicious, to be used for instance in political and personal attacking. For example, Fourandsix Technologies, Inc. maintains an image gallery[2] which collects notable photo manipulations throughout history. Among these tampered pictures, some were abominable enough to have led to severe financial losses and even have brought negative impacts to the society.

The doubt thrown upon digital images has urged the development of **digital image forensics**, trying to restore some trust to digital images. The main objectives of forensics are to analyze a given digital image so as to detect whether it is a forgery, to identify its origin, to trace its processing history, or to reveal latent details invisible to human naked eyes [Fou].

During the last decade, researchers have proposed various image forensic techniques. In the early stage, fragile digital image watermarking was a popular choice for image authentication purposes. Fragile watermarking in literature is regarded as the so-called *active* forensics [Con11]. It actively embeds the authentication information (*i.e.*, the watermark) into the image when it is captured or before its transmission. Thereafter, the image can be authenticated if the extracted watermark matches the embedded one; otherwise the failure of watermark extraction or any mismatch between the extracted and embedded information can serve as evidence of tampering. To this end, a special image acquisition device is required. In fact, the idea of trustworthy camera equipped with a watermarking system was proposed as early as 1993 [Fri93]. However, its realization in industry encountered many difficulties, which currently still remain hard to be resolved. Firstly, it is hard for different camera manufacturers to reach an agreement on a common standard protocol. In addition, the consumers may find it unacceptable regarding the visual quality decrease of the watermarked image. Furthermore, once the inside watermarking system of the so-called trustworthy camera is hacked, its security will become a very problematic matter. One example of attempt in industry is the Aigo V80PLUS camera [Aig] marketed by Beijing Huaqi Information Digital Technology Co., Ltd. in 2005. Inside this camera, a digital watermarking system is included, which embeds the authentication information into the image at the time of recording. Yet, it did not bring a popularization of the trustworthy camera, due to the previously described concerns.

Aware of the limitations of active forensics, researchers are gradually shifting their attention to the so-called *passive* forensics [Far09a, Con11]. Compared with the image authentication based on digital watermarking, the passive forensic techniques seeks to assess the authenticity of a given image in a blind way, without resorting to any *a priori* embedded information (*e.g.*, a watermark). The assumption here is that image manipulation may create forgeries without leaving visual traces, yet it will probably disturb the intrinsic properties of the authentic image. Therefore, tampering can be detected by examining the inconsistency/deviation of underlying statistics of an image. In literature, "passive forensics" is often directly referred to

[2] Available at: http://www.fourandsix.com/photo-tampering-history/.

as "forensics". In the following of this thesis, we will also omit the adjective "passive" for the sake of brevity.

Digital forensics has become a hot research topic during recent years. Figure 1.2 shows the variation trend of the annual number of IEEE publications [Iee] related to forensics, starting from 2000. In the figure, the gray bar stands for publications with the keyword "forensics", whereas the hatched lightgray bar is for the publications with the keywords "forensics" and "image". It can be seen that forensics has received an increasing attention in the last decade. Among the forensics research, as an important branch image forensics takes up around 40% of the total relevant publications.

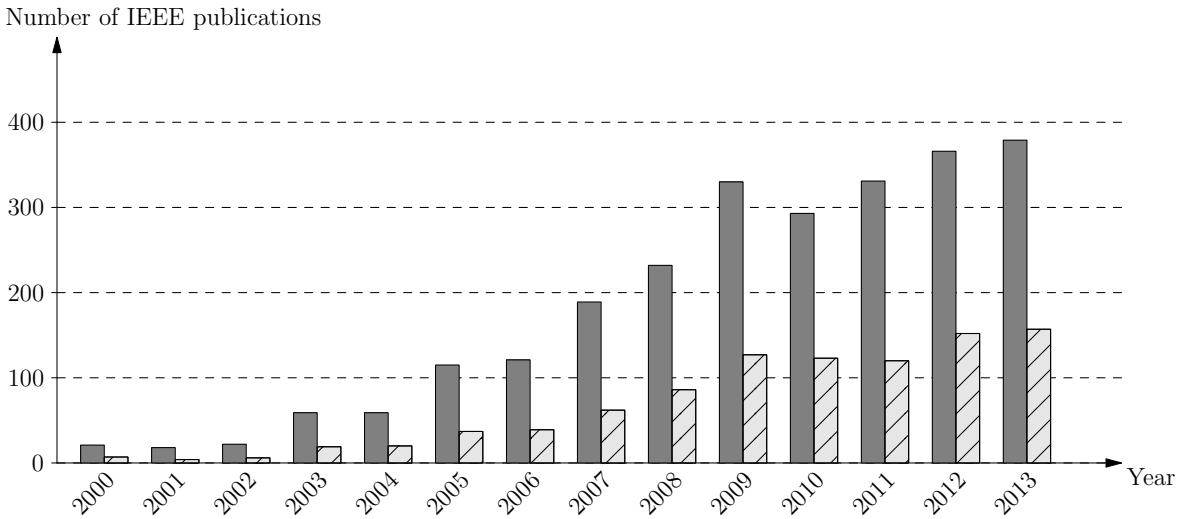


Figure 1.2: Annual number of IEEE publications whose keywords include "forensics" (the gray bar), and both "forensics" and "image" (the hatched lightgray bar).

1.2 Image Anti-Forensics

Every coin has two sides.

Similar to cryptography *vs.* cryptanalysis and steganography *vs.* steganalysis, the two sides of the coin regarding image authentication are forensics and anti-forensics. **Digital image anti-forensics** is the technique to expose the limitations of forensic methods, with the ultimate goal to develop more trustworthy forensics [BK13]. In literature, anti-forensics is also synonymously referred to as counter-forensics. This research direction is important, since we have to design against possible anti-forensics for reliable forensics. The main objective of image anti-forensics is to perform certain operations to digital images, disguising traces left by image editing so that they will no longer be detected by forensic algorithms.

Image forensics is still at its early stage, and image anti-forensics is an even younger research topic [KB07]. In literature, the publications of forensics largely outnumber those of

anti-forensics. Moreover, existing anti-forensic methods often use simple image processing to disguise traces left by a targeted operation, *e.g.*, using filtering to hide compression artifacts [SL11], or using noise addition to disguise footprints left by filtering [WSL13]. Indeed, such anti-forensic methods can be very successful in defeating the targeted forensic detectors, but can also be easily exposed by more advanced detectors. Moreover, the anti-forensic image generated by these methods often suffers from a low visual quality. This is a worrying issue, since an image of low quality (*e.g.*, a blurry/noisy image) may spontaneously rouse the suspicion of its authenticity.

In summary, image anti-forensics has a two-fold end: a good forensic undetectability as well as a high visual quality of the processed image [KR08]. Between these two goals, the forensic undetectability is more important than the image quality for image anti-forensics. Anti-forensics cannot claim to be successful if there exist a certain detector which is able to detect the anti-forensic images.

1.3 Objectives and Contributions

In this thesis, we stand on the image *anti-forensic* side, with the focus on *JPEG* (Joint Photographic Experts Group) *compression* and *median filtering* anti-forensics. From a JPEG compressed or median filtered image, if we can successfully create a "fake" image that appears never processed, it would be a relatively easy task to conduct image processing history falsification or even tampering afterwards. To this end, we employ some frameworks from *image restoration* field meanwhile integrating some anti-forensic terms/strategies, for creating anti-forensic images with a good tradeoff between forensic undetectability and image quality.

1.3.1 JPEG Compression and Median Filtering

Firstly, we choose to conduct our research on *image anti-forensics to JPEG compression*, because JPEG is probably the most common image format in use today on Internet, and can be easily found in various forensic scenarios. According to the statistics of the usage of image file formats for websites on December 8, 2014, JPEG is the most widely used one, with the usage by 68.7% of all the websites [W3t].

We can imagine the following forgery creation scenario involving JPEG compression, as illustrated in Figure 1.3. In order to make people believe that someone has participated in a certain event, the forger will probably create a composite with the scene and the person from two JPEG images with different quality factors[3]. The resulting image is likely to be JPEG compressed again before publishing. Careful image editing may leave no visual clues noticeable by human naked eyes, yet the forgery can be exposed by detecting different kinds of double JPEG compression artifacts present in different areas of the image [BP12b].

[3]The quality factor is an integer between 1 and 100. The greater the quality factor is, the higher the quality of the compressed image is, and the larger the JPEG file will be. A more detailed description of the quality factor can be found in Section 3.1.1.

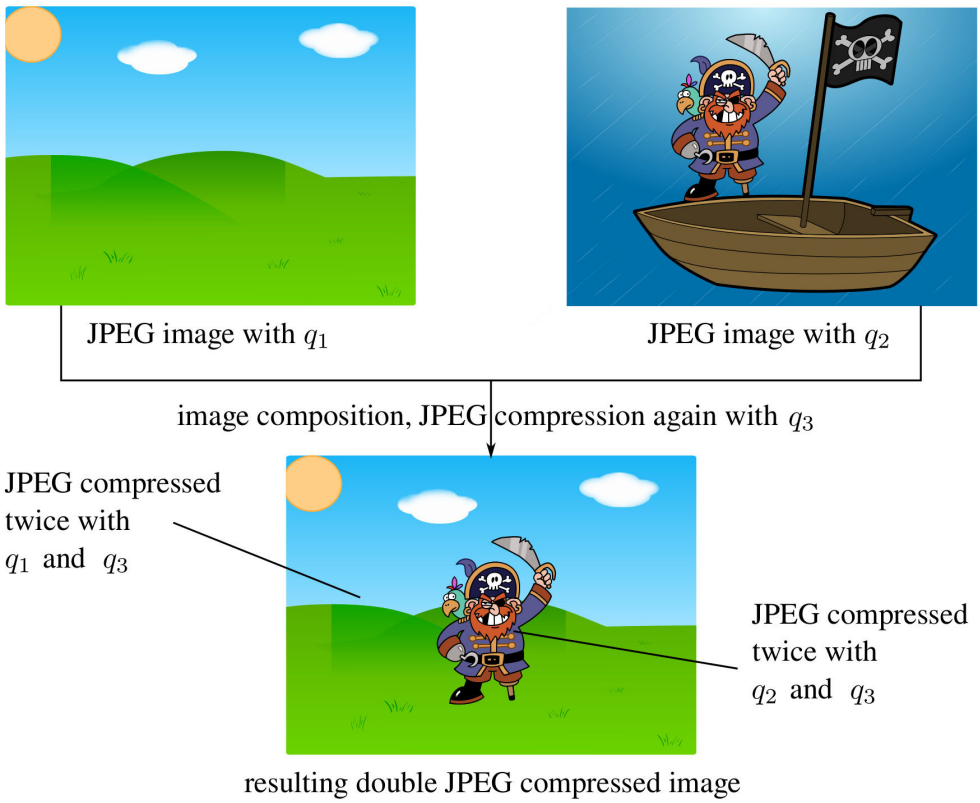


Figure 1.3: Illustration of creating a composite JPEG image. Here q_1, q_2 and q_3 are three possibly different quality factors for JPEG compression.

Generally speaking, the goal of JPEG anti-forensics is to *remove all possible footprints left by JPEG compression*, so that the resulting anti-forensic JPEG image looks as if it were an original uncompressed one. Now, count for the possibility that a smart forger employs JPEG anti-forensics to hide JPEG compression artifacts of the two source JPEG images so that they seem never compressed. Therefore, no (different types of) JPEG compression artifacts indicating image manipulation will appear in the composite image forgery. Thereafter, the image can be re-compressed using another compression setting, so as to either hide traces of double JPEG compression, or falsify the image origin, or serve for other possible anti-forensic purposes [SL11].

In the very recent work of JPEG anti-forensics [SL11] as well as the image resampling anti-forensics conducted in [KR08], median filtering shows its destructive nature to other image processing footprints. However, the median filter, as a well-known and widely used image denoising and smoothing operator, leaves trackable clues in the image. They can be detected by various median filtering forensic detectors. The presence of median filtering traces, not only suggests the image has been previously median filtered, but also implies the possibility that other image processing operations may have been applied to the image. Hence, it is of great significance to conduct *median filtering anti-forensics*, which constitutes the second research subject of this thesis.

1.3.2 Image Anti-Forensics and Image Restoration

When it goes to image coding or processing, we notice that image anti-forensics to some extent shares a similar goal to *image restoration*, which is to recover the information lost during the image degradation, via solving an ill-posed inverse problem. Indeed, for certain anti-forensic scenarios, *e.g.*, attacking physically based or geometric-based forensic algorithms (see Section 2.1.1 for their brief descriptions), this similarity no longer holds. However, their relevant anti-forensic study is beyond the scope of this thesis, and we mainly focus on JPEG compression and median filtering anti-forensics here. The objective of image restoration usually goes to the visual quality improvement of degraded images. While image anti-forensics hopes to recover the underlying statistics of the original, genuine image to the utmost, so that the resulting image forgery appears to be authentic. High visual quality of the processed image is certainly one important goal of image anti-forensics. Besides, it is also worth noticing that image anti-forensics has an additional indispensable goal, *i.e.*, a good forensic undetectability, compared with image restoration.

Given the similarities between image anti-forensics and image restoration, this thesis aims to generate, from a JPEG compressed or median filtered image, a "natural" image which appears to be never processed. To this end, the maximum *a posteriori* (MAP) estimation (or one of its variants) is employed. Under this framework, several different natural image statistical models from image restoration are adopted and enriched by integrating some anti-forensic terms/strategies concerning the forensic undetectability. In order to solve different proposed image anti-forensic problems, several *numerical optimization* methods are used. By doing so, we hope to be able to estimate the "best" anti-forensic image in some sense. At last, the anti-forensic images, with a good tradeoff between forensic undetectability and image quality, are generated.

1.3.3 Methodology

Different from the state-of-the-art JPEG/median filtering anti-forensics work based on simple image processing [SL11, WSL13], here a new research line is proposed to conduct image anti-forensics. In this thesis, we propose sophisticated digital image anti-forensic methods to JPEG compression and median filtering, leveraging on advanced concepts and tools from image restoration, natural image statistics and numerical optimization.

The blocking artifacts in the spatial domain and the quantization artifacts in the DCT (Discrete Cosine Transform) domain of one image are both evidence of JPEG compression [FD03]. In this thesis, the following methods are developed for JPEG anti-forensic purposes:

- Firstly, a constrained Total Variation (TV) based minimization [ADF05] is employed for removing the JPEG blocking artifacts. Moreover, in order to ensure the visual quality of the final obtained anti-forensic JPEG image, a modified Quantization Constraint Set (QCS) projection [RS05] is used.

- For the quantization artifacts in the DCT domain, a perceptual DCT histogram smoothing method is proposed based on the local Laplacian model and the partly recovered DCT-domain information obtained after applying the TV-based deblocking.
- Furthermore, in order to study the impact of a more advanced image prior model to the JPEG anti-forensics task, the Gaussian Mixture Model (GMM) for overlapping image patches [ZW11] and a likelihood term modeling the JPEG compression process [RS05] are considered. Besides, we also propose a new, non-parametric method to DCT histogram smoothing based on calibration [FGH02].

As to median filtering anti-forensics, this thesis proposes an image variational deconvolution framework, inspired by the literature on image deconvolution [KF09, KTF11]. This optimization-based framework consists of a convolution term approximating the median filtering process, a fidelity term with respect to the median filtered image, and an image prior term based on the generalized Gaussian distribution in the pixel value difference domain.

In order to validate the efficiency of the proposed JPEG/median filtering anti-forensic methods, large-scale forensic tests are carried out. Experimental results demonstrate that the proposed methods outperform the state-of-the-art anti-forensics, with a better forensic undetectability against existing forensic detectors as well as a higher visual quality of the anti-forensic image.

1.4 Outline

The remainder of this thesis[4] is organized as follows.

Chapter 2 presents some background knowledge on image forensics and anti-forensics, including the classification, evaluation metrics, natural image datasets used in forensic testing of this thesis, and relevant optimization methods which will be used in the proposed image anti-forensic methods.

Chapter 3 reviews the state-of-the-art image forensic algorithms and anti-forensic methods to JPEG compression and median filtering. The reviewed forensic algorithms are the attacking targets of the proposed anti-forensic methods, while the reviewed anti-forensic methods are used for experimental comparisons.

Chapter 4 proposes a JPEG deblocking method, by optimizing a constrained TV-based minimization problem, whose cost function is composed of a TV term and a TV-based blocking measurement term. Besides a good deblocking effect, the resulting anti-forensic JPEG

[4] In this thesis, most of the contents in Chapters 4-7 were published or have been accepted for publication in international conferences or international journals. As to JPEG anti-forensics, we published three papers [Fan+13a, Fan+14, Fan+13b] where different divisions of datasets, compression quality factors and evaluation metrics were used. For consistency consideration of the thesis, we will use the same setting (see Sections 2.2 and 2.3 for details) across Chapters 4-6. Therefore, the relevant figures and tables may not be exactly the same, but should be in accordance with those shown in our published papers.

image also achieves relatively good forensic undetectability even against quantization artifacts detectors. Yet, the DCT-domain quantization artifacts still to some extent exist, which may be used by potential forensic detectors. Therefore, this method will be further improved in Chapter 5.

Chapter 5 describes an improved JPEG anti-forensic method based on the work in Chapter 4 yet with a different parameter setting. The remaining comb-like quantization artifacts in the TV-based deblocked JPEG image are explicitly filled by a perceptual DCT histogram smoothing procedure.

Chapter 6 presents a JPEG quality enhancement method based on an MAP-based framework using the GMM as the image prior model for the overlapping image patches. For JPEG anti-forensic purposes, the DCT histogram smoothing is performed using a new, non-parametric method based on calibration.

Chapter 7 proposes a median filtered image quality enhancement approach as well as a median filtering anti-forensic method, by approximating the median filtering process using image convolution and by using the generalized Gaussian to model the pixel value difference.

Chapter 8 concludes this thesis, by summarizing the contributions and proposing several perspectives about the future research work on digital image anti-forensics.

Chapter 2

Preliminaries

Contents

2.1 Classification of Image (Anti-)Forensics 10
2.1.1 Farid's Classification of Image Forensics 10
2.1.2 Redi *et al.*'s Classification of Image Forensics 10
2.1.3 Piva's Classification of Image Forensics 11
2.1.4 Stamm *et al.*'s Classification of Image Forensics 12
2.1.5 Böhme and Kirchner's Classification of Image Anti-Forensics 13
2.1.6 Classification of Proposed Anti-Forensic Methods 14
2.2 Evaluation Metrics 14
2.2.1 Forensic (Un)detectability 15
2.2.1.1 Scalar-Based Detectors 17
2.2.1.2 SVM-Based Detectors 18
2.2.2 Image Quality 21
2.2.2.1 PSNR 21
2.2.2.2 SSIM 21
2.2.3 Histogram Recovery 22
2.3 Natural Image Datasets 22
2.3.1 JPEG Forensic Testing 22
2.3.2 Median Filtering Forensic Testing 24
2.4 Relevant Optimization Algorithms 25
2.4.1 Subgradient Method 25
2.4.2 Hungarian Algorithm 26
2.4.3 Half Quadratic Splitting 27
2.4.4 Split Bregman Method 28

This chapter firstly presents the basic knowledge of digital image forensics and anti-forensics, including the classification and evaluation metrics. After that, the natural images datasets used in the forensic testing of this thesis are introduced. Finally, we briefly review the optimization algorithms which will be used for solving the optimization problems proposed in this thesis.

2.1 Classification of Image (Anti-)Forensics

2.1.1 Farid's Classification of Image Forensics

Digital image forensics makes the assumption that the forgery creation process has disturbed a certain kind of intrinsic scene/image properties (*e.g.*, statistical, physical or geometrical properties). In this context, Farid [Far09a] groups digital image forensics into the following five categories:

- *Pixel-based* image forensics analyzes pixel-level anomalies caused by image tampering. Some frequently used image manipulation means are, for instance, copy-and-paste, splicing, resampling and median filtering. Targeting at each of these image operations, various forensic techniques are proposed.

- *Format-based* image forensics detects the image statistical change introduced by a certain lossy compression method. Popular image compression algorithms include JPEG based on the DCT transform, SPIHT (Set Partitioning In Hierarchical Trees) and JPEG2000 based on the wavelet transform, *etc.*

- *Camera-based* image forensics studies digital image ballistics [Far06] from the imaging stage inside the camera. Typical forensic methods in this category are for instance based on the chromatic aberration, the color filter array (CFA), the photo response non-uniformity (PRNU) noise, *etc.*

- *Physically based* image forensics examines anomalies of interaction between objects, light, and the camera in the 3-dimensional physical world. For example, the consistencies of light direction or of lighting environment estimated from different physical objects can be used as criteria for forensic purposes.

- *Geometric-based* image forensics measures the positions of physical objects with respect to the camera. For instance, image tampering can be detected if across the image there exist inconsistencies in the principal point[5].

Despite that [Far09a] mainly reviews image forensics, Farid also points out that new techniques (*i.e.*, anti-forensics) will be developed to create fake images which are harder to be detected. The arms race between forensic analysts and forgers is inevitable.

2.1.2 Redi *et al.*'s Classification of Image Forensics

In [RTD11], Redi *et al.* review image forensic methods in the following two categories:

[5]The principal point is the projection of the camera center onto the image plane.

- *Image source device identification* is to identify the device which is used for the acquisition of the given image. Relevant source identification methods can be further grouped into the following three subcategories:
  - *Identification through artifacts produced in the acquisition phase*, such as forensic methods based on the chromatic aberration, the CFA, *etc.*
  - *Identification through sensor imperfections*, such as forensic methods based on pixel defects, the fixed pattern noise (FPN), the PRNU, *etc.*
  - *Source identification using properties of the imaging device*, such as forensic methods based on the color processing, the JPEG compression, *etc.*
- *Tampering detection* is to expose the intentional image manipulation which modifies the semantic meaning of the image. Relevant tampering detection methods can be further grouped into the following three subcategories:
  - *Detecting tampering performed on a single image*, in one of the most common ways, is to expose the copy-move of a region within an image. Some relevant known forensic methods leverage on the tools such as the principal component analysis (PCA), the scale-invariant feature transform (SIFT), *etc.*
  - *Detecting image composition*, in other words, is to expose image splicing. Inconsistencies of certain properties across different parts of the image can all serve as evidence of tampering, such as the light direction, the complex lighting environment, shadows/reflections of objects, the PRNU, the JPEG compression, *etc.*
  - *Tampering detection independent on the type of forgery*, is a general technique which covers both tampering involving a single image and tampering involving multiple images. In order to realize this general forensic method, possible ways are to exploit the resampling traces, compression artifacts, acquisition device fingerprints, *etc.*

Meanwhile, Redi *et al.* [RTD11] also point out that image anti-forensics is the new phase for developing new and more powerful forensic methods.

2.1.3 Piva's Classification of Image Forensics

In [Piv13], Piva surveys digital image forensics according to the image life cycle, which are divided into three stages: image acquisition, image coding, and image editing. Digital image forensics can therefore be divided into the following three categories:

- *Acquisition footprints* based image forensics. When a digital image is recorded by a camera, each imaging stage introduces intrinsic footprints due to the imperfection of the device (*e.g.*, PRNU noise caused by the image sensor) or because of the different camera manufacturer choices in both the hardware (*e.g.*, different lens leading to different chromatic aberration parameters) and software (*e.g.*, different CFA configurations). These image artifacts vary according to different camera brands and/or models, and can be

considered as the signature of a specific camera type (*i.e.*, the so-called *image ballistics* defined in literature [Far06]). Inconsistencies in these footprints across the image can be considered as evidence of tampering.

- *Coding footprints* based image forensics. Lossy compression is widely used to reduce image redundancy for efficiently storing and transmitting the data. Different coding architectures leave different telltale footprints, which can be used for image forensic purposes. In literature, JPEG compression probably draws the most attention in this category. Researchers also notice that artifacts present in a single JPEG compressed image (*i.e.*, compressed only once) will vary when the JPEG compression is applied again. Image manipulation can therefore be exposed when inconsistencies of coding footprints are detected.

- *Editing footprints* based image forensics. Careful image editing may not leave visual traces, yet will probably disturb the intrinsic properties of the authentic image. Therefore, inconsistency/deviation of these intrinsic properties across the image can be considered as evidence of tampering.

Besides, Piva also provides a brief review and discussion concerning image anti-forensics in [Piv13], based on Böhme and Kirchner's [BK13] classification of image anti-forensics (to be described in Section 2.1.5). Almost all the existing image anti-forensic methods are designed to target one particular forensic tool. Conversely, universal anti-forensics aims to maintain image statistics which are not known to the image forger. It is certainly a more difficult problem and an interesting open research question, but should be able to escape the eternal loop between targeted forensics and anti-forensics.

2.1.4 Stamm *et al.*'s Classification of Image Forensics

Stamm *et al.* [SWL13] review the information forensics in the last decade. Concerning image forensics, the following two aspects are discussed in detail:

- *Detection of tampering and processing operations.* In many scenarios, the processing history of the image is one of the primary concerns to determine whether it can be trusted. This can be identified by exploiting intrinsic properties of the digital content. Relevant forensic techniques can be further divided into the following five subcategories:
  - Forensics based on statistical classifiers;
  - Forensics detecting device fingerprints;
  - Forensics exposing manipulation fingerprints;
  - Forensics examining compression and coding fingerprints;
  - Forensics checking physical inconsistencies.

- *Device forensics.* Digital images have seen a huge growth because of the advancement of digital cameras. Identifying the acquisition device of an image is an important step

to ensure the security and trustworthiness. Relevant forensic techniques can be further divided into the following four subcategories:

- Forensics exploring color processing traces;
- Forensics linking to individual device units;
- Forensics identifying imaging type;
- Forensics detecting manipulation using device fingerprints.

Meanwhile, Stamm *et al.* [SWL13] also present a brief overview of some current image anti-forensic work. More specifically, there exist image anti-forensic techniques based on PRNU, resampling, compression, *etc.*

2.1.5 Böhme and Kirchner's Classification of Image Anti-Forensics

In [BK13] Böhme and Kirchner divided image anti-forensic techniques into two categories along the following three dimensions, respectively:

- *Robustness vs. Security.* In general, image forgers can exploit robustness or security weaknesses of image forensics for anti-forensic purposes.
  - The *robustness* of image forensics is its reliability under legitimate post-processing. For instance, many forensic algorithms (lighting-based forensics is a good example of exception here) fail when strong JPEG compression is applied. Such legitimate post-processing can serve as an anti-forensic technique, as long as it is able to move the plausible forgeries outside the detection region of the forensic detectors.
  - The *security* of image forensics shows how much it is able to expose intentionally concealed illegitimate post-processing. That is to say, the security indicates the ability to withstand anti-forensics. Image forgers can exploit the weaknesses of the image model used by forensics. This more powerful anti-forensic attack creates image forgeries which are moved in a particular direction outwards the decision region of the authentic images.
- *Post-Processing and Integrated Attacks.*
  - *Post-Processing* anti-forensic attacks edit the image, as an additional processing step, such that it does not leave traces which can be detected by image forensics.
  - *Integrated* anti-forensic attacks directly interfere the image generation process. They do not address the robustness of image forensics, by definition.
- *Targeted and Universal Attacks.*
  - If the anti-forensic method exploit the weaknesses of a specific forensic tool, it is *targeted*. It is possible for this kind of anti-forensics to be detected by other forensic algorithms using alternative/improved image models.

– *Universal* anti-forensic attacks try to create image forgeries whose statistical properties are maintained as much as possible, so the fake images remain undetectable even when examined by unknown forensic tools. This is clearly a more difficult task and an interesting open research problem. The main difficulty here is whether we can find a good enough image model which is able to resist combined analysis of forensic algorithms.

2.1.6 Classification of Proposed Anti-Forensic Methods

In this thesis, from the image restoration field, we borrow some MAP (or one of its variants) based frameworks meanwhile integrating some anti-forensic terms/strategies concerning the forensic undetectability, in order to cope with the anti-forensics problems to JPEG compression and median filtering. According to Farid's classification [Far09a] of digital image forensics, JPEG forensics is *format-based*, whereas median filtering forensics is *pixel-based*. Based on Redi *et al.*'s classification [RTD11], JPEG compression can be involved in both *source identification* and *tampering detection*, and median filtering forensics belongs to *tampering detection*. According to Piva's classification [Piv13], JPEG forensics analyzes the *coding footprints*, while median filtering forensics detects *editing footprints*. However, for both JPEG and median filtering forensic techniques, they detect *tampering and processing operations* [SWL13]. More specifically, JPEG forensics examines *compression and coding fingerprints*, and median filtering forensics exposes *manipulation fingerprints*. According to Böhme and Kirchner's classification [BK13] of image anti-forensics, the proposed JPEG and median filtering anti-forensic methods are *post-processing/targeted* attacks analyzing the *security* of forensic algorithms.

2.2 Evaluation Metrics

As noted in Chapter 1, image anti-forensics has a two-fold goal: a good forensic undetectability and a high visual quality of the anti-forensic image. Forensic undetectability (on the anti-forensic side) is a metric versus forensic detectability (on the forensic side). Both of them can be measured by analyzing the ROC (Receiver Operating Characteristic) curve (to be described in Section 2.2.1). Good forensic undetectability means good performance of the anti-forensic method, and poor performance of the forensic algorithm. Conversely, good forensic detectability means poor performance of the anti-forensic method, and good performance of the forensic algorithm.

Image anti-forensics consists in creating visually plausible fake images. Indeed, the forensic undetectability is the priority objective of image anti-forensics. Meanwhile, the visual quality is also an important factor to evaluate the anti-forensic performance, since the low quality image may spontaneously rouse the concerns of its authenticity. The image quality evaluation can be performed by using the well-known PSNR (Peak Signal-to-Noise Ratio) and SSIM (Structural SIMilarity) metrics described in Section 2.2.2.

Besides, JPEG/median filtering anti-forensics often involves certain histogram recovery (see Sections 5.3.1 and 7.3.4 for details). In order to evaluate the histogram restoration, we adopt the Kullback-Leibler divergence [KL51] (in short, KL divergence), which will be described in Section 2.2.3.

2.2.1 Forensic (Un)detectability

In most forensic scenarios, digital image forensics builds a binary classifier[6]. Given a digital image, the forensic detector makes a decision to say that it is either genuine or fake. In other words, the main objective of digital image forensics is to establish a detector, which is able to differentiate genuine images from image forgeries with a high detection accuracy. In this thesis, we study JPEG compression and median filtering and try to disguise the compressed/filtered image as never processed. Here, the genuine image refers to the original, unprocessed image. Whereas the image forgery refers to the processed image, more specifically, the (anti-forensic) JPEG image described in Chapters 4-6, and the (anti-forensic) median filtered image described in Chapter 7.

The performance of a forensic algorithm or an anti-forensic method can be evaluated by comparing the forensic detector's classification outcome obtained on a set of images with the ground-truth. Let us consider a forensic testing with N_N genuine images (*i.e.*, the unprocessed images, here considered as *negative* samples) and N_P fake images (*i.e.*, the processed images, here considered as *positive* samples). Among the N_N negative samples, let N_{FP} denote the number of negative samples which are falsely classified as positive. Among the N_P positive samples, let N_{TP} denote the number of positive samples which are correctly classified as positive. For each classification strategy of a forensic detector, a false positive rate (R_{FP}) and a true positive rate (R_{TP}) can therefore be calculated as follows:

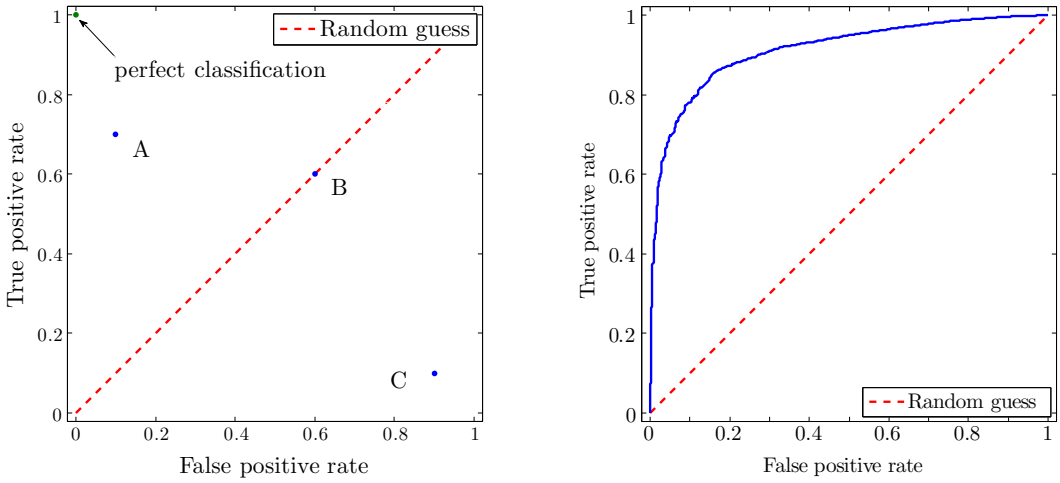
$$R_{FP} = \frac{N_{FP}}{N_N}, \quad R_{TP} = \frac{N_{TP}}{N_P}. \tag{2.1}$$

With (R_{FP}, R_{TP}) pairs obtained by different classification strategies of a forensic detector, a ROC curve can be plotted in the ROC space. The final classification strategy is determined by analyzing this ROC curve. Usually it is required that the false positive rate should not be too high. In layman's terms, we should not wrong an innocent person (*i.e.*, classify a genuine image as fake in digital image forensics), which is critical in some scenarios such as the law enforcement. Under this criterion, the detector chooses the ROC point corresponding to the biggest true positive rate with the constraint that the false positive rate is smaller than a certain threshold. In this context, many anti-forensic methods aim to lower the true positive rate of a detector when the false positive rate is below a certain threshold. Yet, the more comprehensive way is still to analyze the trend of the whole ROC curve.

Figure 2.1-(a) illustrates the ROC space, which is defined by false positive rate R_{FP} and

[6]Indeed, in certain forensic cases, *e.g.*, image source identification, may require a multi-class classifier. However, it is beyond the scope of this thesis and will not be discussed here.

true positive rate R_{TP} as x and y axes respectively. If the upper-left point $(0, 1)$ in the ROC space is included in a ROC curve, it indicates that the corresponding forensic detector is able to achieve the so-called *perfect classification*. In this case, the detector can successfully discriminate between genuine and fake images without any error. This is ideal for forensics, yet very hard to achieve in practice. In general, the goal of the forensic detector is to drag the ROC curve towards the perfect classification point of the ROC space to the utmost. An example can be seen at point A in Figure 2.1-(a). Here, the forensic detector is able to achieve a relatively high true positive rate with a relatively low false positive rate.



(a) ROC space (b) ROC curve

Figure 2.1: The ROC space and an example of ROC curve.

On the anti-forensic side, the forger hopes to create image forgeries so that the ROC curve achieved by a forensic detector is close to the diagonal line (*i.e.*, the *random guess* line) of the ROC space. In this case, the detector is not able to tell the image forgeries from the genuine images even when the "best" classification strategy is used. For example, at point B in Figure 2.1-(a), the detector's classification outcome is equivalent to a random guess. That is to say, given any image, the chances of the detector saying it is either genuine or fake are equal. It is equivalent to randomly deciding whether a given image is genuine/fake. This is the worst classification outcome of a detector. Statistically, the forensic features of the genuine images and of the image forgeries are well mixed together, which are non-distinguishable by the forensic detector.

The point C in Figure 2.1-(a) indicates that the detector achieves a quite low true positive rate at a quite high false positive rate. In this case, the classification outcome appears to be even worse than that of point B on the random guess line. Yet, this does not imply that the forensic feature is not well designed. Conversely, it indicates that the classification strategy is not appropriate. This issue can be resolved by simply flipping the classification strategy, *i.e.*, classifying the image deemed as forgery in the original classification strategy as genuine, meanwhile classifying the image deemed as genuine image in the original classification strategy as fake. Therefore, the ROC point in the bottom right corner of the ROC space will switch

to the top left corner.

With multiple ROC points, a ROC curve can be plotted in the ROC space. An example can be found in Figure 2.1-(b). It is more comprehensive to analyze the trend of the whole ROC curve than a single ROC point, for evaluating the forensic undetectability of the anti-forensic method or the forensic detectability of the forensic algorithm. For a quantitative evaluation, we use the popular AUC (area under curve) metric, by measuring the area under the ROC curve. The AUC value varies in the range $[0, 1]$. When the AUC value is close to 1, it means that the ROC curve is close to the perfect classification point of the ROC space, indicating a good forensic detectability and a poor forensic undetectability. When the AUC value is close to 0.5, it means that the ROC curve is close to the random guess line, indicating a good forensic undetectability and a poor forensic detectability. When the AUC value is close to 0, it means that the ROC curve is close to the bottom-right point of the ROC space. This does not really mean that the forensic detectability of the detector is poor. As stated above, the AUC value can reach close to 1 when the classification strategy is flipped.

The forensic testing requires a set of images including both genuine and fake images, as well as a forensic detector. In the forensic experimental setting of this thesis, certain image dataset (described in Section 2.3) containing genuine images is used. For each genuine image, it is processed to generate a corresponding image forgery. At last, an equal number of genuine images and image forgeries are used for forensic testing. On the other hand, the feature of a forensic algorithm can be either a scalar or a vector, depending on the specific method. Their corresponding forensic tests are to some extent different. In the image forensics literature where the feature is vector-based, it is a common practice to adopt the SVM (Support Vector Machine) for training the forensic detector. In Sections 2.2.1.1 and 2.2.1.2, we will describe how to conduct the forensic test for forensic (un)detectability evaluation, using scalar-based detectors and SVM-based detectors, respectively.

2.2.1.1 Scalar-Based Detectors

Given an image, the scalar-based forensic algorithm outputs a feature value. The forensic detector classifies the image as either genuine or fake, by comparing this feature value with a pre-defined threshold. If the feature value is greater (or smaller) than the threshold, the image is classified as fake. Otherwise, the image is classified as genuine.

The threshold can vary, thereafter producing different classification outcomes. Then different classification strategies of the detector are defined by different thresholds. Indeed, the threshold can be defined as any value. In the experiments of forensic testing, the common practice is as follows. The detector outputs the feature value for each image under examination. Then, all these obtained feature values (or their uniformly sampled values) are used as the different thresholds for producing different classification strategies of the detector. Thereafter, a ROC curve can be plotted and an AUC value can be calculated.

2.2.1.2 SVM-Based Detectors

Many forensic features are vector-based. In the image forensics literature, probably the most common way to build a forensic detector based on a vector-based feature is to use a two-class SVM[7]. In this thesis, we follow this common practice of adopting the SVM in image forensics.

For the feature vectors computed from the positive and negative samples, the SVM builds a maximum-margin hyperplane that achieves the largest distance to the nearest training feature vector of either of the two classes. On the two sides of the separating hyperplane, two parallel supporting hyperplanes are selected in a way that there are no feature vectors between them (namely, hard margin) and their distance is maximized. The region bounded by these two hyperplanes is the so-called "*margin*". The feature vectors on the margin are, namely, the "*support vectors*".

In practice, it happens that certain training samples are mislabeled, *i.e.*, some positive samples are labeled negative, or some negative samples are labeled positive. In order to control the sensitivity of the SVM to noisy data, the so-called *soft-margin* method (with respect to the hard-margin one described earlier) is introduced. It allows that a few training samples fall into the margin but are on the correct side of the separating hyperplane, or that some training samples are even on the wrong side of the separating hyperplane. For the soft-margin SVM, the degree of misclassification of the data is penalized by the parameter C.

It often happens that the positive and negative samples are not linearly separable in the feature space. To cope with this problem, the so-called *kernel trick* is introduced to implicitly map the feature vectors into a higher-dimensional space, where hopefully the separation is easier. This trick makes the SVM very powerful to perform non-linear classifications, and therefore bring its popularization in various classification applications. Some widely used kernels include, the Gaussian kernel (also known as the Radial Basis Function (RBF) kernel), the polynomial kernel, the sigmoid kernel, *etc.*

In the image forensics literature where the feature is vector-based, the soft-margin SVM with the following Gaussian kernel is often used:

$$k(\boldsymbol{\phi}, \boldsymbol{\psi}) = \exp(-\gamma ||\boldsymbol{\phi} - \boldsymbol{\psi}||_2^2), \quad \gamma > 0, \tag{2.2}$$

where $\boldsymbol{\phi}$ and $\boldsymbol{\psi}$ are two feature vectors, and γ is the kernel parameter. To follow the common practice in image forensics literature, this thesis keeps the above choice of using soft-margin SVM with Gaussian kernel so as to build the SVM-based forensic detector, when the forensic feature is vector-based.

The penalization parameter C and the kernel parameter γ (interested readers can refer to [CL11] for more details) need to be determined before the SVM classifier is trained. The complexity and accuracy of the SVM classifier are balanced by the parameter setting of (C, γ). Pevný *et al.* [PBF10] have discussed the impact of different values of C and γ to the SVM

[7] In the following, we omit the word "two-class" for describing the SVM, for the sake of brevity. Here, we do not consider multi-class SVM. The SVMs we use in this thesis are all two-class.

classifier in detail. High value of C can improve the accuracy of the classifier on the training dataset but also increases the complexity. On the contrary, low value of C decreases the accuracy of the classifier but can normally achieve a good generalization with low complexity. Similarly, high value of γ may cause the over-fitting problem of the classifier, while low value of γ has the opposite impact.

The parameter setting of (C, γ) should well balance both the accuracy and complexity of the SVM classifier. In order to choose proper values for them, the standard way is to carry out a grid search with cross-validation. In this thesis, we follow the suggestion in [PBF10] and use the five-fold cross-validation with the following multiplicative grid:

$$\begin{array}{rcl} C & \in & \{0.001, 0.01, \cdots, 10000\} \\ \gamma & \in & \left\{2^i \mid i \in \{-\log_2 D - 3, \cdots, -\log_2 D + 3\}\right\}, \end{array} \tag{2.3}$$

where D is the dimensionality of the feature vector. In this thesis, we use the *LIBSVM* library [CL11] to train and test the SVM-based forensic detectors.

In this thesis, we consider some forensic feature which can be as high as 686-dimensional (*i.e.*, the K_{SPAM}^{S686} in Table 3.3). However, for the forensic testing of both JPEG compression and median filtering, we "only" use 500 samples (the datasets used for forensic testing are to be described in Sections 2.3.1 and 2.3.2) for each of the positive and negative classes for training the SVM classifiers. Fortunately, the SVM is by concept a classifier designed to be less sensitive to the problem of curse of dimensionality. As pointed out by Vapnik [Vap98], "One can consider the SV technique as a new type of parameterization of multidimensional functions that in many cases allows us to overcome the curse of dimensionality." (Note: here the "SV" stands for the "Support Vector".) In fact, the SVM appears to be especially advantageous when there are only a few training samples available and the feature is high-dimensional, a scenario rather quite frequently encountered in practical applications. This is mainly because of the fact that the classification strategy of the SVM is based on maximizing the margin in an implicit higher-dimensional space. This "geometrical" nature of the SVM does not require an estimation of the statistical distributions of classes, leading to a good generalization even when there are only limited number of training samples [MB04].

Steganography is the art of embedding secret message into an image, a video, or other forms of data. It is another domain in multimedia security field, which is different from anti-forensics but shares some common goals with anti-forensics [BK13]. In one of the most recent steganography work [HF13], the undetectability of the stego-images becomes poor when the payload reaches 0.5 bpp (bits per pixel).

It is a popular way to create image forgeries by compositing two images. An example involving JPEG compression is illustrated in Figure 1.3. Inspired by this common practice, we use a similar experimental setup to steganalytic testing, in order to conduct forensic testing using SVM-based detectors. As illustrated in Figure 2.2, for a given genuine, unprocessed image, the center part of the image is replaced by its corresponding processed image (JPEG compressed/median filtered, or processed again using anti-forensic methods) with a replacement rate around the values in $\{0.05, 0.1, 0.15, 0.2, 0.3, 0.4, 0.5, 0.6, 0.8, 1\}$ to create the image

forgery. For each replacement rate, the resulting composite image forgeries and the original images are then mixed together for training and testing the SVM-based detectors. Here, the image replacement rate in forensic testing can be taken as a counterpart of the bpp in image steganography.

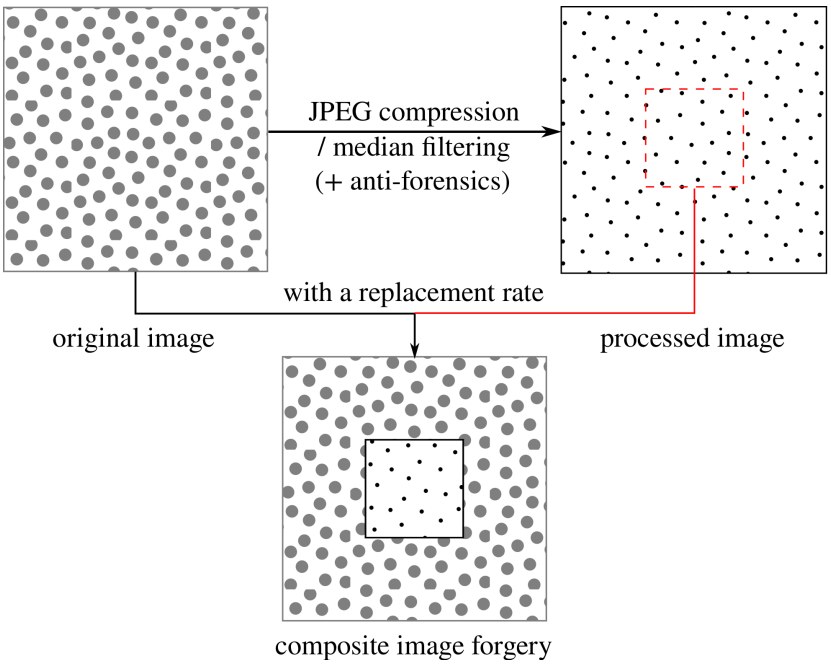


Figure 2.2: Illustration of creating a composite image forgery for training/testing the SVM-based detector.

Given an SVM-based detector and a set of images for forensic testing, a decision value is output by the SVM for each image. Similar to the scalar-based detectors, different classification strategies are implemented by varying the thresholds for the decision values. Thereafter, a ROC curve is plotted and an AUC value is calculated.

The SVM-based detectors are built based on the assumption that the forensic investigator has the knowledge of the anti-forensic method and is able to create a large amount of image forgeries for training the detector. This is standing on the beneficial side for forensics and is the *worst-case* scenario for anti-forensics. In practice, we also find it very difficult to disguise a whole processed (JPEG compressed or median filtered in this thesis) image as never processed, when examined under an SVM-based detector under this worst-case scenario. Currently, there are no JPEG or median filtering image anti-forensic algorithms which are able to fool powerful SVM-based detectors. This can be reflected by high AUC values when the replacement rate is close to 1 (Please see Sections 5.3.2, 7.3.4 and 7.4 for detailed experimental results). This observation in practice backs the statement of [BK13]: it is a challenging task to develop anti-forensic methods capable of resisting machine learning based detectors.

Standing on the anti-forensic side, this thesis mainly considers the results obtained with a relatively low replacement rate. We remain reserved on the undetectability of a whole image generated by anti-forensics. However, it can still find many applications in various anti-

forensic scenarios, *e.g.*, image splicing and tampering with relatively low replacement rate. For example, an image forgery can be created by replacing the head of one person in the picture without being detected.

2.2.2 Image Quality

The image quality assessment methods can be grouped into two categories: the *no-reference* methods and the *full-reference* methods, depending on whether the original "clean" image is known [WB06]. In real-world scenarios, the original image is usually unaccessible. However, we often hold the ground-truth in the scientific research. In this case, algorithm performance evaluation is possible in the full-reference way. In this thesis, we only consider 8-bit grayscale images. Here, we adopt two widely used full-reference image quality evaluation metrics: PSNR and SSIM, which will be respectively described in detail in the following.

2.2.2.1 PSNR

PSNR is probably the most widely used image quality assessment criterion in literature. It can be defined via the well-known MSE (Mean Squared Error). Given a reference grayscale image $\mathbf{X}$ with size $H \times W$, and its approximation or degraded version $\mathbf{Y}$, the PSNR in dB (deciBel) is calculated by:

$$\mathrm{PSNR}(\mathbf{X}, \mathbf{Y}) = 10 \times \log_{10} \left(\frac{\mathrm{MAX}^2}{\mathrm{MSE}} \right), \tag{2.4}$$

where the MSE is defined as:

$$\mathrm{MSE}(\mathbf{X}, \mathbf{Y}) = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} (\mathbf{X}_{i,j} - \mathbf{Y}_{i,j})^2 . \tag{2.5}$$

Note that MAX is the maximum possible image pixel value. For 8-bit grayscale images, MAX = 255. The PSNR is a symmetric measure, *i.e.*, $\mathrm{PSNR}(\mathbf{X}, \mathbf{Y}) = \mathrm{PSNR}(\mathbf{Y}, \mathbf{X})$. The greater the PSNR value is, the closer to the reference image $\mathbf{X}$ the approximated or degraded image $\mathbf{Y}$ is. In other words, the image quality of $\mathbf{Y}$ is higher.

2.2.2.2 SSIM

As an improvement of the traditional image quality evaluation metrics such as PSNR, Wang *et al.* [Wan+04] proposed the SSIM metric which correlates better with subjective scores of image visual quality. Compared with PSNR, the properties of human visual system (HVS) are taken into consideration for SSIM. Given the reference grayscale image $\mathbf{X}$ and its approximation or degraded version $\mathbf{Y}$ to be evaluated, the SSIM is defined as:

$$\mathrm{SSIM}(\mathbf{X}, \mathbf{Y}) = (l(\mathbf{X}, \mathbf{Y}))^a \times (c(\mathbf{X}, \mathbf{Y}))^b \times (s(\mathbf{X}, \mathbf{Y}))^c , \tag{2.6}$$

where $l(\mathbf{X}, \mathbf{Y})$ compares the luminance, $c(\mathbf{X}, \mathbf{Y})$ compares the contrast, and $s(\mathbf{X}, \mathbf{Y})$ compares the structure (see [Wan+04] for more details). Parameters a, b and c are used to balance the above three comparison functions. The widely used setting is $a = b = c = 1$. The SSIM is also a symmetric measure, *i.e.*, $\text{SSIM}(\mathbf{X}, \mathbf{Y}) = \text{SSIM}(\mathbf{Y}, \mathbf{X})$. The greater the SSIM value is, the higher the image quality of $\mathbf{Y}$ is. When $\mathbf{X} = \mathbf{Y}$, the SSIM metric achieves its maximum value 1.

2.2.3 Histogram Recovery

In information theory, the KL divergence [KL51] is a widely used non-symmetric metric to evaluate the similarity between two discrete distributions. Given a discrete distribution P_T and its approximation P_F, the KL divergence of P_F from P_T is defined as:

$$D_{KL}(P_T, P_F) = \sum_i P_T(i) \times \log \frac{P_T(i)}{P_F(i)}, \qquad \begin{array}{l} P_T(i) > 0, P_F(i) > 0, \\ \sum_i P_T(i) = 1, \text{ and } \sum_i P_F(i) = 1. \end{array} \tag{2.7}$$

It measures the information lost during the approximation. The smaller the KL divergence value is, the closer the two distributions in comparison are.

The KL divergence will be used to evaluate the DCT histogram of (anti-forensic) JPEG images (in Sections 5.2.1, 5.3.1 and 6.3) and the pixel value difference histogram of (anti-forensic) median filtered images (in Section 7.3.4).

2.3 Natural Image Datasets

As mentioned in Section 2.2.1, the forensic testing requires certain image dataset with genuine images. In the following, we will respectively present public natural image datasets which are used in the JPEG forensic testing and the median filtering forensic testing in the large-scale experimental tests of this thesis.

2.3.1 JPEG Forensic Testing

In the literature of JPEG anti-forensics [Sta+10a, Sta+10b, VTT11, SS11], the UCID (v2) corpus [SS04] is used for forensic testing. In this dataset, there are 1338 24-bit RGB (Red, Green, Blue) images of size 512×384 and of TIFF (Tagged Image File Format) format. During the JPEG compression, the color representation of the image is firstly converted from RGB to $Y'C_BC_R$. In each of the Y' (luma), C_B (blue-difference chroma component), and C_R (red-difference chroma component) components, the data is individually JPEG compressed. Without loss of generality, only the luma component of the image is considered in the JPEG forensic testing of this thesis. We use the Matlab function `rgb2ycbcr` to extract the luma data from each RGB UCID image, which is thereafter saved as an 8-bit grayscale image in PGM (Portable GrayMap) format.

In Chapter 6, the proposed JPEG anti-forensic method requires an image prior model and the covariance matrices of the spatial-domain JPEG compression noise, which need to be learned on a dataset. To this end, the last 338 images of the UCID corpus are put into a dataset called *UCIDLearn*. The first 1000 UCID images are then put into a dataset called *UCIDTest* for forensic testing. Since there is a training procedure for establishing the SVM-based forensic detectors, we randomly choose 500 images from UCIDTest dataset and put them into the *UCIDTR* dataset for this purpose. The rest 500 images in the UCIDTest dataset then constitute the *UCIDTE* dataset for forensic testing using the SVM-based detectors. Note that for the forensic testing using scalar-based detectors, all the 1000 images in UCIDTest dataset are used. For JPEG forensic testing, each image of UCIDTest dataset is JPEG compressed using a quality factor randomly selected from $\{50, 51, \cdots, 95\}$.

In Sections 4.4.1, 5.2.1, and 5.2.2.4, in order to tune a proper parameter setting or quickly analyze the proposed JPEG anti-forensic method, we use another dataset called *UCIDTest92*. It contains 92 images randomly chosen from the UCIDTest. Each UCIDTest92 image is JPEG compressed with a quality factor selected from $\{50, 51, \cdots, 95\}$, and every two images have the same factor. Besides, in Section 5.4.3, we conduct the double JPEG compression forensic testing on a relatively small dataset called *UCIDTest100*, which includes 100 randomly chosen images from the UCIDTest dataset.

In summary, the following image datasets are used for conducting the test of the JPEG anti-forensic methods:

- UCIDLearn, the last 338 images of UCID corpus, for learning an image prior model and the covariance matrices of the spatial-domain JPEG compression noise.
- UCIDTest, the first 1000 images of UCID corpus, for forensic testing using the scalar-based detectors, and also for the evaluations the image quality and the DCT histogram recovery.
  - UCIDTR, 500 images randomly selected from UCIDTest, for testing the SVM-based detectors.
  - UCIDTE, the rest 500 images in UCIDTest which are not chosen by UCIDTR, for training the SVM-based detectors.
  - UCIDTest92, 92 randomly selected images from UCIDTest, for tuning the parameters or for quickly analyzing the performance of the proposed JPEG anti-forensic method.
  - UCIDTest100, 100 randomly selected images from UCIDTest, to be used in a scenario of double JPEG compression anti-forensics.

Besides the above image datasets constituted from the UCID corpus [SS04], we also consider another relatively big image dataset – BOSSBase (v1.01) [BFP11]. This is for a deep analysis of the quantization table estimation based JPEG forensic detector [FD03] (to be described in Section 4.2.1). BOSSBase contains 10,000 raw images. For forensic testing, each of the raw image is converted to an 8-bit grayscale image of PGM format, using the *UFRaw*

utility[8]. The resulting image is thereafter down-scaled and then cropped to generate the final testing images of size 512×512. The 10,000 images are divided into 10 archives, according to the original numbering of the BOSSBase raw image file. The i-th ($i = 1, 2, \cdots, 10$) archive contains 1000 images from the number $1000 \cdot (i-1) + 1$ to $1000 \cdot i$.

The BOSSBase images are relatively smooth compared to other images (*e.g.*, UCID images [SS04] or the images described in Section 2.3.2 for median filtering forensic testing) which are not originally stored in the raw format. This is probably due to the fact that the sharpening feature is still missing in the UFRaw utility (see its webpage for details). The BOSSBase images may not be "natural" enough, leading to possible bias in forensic testing. Compared with cropping sub-images directly from the high-resolution PGM image, the down-sampling operation may produce final testing images containing more texture, hoping to some extent to mitigate the smoothness in the original image. In our JPEG forensic testing, we do not consider any interference from resampling. Moreover, one of the weaknesses of the quantization table estimation based JPEG forensic detector [FD03] is exposed under relatively smooth images. Therefore, in this thesis we use the BOSSBase dataset only for JPEG forensic testing using the quantization table estimation based detector [FD03].

2.3.2 Median Filtering Forensic Testing

As to the median filtering forensic testing, we do not use the UCID (v2) corpus [SS04], though it is the testing dataset for the median filtering anti-forensic work presented in [FB12, WSL13, DN+13]. This is based on the confirmation that UCID images have been processed by down-sampling, according to a personal communication with the authors of [SS04]. In addition, we do not want the resampling artifacts to interfere with the forensic testing of median filtering. In Section 7.4.1, we will consider the application for median filtering in disguising image resampling artifacts, where never resampled images are required for forensic testing. We also do not consider raw image datasets such as BOSSBase [BFP11], given the reasons described in the end of Section 2.3.1.

Based on the above considerations, we choose other image datasets which are created from 545 never resampled, non-compressed TIFF images capturing various indoor and outdoor scenes. Among them, 200 images[9] were taken by a Nikon D60, and have been used for resampling forensics [VPPG11, VPPG12]; whereas the rest 345 images[10] were taken by a Nikon D90, a Canon EOS 450D, and a Canon EOS 5D, and have been used for double JPEG compression forensics [BP12a, BP12b]. We crop 3 adjacent non-overlapping sub-images of size 512×512 from the center of each original high-resolution TIFF images, and convert them to grayscale images using the Matlab function `rgb2gray`. For some sub-images, one or more median filtering forensic detector outputs of Eqs. (3.12)-(3.15) (to be presented in Section 3.2.2) cannot yield valid values. For instance, the 0-valued denominator of Eq. (3.15)

[8] The UFRaw is probably the best utility for reading and manipulating raw images from digital cameras. It can be downloaded from: `http://ufraw.sourceforge.net`.

[9] Available at: `ftp://firewall.teleco.uvigo.es:27244/DS_01_UTFI.zip`.

[10] Available at: `ftp://lesc.dinfo.unifi.it/pub/Public/JPEGloc/dataset/`.

leads to the infinity. We observe that this normally happens for very smooth images, *e.g.*, the whole image only contains the scene of the sky. Besides the invalid detector outputs, it is not very likely for this kind of images to appear in real anti-forensic scenarios, which are therefore excluded from our forensic testing.

At last, we obtain 1607 8-bit grayscale images, which are randomly divided into 3 datasets: MFTE (Median Filtering TEsting), MFTR (Median Filtering TRaining), and MFPE (Median Filtering Parameter Estimation). MFTE contains 1000 images for forensic testing. MFTR comprises 500 images for training the SVM-based median filtering detectors. The remaining 107 images are put into MFPE for parameter estimation, which will be used for creating Wu *et al.*'s [WSL13] and our anti-forensic median filtered images (to be described in Section 7.3). Note that, we also make sure that all the sub-images (at most 3) cropped from a single high-resolution TIFF image are in either MFTE, or MFTR, or MFPE. Moreover, 100 images are randomly picked from MFTE to form a smaller dataset MFTE100, which will be used for the parameter grid search to be described in Sections 7.3.2, 7.3.3 and 7.3.4.

In summary, we use the following four image datasets for the forensic testing of median filtering:

- MFTR, 500 images for training the SVM-based forensic detectors.
- MFTE, 1000 images for forensic testing using the scalar-based or SVM-based forensic detectors, and also for the evaluations of the image quality and the pixel value difference histogram recovery.
- MFPE, 107 images for parameter estimation.
- MFTE100, 100 images randomly selected from MFTE dataset, for the grid search of parameters of the proposed median filtering anti-forensic method.

2.4 Relevant Optimization Algorithms

In this thesis, the proposed JPEG or median filtering anti-forensic methods often involve numerical optimization so as to estimate the "best" processed image in some sense. In this section, we briefly review four optimization algorithms that will be used in the subsequent chapters: the subgradient method, the Hungarian algorithm, and the "Half Quadratic Splitting" as well as the split Bregman method.

2.4.1 Subgradient Method

In order to minimize the cost function $f(\mathbf{u}) : \mathbb{R}^n \to \mathbb{R}$ of a convex optimization problem, the subgradient method iteratively solves the problem by [BXM03]:

$$\mathbf{u}^{(k+1)} = \mathbf{u}^{(k)} - t_k g^{(k)}. \tag{2.8}$$

Note that $\mathbf{u}^{(k)}$ is the k-th iterate, $g^{(k)}$ is one subgradient of f at $\mathbf{u}^{(k)}$, and $\alpha_k > 0$ is the step size at the k-th iteration. It is possible that f has more than one subgradient. When f is differentiable, the only subgradient of f is its gradient. The subgradient method is not a descent method, it is therefore common to keep the smallest objective function value found so far, that is:

$$f_{best}^{k} = \min\{f_{best}^{k-1}, f(\mathbf{u}^{(k)})\}. \tag{2.9}$$

There are many different types of step-size rules used by the subgradient method, such as the constant step size, the constant step length, the square summable but not summable step size, *etc.* In this thesis, we use the nonsummable diminishing step size [ADF05], *i.e.*,

$$t_k \geq 0, \quad \lim_{k\to\infty} t_k = 0, \quad \sum_{k=1}^{\infty} t_k = \infty. \tag{2.10}$$

For this step size rule, the algorithm is guaranteed to converge to the optimal value [BXM03].

One important extension of the subgradient method is the *projected subgradient method*, which is to solve the following constrained convex optimization problem:

$$\begin{array}{ll} \text{minimize} & f(\mathbf{u}) \\ \text{subject to} & \mathbf{u} \in \mathcal{C}, \end{array} \tag{2.11}$$

where $\mathcal{C}$ is a convex set. The projected subgradient method iteratively solves the problem by:

$$\mathbf{u}^{(k+1)} = P\left(\mathbf{u}^{(k)} - t_k g^{(k)}\right). \tag{2.12}$$

Here $P(\cdot)$ projects the input onto the constrained convex set $\mathcal{C}$.

We will use the subgradient method in Sections 4.3.1, 4.3.2, and 6.4.

2.4.2 Hungarian Algorithm

The Hungarian algorithm [Kuh55] is a well-known optimization method for solving the *assignment problem* in polynomial time. The assignment problem is defined as follows. Given two sets, $\mathcal{X}$ and $\mathcal{Y}$ of equal cardinality n, and the nonnegative weight function $W : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$, the goal is to find a bijection $f : \mathcal{X} \to \mathcal{Y}$ such that the cost function:

$$\sum_{x\in\mathcal{X}} W(x, f(x)), \tag{2.13}$$

is minimized.

The above assignment problem can also be interpreted by a square matrix of size $n \times n$. The (i, j)-th entry of the matrix is the cost (defined by the weight function W) of assigning the i-th member of $\mathcal{X}$ to the j-th member of $\mathcal{Y}$. The Hungarian algorithm finds an optimal assignment by applying the following steps in Algorithm 2.1 to the cost matrix.

Algorithm 2.1 The Hungarian algorithm (in brief description).

Require: Cost matrix $\mathbf{C}$ of size $n \times n$, minimum number of zero-covering lines $m = 0$

```
For each row, subtract all the entries of C by the row minimum.
For each column, the column minimum is found and subtracted from all the column entries of C.
while m ≠ n do
    Cover all the zero entries of the C, by drawing the minimum number of lines across rows and columns.
    if The minimum number of zero-covering lines is n then
        The optimal assignment is found.
        return.
    else
        Find the minimum among the non-covered entries of the C.
        Subtract this entry from each uncovered row of C, and then add it to each covered column of C.
        continue.
    end if
end while
```

The time complexity of the Hungarian algorithm is $O(n^3)$. When n is small, the problem can however be solved in a reasonable time.

Compared with other optimization methods (described in Sections 2.4.1, 2.4.3, and 2.4.4) which output an image in our image anti-forensic applications, the Hungarian algorithm is the only discrete optimization method used in this thesis. We will use the Hungarian algorithm in Sections 5.2.2.3, for the solving the perceptual DCT coefficient mapping problem. For this problem, the two finite DCT coefficient sets $\mathcal{X}$ and $\mathcal{Y}$ are known.

2.4.3 Half Quadratic Splitting

In image restoration, a popular method is to minimize an objective function with regularization. Linear regularization function is easy to optimize but incapable of recovering certain image details such as discontinuities. In contrast, non-linear (even non-convex) regularization is able to achieve this goal but is hard to optimize. In order to solve this kind of optimization problem, Geman and Yang [GY95] proposed the "Half Quadratic Splitting" method. It is thereafter widely adopted by various image restoration methods [KF09, ZW11], especially when the image prior term (can be taken as regularization) of the cost function is non-convex.

Let $\mathbf{y}$ denote the image corrupted from its original version $\mathbf{x}$. In order to estimate the original image from $\mathbf{y}$, the typical optimization problem takes the following form:

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{u}} \Theta(\mathbf{u};\mathbf{y}) = \arg\min_{\mathbf{u}} \lambda\Psi(\mathbf{u};\mathbf{y}) + \Phi(\mathbf{u}), \tag{2.14}$$

where λ is a parameter balancing different terms in the cost function. A common choice for

$\Psi(\mathbf{u};\mathbf{y})$ is $||\mathbf{A}\mathbf{u}-\mathbf{y}||_2^2$, where $\mathbf{A}$ depends on the image corruption model. The regularization term $\Phi(\mathbf{u})$ can be non-convex, which leads to direct optimization difficulties.

The basic idea of "Half Quadratic Splitting" is to introduce an auxiliary variable $\mathbf{w}$. Then the resulting new cost function takes the form:

$$\Theta'(\mathbf{u},\mathbf{w};\mathbf{y}) = \lambda\Psi(\mathbf{u};\mathbf{y}) + \Phi'(\mathbf{u},\mathbf{w}), \tag{2.15}$$

which has the same global minimum in $\mathbf{u}$ as Θ in Eq. (2.14). It is required that the choice of $\Phi'(\mathbf{u},\mathbf{w})$ should satisfy:

$$\Phi(\mathbf{u}) = \min_{\mathbf{w}} \Phi'(\mathbf{u},\mathbf{w}), \tag{2.16}$$

for every $\mathbf{u}$. This method is "half quadratic" because it is requested that $\Phi'(\mathbf{u},\mathbf{w})$ is quadratic in $\mathbf{u}$ when fixing $\mathbf{w}$. Thereafter, the cost function can be minimized by alternatively solving the following two sub-problems:

- $\mathbf{w}$ sub-problem, solving $\mathbf{w}$ given $\mathbf{u}$;
- $\mathbf{u}$ sub-problem, solving $\mathbf{u}$ given $\mathbf{w}$.

We will use the "Half Quadratic Splitting" method in Sections 6.4 and 7.3.1.

2.4.4 Split Bregman Method

The split Bregman method is an effective method which can solve a broad range of ℓ_1-regularized optimization problems [GO09]. More specifically, we aim to solve the following general ℓ_1-regularized problem:

$$\min_{\mathbf{u}} H(\mathbf{u}) + ||\Phi(\mathbf{u})||_1, \tag{2.17}$$

where $H(\mathbf{u})$ and $||\Phi(\mathbf{u})||_1$ are convex, and $\Phi(\mathbf{u})$ is differentiable. Usually, $H(\mathbf{u})$ is a ℓ_2 term, *e.g.*, $H(\mathbf{u}) = ||\mathbf{A}\mathbf{u}-\mathbf{y}||_2^2$. The "coupling" between the ℓ_1 and ℓ_2 terms makes the optimization problem in Eq. (2.17) hard to solve. In order to "de-couple" these two ℓ_1 and ℓ_2 components of this cost function, an auxiliary variable is introduced:

$$\mathbf{w} = \Phi(\mathbf{u}). \tag{2.18}$$

We therefore hope to solve the following constrained problem:

$$\min_{\mathbf{u},\mathbf{w}} H(\mathbf{u}) + ||\mathbf{w}||_1, \quad \text{such that } \mathbf{w} = \Phi(\mathbf{u}), \tag{2.19}$$

which can be further converted to an unconstrained problem:

$$\min_{\mathbf{u},\mathbf{w}} H(\mathbf{u}) + ||\mathbf{w}||_1 + \frac{\lambda}{2}||\mathbf{w}-\Phi(\mathbf{u})||_2^2, \tag{2.20}$$

by adding an ℓ_2 penalty term whose strength is control by the parameter λ. This introduction of the auxiliary variable $\mathbf{w}$ can also be taken as a "Half Quadratic Splitting" described in Section 2.4.3. It can be solved by alternating between the $\mathbf{w}$ sub-problem and the $\mathbf{u}$ sub-problem, with increasing λ. When $\lambda \to \infty$, we restrict $\Phi(\mathbf{u})$ to be equal to the auxiliary variable $\mathbf{w}$ and the solutions of Eq. (2.17) and Eq. (2.20) converge [ZW11]. This is the so-called *continuation* method, where $\Phi(\mathbf{u})$ in fact is not necessarily convex.

By contrary to the continuation method, the split Bregman method recursively solves the problem in Eq. (2.20) by:

$$\begin{cases} \left(\mathbf{u}^{(k+1)}, \mathbf{w}^{(k+1)}\right) = \arg\min\limits_{\mathbf{u},\mathbf{w}} H(\mathbf{u}) + ||\mathbf{w}||_1 + \frac{\lambda}{2}||\mathbf{w} - \Phi(\mathbf{u}) - \mathbf{b}^{(k)}||_2^2 \\ \mathbf{b}^{(k+1)} = \mathbf{b}^{(k)} + \left(\Phi(\mathbf{u}^{(k+1)}) - \mathbf{w}^{(k+1)}\right), \end{cases} \tag{2.21}$$

where $\mathbf{b}$ is the Bregman variable. The above minimization problem can also be solved by alternating between the $\mathbf{w}$ sub-problem and the $\mathbf{u}$ sub-problem, similar to Section 2.4.3.

The split Bregman method has several advantages over the traditional continuation methods [GO09]. The Bregman iteration can converge quickly, especially when there is an ℓ_1 term and a properly chosen λ. Moreover, the parameter λ remains constant, which avoids the instabilities that happen when $\lambda \to \infty$ for the continuation method.

Indeed, the Bregman method requires the original cost function in Eq. (2.17) is ℓ_p-regularized and $p = 1$. Practically, it can also give good results for $p < 1$ [Cha09, KTF11]. We will use the split Bregman method to solve the image variational deconvolution problem proposed in Section 7.3.1, for quality enhancement and anti-forensic purposes of median filtered images.

Chapter 3

Prior Art of JPEG and Median Filtering (Anti-)Forensics

Contents

3.1 JPEG Forensics and Anti-Forensics . . . 32
3.1.1 Basics of JPEG Compression . . . 32
3.1.2 JPEG Artifacts . . . 33
3.1.2.1 DCT-Domain Quantization Artifacts . . . 33
3.1.2.2 Spatial-Domain Blocking Artifacts . . . 34
3.1.3 JPEG Image Quality Enhancement . . . 35
3.1.4 Detecting JPEG Compression . . . 36
3.1.5 Disguising JPEG Artifacts . . . 36
3.1.6 Attacking JPEG Anti-Forensics . . . 37
3.1.7 Other Relevant Methods . . . 38
3.1.8 Summary . . . 38
3.2 Median Filtering Forensics and Anti-Forensics . . . 40
3.2.1 Median Filtering Basics and Artifacts . . . 40
3.2.2 Detecting Median Filtering . . . 41
3.2.2.1 Kirchner and Fridrich's Method . . . 41
3.2.2.2 Cao *et al.*'s Method . . . 42
3.2.2.3 Yuan's Method . . . 42
3.2.2.4 Chen *et al.*'s Method . . . 43
3.2.2.5 Kang *et al.*'s Method . . . 43
3.2.2.6 Zhang *et al.*'s Method . . . 43
3.2.3 Disguising Median Filtering Artifacts . . . 44
3.2.3.1 Fontani and Barni's Method . . . 44
3.2.3.2 Wu *et al.*'s Method . . . 44
3.2.3.3 Dang-Nguyen *et al.*'s Method . . . 44
3.2.4 Summary . . . 45

In this chapter, we present the related work of JPEG and median filtering (anti-)forensics, respectively. Firstly, the basics and artifacts of JPEG compression and median filtering are described. Then, the state-of-the-art forensic and anti-forensic methods are introduced. Finally, we summarize the existing forensic detectors which we aim to attack in the proposed anti-forensic methods, and the existing anti-forensic methods which we will compare with the proposed anti-forensic methods in this thesis.

3.1 JPEG Forensics and Anti-Forensics

3.1.1 Basics of JPEG Compression

Detailed descriptions of JPEG compression can be found in [PM93]. Here we briefly review the process of JPEG compression and decompression for 8-bit grayscale images.

The original uncompressed image $\mathbf{X}$ of size $H \times W$ is firstly split into L non-overlapping 8×8 pixel value blocks. Here, without loss of generality, we assume that both H and W are integer multiples of 8. For each block, a 2-dimensional DCT is applied to obtain its corresponding DCT coefficient block. As DCT is an orthogonal linear transform, this mapping can be modeled as a matrix multiplication $\mathbf{DX}$. More specifically, the (r, c)-th $(r, c = 1, 2, \cdots, 8)$ DCT coefficient of the l-th $(l = 1, 2, \cdots, L)$ block for $\mathbf{X}$, namely, $(\mathbf{DX})^l_{r,c}$ can be calculated as follows:

$$(\mathbf{DX})^l_{r,c} = \sum_{i=1}^{8} \sum_{j=1}^{8} \mathbf{X}^l_{r,c} \cdot f^b_{r,c}(i, j), \tag{3.1}$$

where the (r, c)-th DCT *basis function* $f^b_{r,c}(i, j)$ is defined as:

$$f^b_{r,c}(i, j) = \frac{w(r)w(c)}{4} \cos\left(\frac{\pi}{16}(2i-1)(r-1)\right) \cos\left(\frac{\pi}{16}(2j-1)(c-1)\right), \quad \text{for} \quad i, j = 1, 2, \cdots, 8. \tag{3.2}$$

Note that, $w(k) = 1/\sqrt{2}$ if and only if $k = 1$, and $w(k) = 1$ otherwise. Fixing a (r, c) pair, we call the set of DCT coefficients $\{(\mathbf{DU})^l_{r,c} | l = 1, 2, \cdots, L\}$ the "*subband* (r, c)", for a generic image $\mathbf{U}$. The $(0, 0)$ subband is the DC (Direct Current) component, whose DCT coefficients are called DC coefficients. The remaining 63 subbands are in general the AC (Alternating Current) components, whose DCT coefficients are called AC coefficients.

Afterwards, the (r, c)-th DCT coefficient $(\mathbf{DX})^l_{r,c}$ of the l-th block is uniformly quantized by:

$$(\mathcal{Q}(\mathbf{DX}))^l_{r,c} \doteq \text{round}\left(\frac{(\mathbf{DX})^l_{r,c}}{\mathbf{Q}_{r,c}}\right), \tag{3.3}$$

where the 8×8 matrix $\mathbf{Q}$ with positive integers is the so-called *quantization table*, and $\mathbf{Q}_{r,c}$ is its (r, c)-th entry (also called "quantization step"). Moreover, round$(\cdot)$ is the rounding function. The resulting, quantized DCT coefficients $\mathcal{Q}(\mathbf{DX})$ are then losslessly encoded.

The quantization steps in a quantization table $\mathbf{Q}$ can vary according to a user's concrete setting. In this thesis, we consider the widely used JPEG quantization tables suggested by the Independent JPEG Group (IJG) [Ijg]. In their method, the quantization matrix is defined by an integer *quality factor* q $(q = 1, 2, \cdots, 100)$. According to the concrete value of the quality factor, the corresponding quantization matrix $\mathbf{Q}$ is obtained by properly scaling a template table using q. The smaller the value of q is, the greater the quantization steps in $\mathbf{Q}$ are. This leads to a lower bit-rate JPEG compression. The resulting JPEG compressed image will be of lower visual quality, but of smaller file size.

As to the decompression, the quantized DCT coefficient is extracted from the decoded bitstream, and then dequantized by multiplying it by the corresponding quantization step:

$$(\mathcal{Q}^{-1}(\mathcal{Q}(\mathbf{DX})))^l_{r,c} \doteq (\mathcal{Q}(\mathbf{DX}))^l_{r,c} \times \mathbf{Q}_{r,c}. \tag{3.4}$$

The dequantized DCT coefficients are then transformed to the spatial domain by the 2-dimensional inverse discrete cosine transform (IDCT), which can be expressed as multiplication by the 8×8 block IDCT matrix $\mathbf{D}^{-1}\mathcal{Q}^{-1}(\mathcal{Q}(\mathbf{DX}))$. At last, rounding and truncation operation (denoted as $\mathcal{RT}(\cdot)$) is applied to constrain the pixel values to be integers within $[0, 255]$, and the decoded JPEG image is obtained by:

$$\mathbf{Y} = \mathcal{RT}(\mathbf{D}^{-1}\mathcal{Q}^{-1}(\mathcal{Q}(\mathbf{DX}))). \tag{3.5}$$

In this thesis, at the standpoint of a forger, we make the reasonable assumption that the quantized DCT coefficients $\mathcal{Q}(\mathbf{DX})$ and the quantization table $\mathbf{Q}$ are available to us. For example, we can use Sallee's Matlab JPEG toolbox [Sal03] to read them from the JPEG file that we want to manipulate.

3.1.2 JPEG Artifacts

After the JPEG compression, two known kinds of artifacts appear, indicating the JPEG compression history of one image. They are the *quantization artifacts* in the *DCT domain* and the *blocking artifacts* in the *spatial domain*, respectively. Both of them are traces left from an image's JPEG compression history.

Figure 3.1 shows the two kinds JPEG artifacts present in the "Lena" image after it is JPEG compressed with quality factor 10. Figure 3.1-(a) is the close-up image of original, uncompressed "Lena" image, while -(b) is its corresponding JPEG compressed version. In Figure 3.1-(c) and -(d), the DCT histograms of subband $(2, 2)$ corresponding to -(a) and -(b) are shown, respectively. Note that in Figure 3.1-(d), the DCT coefficient error introduced by the rounding and truncation operation $\mathcal{RT}(\cdot)$ during the JPEG decompression process is not counted.

3.1.2.1 DCT-Domain Quantization Artifacts

From Eq. (3.4), we can see that the dequantized DCT coefficient $(\mathcal{Q}^{-1}(\mathcal{Q}(\mathbf{DX})))^l_{r,c}$ are obtained by multiplying the quantized DCT coefficient $(\mathcal{Q}(\mathbf{DX}))^l_{r,c}$ by its corresponding quantization step length $\mathbf{Q}_{r,c}$. Therefore, the DCT coefficients in the JPEG images are clustered around the integer multiples of the quantization step. From Figure 3.1-(d), compared with -(c), a comb-like distribution of DCT coefficients can be seen. This kind of artifacts is namely the quantization artifacts in the DCT domain of the JPEG image.

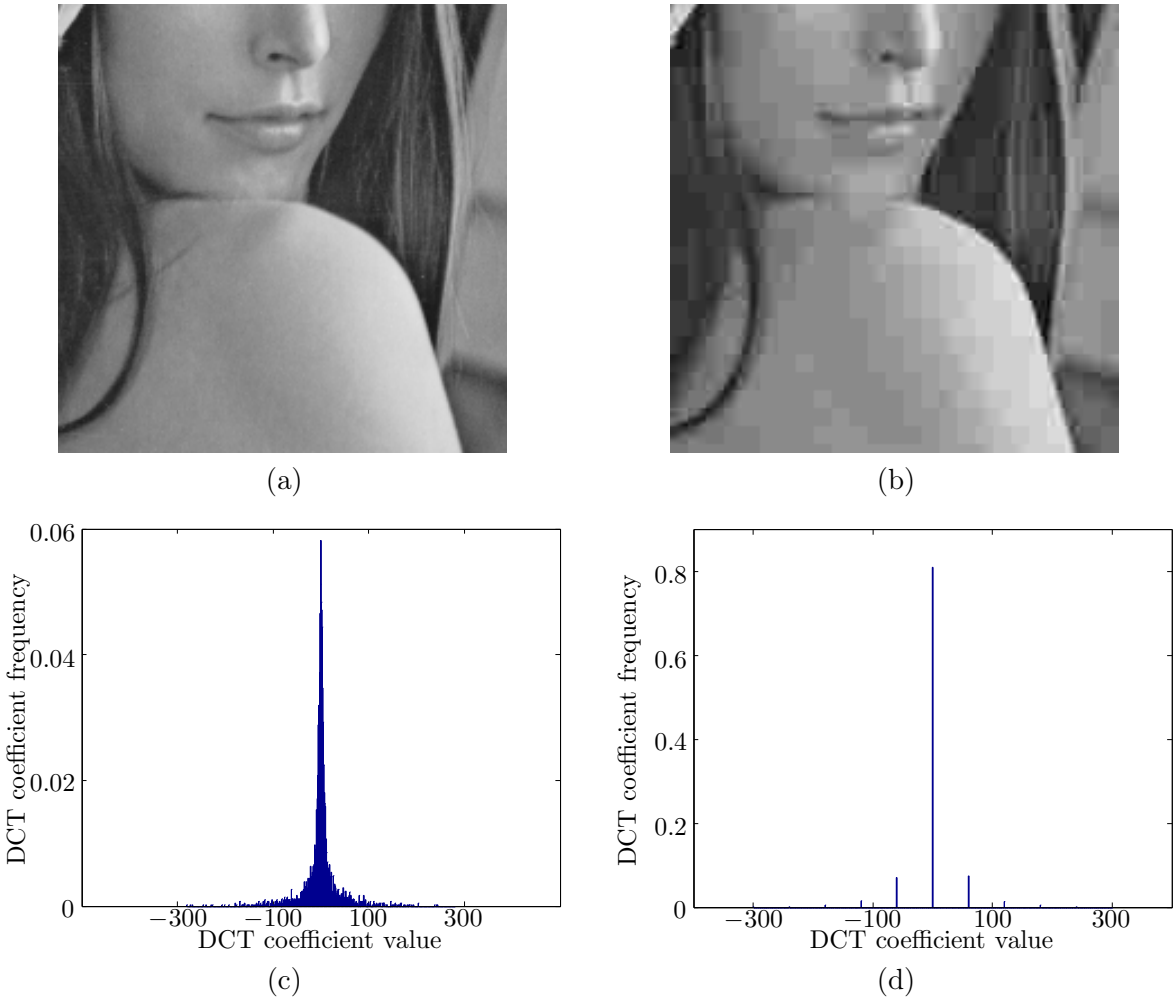


Figure 3.1: Examples of JPEG artifacts. Here the JPEG image is compressed from the "Lena" image with quality factor 10.

3.1.2.2 Spatial-Domain Blocking Artifacts

As shown in Figure 3.1-(b), in the spatial domain of JPEG images, there are blocking artifacts. This is mainly due to that JPEG compression is based on the block DCT transform. During the JPEG compression process, the image is first divided into non-overlapping 8×8 blocks. Each block is thereafter individually DCT transformed and quantized, leading to pixel value discontinuities across the block borders [Far09a], namely, the blocking artifacts.

Besides, Robertson and Stevenson [RS05] also point out that the form of blocking artifacts is related to the image itself. For the relatively smooth signal, there is more energy contained in the low-frequency subbands. The high-frequency subbands contribute less quantization noise, since they have relatively little energy. Because of the DCT basis functions (see Eq. (3.2) for the formula, and Figure 3.2 for the illustration), the considerable quantization noise in the low-frequency subbands leads to high error variance in the locations next to the block

boundaries. On the other hand, for the relatively complex signal, *e.g.*, the textured regions of the image, a higher error variance will appear in the center of the block than the borders. The above also explains the presence of blocking artifacts in the JPEG images.

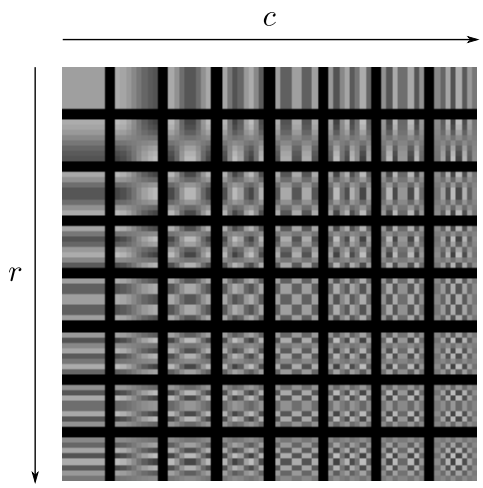


Figure 3.2: Illustration of DCT basis functions. Basis image values are normalized for a better visibility. Each 8×8 block corresponds to a (r, c) pair.

3.1.3 JPEG Image Quality Enhancement

As discussed in Sections 1.3.2-1.3.3, we propose to conduct JPEG anti-forensics by leveraging on advanced concepts/methods from the image restoration field. There are various JPEG post-processing methods in the literature of image restoration. They are designed to improve the visual quality of the JPEG compressed image especially at low bit-rates [YGK95, LY04, ADF05, RS05, SC07, Zha+08], without any special consideration of the forensic undetectability. In this section, we will briefly describe two JPEG image quality enhancement (or post-processing) methods which inspired us for designing the proposed JPEG anti-forensic methods in Chapters 4-6.

Alter *et al.* [ADF05] proposed a JPEG image quality enhancement method based on the adapted TV, by solving a constrained TV minimization problem. This method is able to partly remove the spatial-domain blocking artifacts from a JPEG image, for improving its visual quality. Yet, Figure 4.6-(a) indicates that it is not able to create anti-forensic JPEG images with a good forensic undetectability. The TV-based deblocking method to be proposed in Chapter 4 is to some extent inspired by this TV-based JPEG image post-processing method [ADF05], yet with a properly re-designed variation form and a new TV-based blocking measurement. More details about the proposed JPEG anti-forensic method using TV-based deblocking can be found in Section 4.3.1.

For JPEG image post-processing, Robertson and Stevenson [RS05] proposed to model the JPEG compression process as an addition of spatial-domain compression noise to the original image. The 0-mean multivariate Gaussian is used to model this compression noise, which is

treated as a random quantity. Moreover, the compression noise and the original image are assumed to be independent. With a proper image prior model, quality enhanced JPEG image can be obtained by using the MAP estimation. A more detailed description can be found in Section 6.2.1. Moreover, a novel JPEG image quality enhancement method is proposed in Section 6.2.2, inspired by the JPEG compression noise model [RS05] and a sophisticated image prior model [ZW11]. From Table 6.5, again we can see that JPEG image quality enhancement method does help to partly remove JPEG artifacts, yet not enough to reach a satisfactory forensic undetectability level. Anti-forensic terms/strategies are to be proposed to fulfil the task for further removing the remaining JPEG artifacts for anti-forensic considerations. A detailed description of the proposed JPEG anti-forensic method supported by experimental results can be found in Sections 6.3-6.4.

3.1.4 Detecting JPEG Compression

Fan and De Queiroz [FD03] proposed an algorithm for maximum-likelihood estimation (MLE) of the JPEG quantization table, from a spatial-domain bitmap representation of the image. The method can also serve as a detector to classify an image as not JPEG compressed, if each entry of the estimated quantization table is either 1 or "undetermined" [FD03, Sta+10a, SL11].

Focusing on 8×8 block boundaries, Fan and De Queiroz [FD03] also proposed a JPEG blocking signature measure as:

$$K_F = \sum_k |H_I(k) - H_{II}(k)|, \tag{3.6}$$

where H_I and H_{II} are normalized histograms of pixel value differences across block boundaries and within the block, respectively (see [FD03] for details).

Luo *et al.* [LHQ10] proposed a scalar feature to distinguish JPEG images from uncompressed ones, based on the AC coefficient distribution change in the range of $(-1, 1)$ and that in the union range of $(-2, -1]$ and $[1, 2)$.

As to the quantization step estimation, Luo *et al.* [LHQ10] further estimated the AC component quantization step as the integer, which is greater than 2 and gives the maximum value of the DCT histogram. Experimental results show that this JPEG image detector outperforms Fan and De Queiroz's method [FD03].

3.1.5 Disguising JPEG Artifacts

In the DCT domain, subband (r, c) contains the (r, c)-th DCT coefficient from each DCT coefficient block. For the AC component, *i.e.*, subband $(r, c) \neq (1, 1)$, the Laplacian distribution is a popular choice [LG00] for modeling the distribution of unquantized DCT coefficients. For the DC component, *i.e.*, subband $(1, 1)$, there is no general model representing its distribution.

Nevertheless, its DCT *quantization noise* can be assumed to follow a uniform distribution [Sta+10a, SL11].

In order to disguise the DCT-domain quantization artifacts of a JPEG image, Stamm *et al.* [Sta+10a, SL11] proposed to add a dithering signal to the dequantized DCT coefficients $\mathcal{Q}^{-1}(\mathcal{Q}(\mathbf{DX}))$, so that the dithered DCT coefficients approximate the estimated distribution of the unquantized ones. This JPEG anti-forensic method succeeds in fooling the quantization table estimation based detector proposed in [FD03].

After the dithering operation, Stamm *et al.* [Sta+10b, SL11] later proposed an anti-forensic deblocking operation against the blocking artifacts detector in [FD03], that is, the blocking signature measure K_F of Eq. (3.6). For a given image $\mathbf{U}$, the anti-forensically deblocked image $\tilde{\mathbf{U}}$ is obtained according to:

$$\tilde{\mathbf{U}}_{i,j} = \mathrm{med}_s(\mathbf{U}_{i,j}) + w_{i,j}, \tag{3.7}$$

where $\mathrm{med}_s(\cdot)$ is the median filtering operation with local window size $s \times s$, and $w_{i,j}$ is a low-power white Gaussian noise of variance σ^2.

Valenzise *et al.* proposed a perceptual anti-forensic dithering operation [VTT11], whose resulting anti-forensic JPEG image has a higher perceptual quality than the one processed by Stamm *et al.*'s dithering method [Sta+10a, SL11]. A "just-noticeable distortion" (JND) [WN09] criterion is adopted to control the amount of introduced distortion. A minimum-cost bipartite graph matching problem is used as the mathematical model for the adaptive insertion of the dithering signal. A greedy algorithm is implemented to get an approximate solution in order to reduce the computation cost.

3.1.6 Attacking JPEG Anti-Forensics

Valenzise *et al.* [VTT11, Val+11, VTT13] claimed that the dithering signal of [Sta+10a] degrades the image quality by introducing noise. Inspired by the JPEG ghosts detector [Far09b], they designed an efficient detector against JPEG anti-forensic dithering, which examines the noisiness of re-compressed versions of the image under test. The TV of the re-compressed image (the ℓ_1 norm of the spatial first-order derivatives) [ROF92] is employed as the image noisiness measure. For a given image $\mathbf{U}$, the detector re-compresses it using different quality factors $q = 1, 2, \ldots, 100$, as a function of which, $\mathrm{TV}(q)$ is computed as the total variation of the re-compressed image. The first order backward finite difference $\Delta\mathrm{TV}(q)$ is calculated as:

$$\Delta\mathrm{TV}(q) = \mathrm{TV}(q) - \mathrm{TV}(q-1), \tag{3.8}$$

with $\mathrm{TV}(0)$ prescribed to be 0. The forensic measure is:

$$K_V = \max_{q \in \{1,2,\cdots,100\}} \Delta\mathrm{TV}(q). \tag{3.9}$$

Lai and Böhme [LB11] proposed another calibration-based detector to counter Stamm

et al.'s JPEG anti-forensic method [Sta+10a, SL11], borrowing the idea of calibration from steganalysis [FGH02]. The detector compares the variances of the corresponding subbands from a given image $\mathbf{U}$ and its calibrated version $\mathbf{U}_{cal}$, which is obtained by cropping $\mathbf{U}$ by 4 pixels both horizontally and vertically. The calibrated feature K_L is then established as:

$$K_L = \frac{1}{28}\sum_{k=1}^{28}\left|\frac{\mathrm{var}(\mathbf{D}_k\mathbf{U}) - \mathrm{var}(\mathbf{D}_k\mathbf{U}_{cal})}{\mathrm{var}(\mathbf{D}_k\mathbf{U})}\right|, \tag{3.10}$$

where $\mathrm{var}(\cdot)$ returns the sample variance of the input vector, and $\mathbf{D}_k$ is a matrix extracting the DCT coefficients of the k-th high-frequency subband (as defined in [LB11]).

3.1.7 Other Relevant Methods

Moreover, some relative techniques can also be extended for JPEG anti-forensics, *e.g.*, the Shrink-and-Zoom (SAZ) attack proposed by Sutthiwan and Shi [SS11], though it was initially designed for double JPEG anti-forensics. Given a JPEG image, a shrinkage (image down-scaling) operation is firstly applied; then the processed image is zoomed back to the same size as the original one, to obtain the anti-forensic JPEG image.

Li *et al.* [LLH12] considered the process of creating anti-forensic JPEG image [Sta+10a, Sta+10b, SL11] as data hiding, in the view of JPEG steganalysis. The anti-forensic process changes the intra- and inter-block statistics of the image, which can be measured using a group of Markov process transition probability matrices [CS08]. A 100-dimensional feature vector is then extracted and fed to an SVM for building the JPEG forensic detector. Similarly, the well-known Subtractive Pixel Adjacency Matrix (SPAM) feature vector [PBF10] has also been used for countering JPEG anti-forensics [VTT13].

3.1.8 Summary

In this thesis, the JPEG forensic detectors described in Sections 3.1.4, 3.1.6, and 3.1.7 are used for forensic testing for different (anti-forensic) JPEG images. For the sake of conciseness, we hereafter use symbols for referring to the detectors, which are summarized in Table 3.1. Here, we mainly name the detectors directly after the feature value name, *e.g.*, K_F, K_V, and K_L in Eqs. (3.6), (3.9), and (3.10), respectively. The subscript of the detector name is based on the surname of the first author of the corresponding algorithm. We use the superscript 'S' together with the dimensionality of the feature vector to indicate that K_{Li}^{S100} and K_P^{S686} are SVM-based detectors. Meanwhile, the superscript 'Q' of K_F^Q and K_{Luo}^Q shows that they estimate quantization steps. Note that K_U^p (parameters $p = 1$, and $p = 2$ are considered here) is a new family of JPEG blocking detectors proposed in this thesis (see Eq. (4.3) in Section 4.2.2).

Practically, in some high-frequency subbands of highly JPEG compressed images, all the DCT coefficients are quantized to 0 and no quantization step could be determined. Hence,

Table 3.1: JPEG forensic detectors.

K_F^Q	Fan and De Queiroz's [FD03] quantization table estimation based detector;
K_F	Fan and De Queiroz's [FD03] JPEG blocking artifacts detector;
K_{Luo}	Luo *et al.*'s [LHQ10] JPEG identifying detector;
K_{Luo}^Q	Luo *et al.*'s [LHQ10] quantization step estimation based detector;
K_V	Valenzise *et al.*'s [Val+11, VTT13] TV-based JPEG anti-forensic detector;
K_L	Lai and Böhme's [LB11] calibration-based detector;
K_U^1, K_U^2	the proposed JPEG blocking artifacts detectors (see Eq. (4.3));
K_{Li}^{S100}	Li *et al.*'s [LLH12] 100-dimensional intra- and inter-block correlation feature [CS08] based detector;
K_P^{S686}	Pevný *et al.*'s [PBF10] 686-dimensional SPAM feature based detector.

for detector K_{Luo}^Q, we use the number of defined estimated quantization steps greater than 1 as the final feature value. Besides, the range of the re-compression quality factor q in Eq. (3.9) is shrunk to $\{45, 46, \cdots, 100\}$, as we use the JPEG images compressed with quality factors randomly selected from $\{50, 51, \cdots, 95\}$ in this thesis. Though detector K_F^Q is analyzed as not very reliable in Section 4.2.1, as it may result in relatively high false positive rate (detecting never compressed images as anti-forensic JPEG images), we still test it. Following the suggestion in [FD03, Sta+10a, SL11], a given image is classified as never compressed if and only if its estimated quantization table is full of entries 1 or "undetermined". We will report the rate of correctly detecting anti-forensic JPEG images for detector K_F^Q.

Table 3.2 briefly reviews the state-of-the-art anti-forensic JPEG images, which will be used for comparison with the anti-forensic JPEG images processed using the proposed methods in Chapters 4-6. In order to create this state-of-the-art anti-forensic JPEG image $\mathcal{F}_{S_q S_b}^J$ [Sta+10a, Sta+10b, SL11], we set $s = 3$ and $\sigma^2 = 2$ (see Section 3.1.5 and Eq. (3.7)), following the setting in the original papers. Also for the sake of brevity, we hereafter use the symbols listed in Table 3.2 for referring to the original, JPEG, or state-of-the-art anti-forensic JPEG images.

From Sections 3.1.5 and 3.1.7, we can see that the state-of-the-art JPEG anti-forensic methods use noise injection [Sta+10a, SL11, VTT11] or simple image processing operations (*e.g.*, median filtering [Sta+10b, SL11], and resampling [SS11]), in order to remove telltale artifacts from a JPEG image. These operations may be effective for removing the DCT-domain quantization artifacts or the spatial-domain blocking artifacts. However, they also largely degrade the visual quality of the processed image [VTT11, Val+11, VTT13], which can still detected by other advanced JPEG forensic detectors [Val+11, LB11, VTT13].

In Chapters 4-6, we follow another research line for JPEG anti-forensics, by adopting some concepts/methods from JPEG image restoration field. We hope to created anti-forensic JPEG images with a better forensic undetectability against the detectors listed in Table 3.1 as well

Table 3.2: Notations for original, JPEG, and state-of-the-art anti-forensic JPEG images.

$\mathcal{I}$	genuine, uncompressed image;
$\mathcal{J}$	JPEG image;
$\mathcal{F}^J_{S_q}$	Stamm *et al.*'s [Sta+10a, SL11] anti-forensic JPEG image created from $\mathcal{J}$ using the dithering operation (see Section 3.1.5);
$\mathcal{F}^J_{S_qS_b}$	Stamm *et al.*'s [Sta+10b, SL11] anti-forensic JPEG image created from $\mathcal{F}^J_{S_q}$ using the median filtering based method (see Section 3.1.5);
$\mathcal{F}^J_V$	Valenzise *et al.*'s [VTT11] anti-forensic JPEG image created from $\mathcal{J}$ using the perceptual anti-forensic dithering operation (see Section 3.1.5);
$\mathcal{F}^J_{Su}$	Sutthiwan and Shi's [SS11] anti-forensic JPEG image created from $\mathcal{J}$ using the SAZ attack (see Section 3.1.7).

as a higher visual quality of the anti-forensic JPEG image, compared with the state-of-the-art anti-forensic JPEG images listed in Table 3.2.

3.2 Median Filtering Forensics and Anti-Forensics

3.2.1 Median Filtering Basics and Artifacts

Let $\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_i, \cdots, \mathbf{x}_N)^T$ denote the given original grayscale image (in vectorized form) of N pixels, the i-th pixel of its median filtered version can be obtained by:

$$\mathbf{y}_i = \text{median}\ (\mathbf{P}^i\mathbf{x}), \tag{3.11}$$

where $\mathbf{P}^i$ is an $s^2 \times N$ sparse matrix extracting the i-th overlapping image block of size $s \times s$ (s is an odd number), and $\mathbf{x}_i$ locates in the geometrical center of block $\mathbf{P}^i\mathbf{x}$. Note that median$(\cdot)$ returns the median value of the input sample vector. Different windows can be used for image median filtering [PV92]. In this thesis, we only consider the square window, which probably is the most commonly used one. We use the notation $\mathcal{MF}^{(m)}(\cdot)$ for the image processing operation, which applies median filtering to the input for m (> 0) time(s).

Without loss of generality, in this thesis, we perform median filtering anti-forensics on images which have been median filtered once, with $s = 3$. It is also the experimental setting in all the state-of-the-art median filtering anti-forensic methods [FB12, WSL13, DN+13]. For the sake of brevity, we hereafter use the abbreviation "MF image" to refer to the "median filtered" image.

Within a certain neighborhood of the MF image, the probability of pixel values originate from the same pixel of the original image is high. This effect is called "*streaking*" by Bovik [Bov87]. The streaking artifacts can also be inferred from the shape change of the pixel

value difference histogram after median filtering. Comparing Figure 3.3-(b) with -(a), it can be seen that the peak around the bin 0 of the first-order horizontal pixel value difference histogram becomes higher after the image is median filtered (more discussion can be found in Section 7.2.2). It indicates that the pixels in a certain neighborhood tend to have close or even identical values in the MF image.

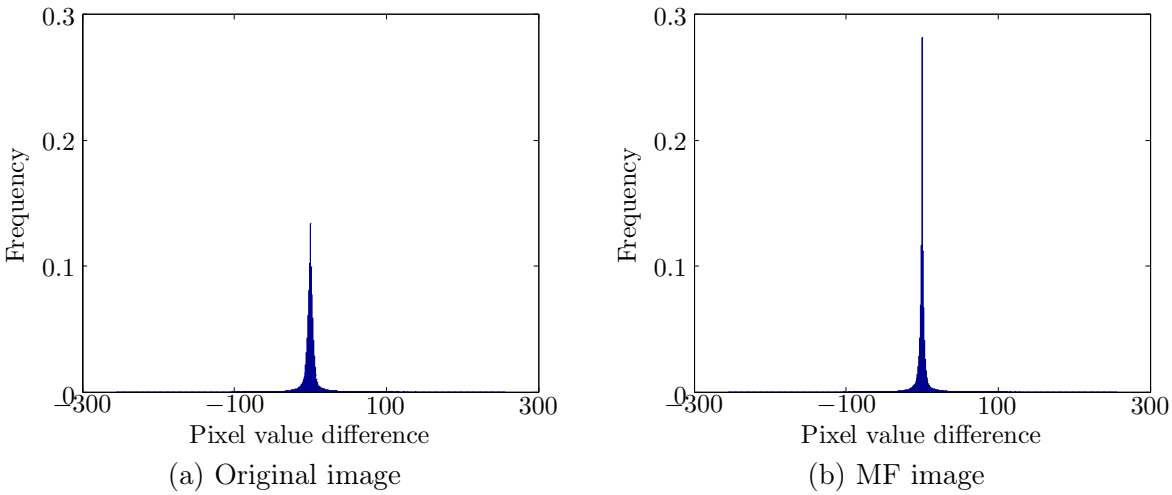


Figure 3.3: Example results of first-order horizontal pixel value difference histograms of the original image and its corresponding MF image, respectively.

3.2.2 Detecting Median Filtering

3.2.2.1 Kirchner and Fridrich's Method

Based on the "streaking artifacts" [Bov87], Kirchner and Fridrich [KF10] proposed a forensic feature for detecting MF images:

$$K_K = h_0/h_1, \tag{3.12}$$

where h_0 and h_1 are the occurrance frequencies of bins 0 and 1 of the first-order pixel value difference histogram, respectively. This feature K_K is expected to be around 1 for original images, whereas to be much greater than 1 for MF images. The underlying motive here is that the median filtering acts approximately as a low-pass filter, in a way that processed pixel values are appearing with a growing frequency to be constant in a certain neighborhood [Bov87]. Furthermore, in order to avoid strong saturation effects in original images, the given image is firstly divided into a set of $b \times b$ non-overlapping blocks. In the i-th block, the histogram ratio in Eq. (3.12), namely K_K^i, is calculated. A more powerful median filtering forensic feature is thereafter constructed as:

$$\hat{K}_K = \underset{i}{\text{median}}\ (w^i K_K^i), \tag{3.13}$$

where $w^i = 1 - \left(\frac{h_0^i}{b^2-b}\right)$ is the weight function. Note that h_0^i is the occurrence frequency of bin 0 for the first-order pixel value difference histogram in the i-th block. In [KF10], it is suggested to set $b = 64$, for a good tradeoff between the forensic detectability and computation cost of the feature.

By adopting the SVM, Kirchner and Fridrich [KF10] also showed the effectiveness of using the well-known SPAM feature vector [PBF10] to discriminate MF images from the original images.

3.2.2.2 Cao *et al.*'s Method

After an original image is median filtered, the probability of zero values in the first-order pixel value difference map increases, especially in textured regions. Based on this analysis, Cao *et al.* [Cao+10] proposed another median filtering forensic feature:

$$K_C = \frac{\sum_i (\Delta^r \mathbf{u})_i \cdot \mathbf{v}_i}{\sqrt{2}\sum_i \mathbf{v}_i} + \frac{\sum_i (\Delta^c \mathbf{u})_i \cdot \mathbf{v}_i}{\sqrt{2}\sum_i \mathbf{v}_i}, \tag{3.14}$$

where $(\Delta^r \mathbf{u})_i$ (or $(\Delta^c \mathbf{u})_i$) is 1 if the i-th element in the first-order row (or column) pixel value difference map is zero, or is 0 otherwise; and $\mathbf{v}$ is the binary local variance map, indicating textured regions ($\mathbf{v}_i = 1$) or smooth regions ($\mathbf{v}_i = 0$) of the image (see [Cao+10] for more details).

3.2.2.3 Yuan's Method

Median filtering is based on order-statistics. Therefore, it affects the order and the pixel value level inside the image block $\mathbf{P}^i\mathbf{u}$. Besides, the median filter works on overlapping blocks, which leads to the appearance of certain dependency artifacts. In order to capture these local dependency artifacts, Yuan [Yua11] proposed 5 feature vectors: $\mathbf{f}^{DBM}$, $\mathbf{f}^{OBC}$, $\mathbf{f}^{QGL}$, $\mathbf{f}^{DBC}$, and $\mathbf{f}^{FBC}$ (see [Yua11] for details), which are together recognized as the MFF (Median Filtering Forensics) feature vector $\mathbf{f}^{MFF}$. The median filtering forensic detector is thereafter built with the adoption of the SVM. Moreover, the above 5 feature vectors can be further merged as a single scalar feature:

$$K_Y = \frac{\mathbf{f}_5^{DBM}\mathbf{f}_2^{OBC}\mathbf{f}_6^{QGL}\left(\mathbf{f}_3^{DBC} + \mathbf{f}_7^{DBC} - \mathbf{f}_2^{DBC} - \mathbf{f}_6^{DBC}\right)\mathbf{f}_3^{FBC}}{\mathbf{f}_1^{OBC}\mathbf{f}_9^{QGL}\left(\mathbf{f}_2^{DBC} + \mathbf{f}_8^{DBC} - \mathbf{f}_1^{DBC} - \mathbf{f}_9^{DBC}\right)\mathbf{f}_2^{FBC}\mathbf{f}_9^{FBC}}. \tag{3.15}$$

Experimental results show that the merged discriminating feature also performs excellently, though the MFF feature is more powerful.

3.2.2.4 Chen *et al.*'s Method

It is obvious that the overlapping median filter windows lead to certain correlations in a neighborhood of the MF image. Moreover, the median filter is based on order statistics, and is known to have good edge preserving capability. This makes the median filtered pixel neighbors correlate in a different way compared to those of the original or average filtered images. For median filtering detection, Chen and Ni [CN11] proposed the EBPM (Edge Based Prediction Matrix) feature to examine the statistical change among edge regions of the image. The given image is firstly divided into non-overlapping blocks, which are classified to three types. For each type of blocks, a neighborhood prediction model is employed, and the estimated prediction coefficients constitute the discriminating feature vector.

Chen *et al.* [Che+12, CNH13] further proposed a more powerful median filtering detection feature based on the analysis of statistical characterization in the pixel value difference domain. The proposed Global and Local Feature (GLF) vector $\mathbf{f}^{GLF}$ consists of two sub-vectors: the global probability feature vector $\mathbf{f}^{GPF}$, and the local correlation feature vector $\mathbf{f}^{LCF}$. The construction of $\mathbf{f}^{GPF}$ is based on the observation that the empirical cumulative distribution in the pixel value difference domain differs considerably among different types of images, *e.g.*, original, median filtered, and average filtered images. The motivation to build $\mathbf{f}^{LCF}$ is that the MF image has a rather distinct intrinsic fingerprint for the adjacent difference pair correlation in the pixel value difference domain.

3.2.2.5 Kang *et al.*'s Method

Kang *et al.* [PK12, Kan+12, Kan+13] proposed to use the median filter residual of a given image $\mathbf{u}$ (input of the forensic detector) to distinguish MF images from other kinds of images:

$$\mathbf{r} = \mathcal{MF}^{(1)}(\mathbf{u}) - \mathbf{u}. \tag{3.16}$$

In [PK12], Peng and Kang used the second-order Markov process [PBF10] to model the median filtered residual. The MFRTP (Median Filter Residual Transition Probabilities) feature vector is thereafter extracted and fed to the SVM for building the median filtering detector. Later, Kang *et al.* [Kan+12, Kan+13] adopted the autoregressive model to analyze the median filter residual, and the coefficients are extracted as the MFRAR (Median Filter Residual AutoRegressive) feature vector, which has a low dimensionality but still performs very well. Their methods also achieve good robustness of median filtering detection against JPEG compression post-processing.

3.2.2.6 Zhang *et al.*'s Method

Zhang *et al.* [Zha+14] investigated the median filtering from the angle of micro-texture structure, with the adoption of the so-called local ternary pattern [TT10] and local derivative

pattern [Zha+10]. In order to speed up the SVM training procedure, the dimensionality of the extracted MFLTP (Median Filter Local Ternary Patterns) feature is reduced from 2048 to 220 using the kernel principal component analysis technique, without harming the forensic performance of the detector.

3.2.3 Disguising Median Filtering Artifacts

3.2.3.1 Fontani and Barni's Method

Fontani and Barni [FB12] pioneered the median filtering anti-forensic work, against the forensic detectors proposed in [KF10, Cao+10, Yua11] (see Sections 3.2.2.1-3.2.2.3). The main idea of their method is to find a proper convolution filter f, so that the filtered MF image $f(\mathbf{y})$ has low detector output while still close to $\mathbf{y}$. The searching of f is performed by minimizing the following cost function:

$$\hat{f} = \arg\min_{f} \left(\exp\left(K_Y\left(f\left(\mathbf{y}\right)\right) - 2.2\right) - \mathrm{PSNR}\left(\mathbf{y}, f\left(\mathbf{y}\right)\right)\right), \tag{3.17}$$

where $K_Y(\cdot)$ (see Eq. (3.15)) returns the value of Yuan's merged discriminating feature [Yua11] for the input image, and the definition of $\mathrm{PSNR}(\cdot,\cdot)$ can be found in Section 2.2.2.1. The final anti-forensic MF image is generated by convolution filtering: $\tilde{\mathbf{x}} = \hat{f}(\mathbf{y})$.

3.2.3.2 Wu *et al.*'s Method

Wu *et al.* [WSL13] modeled the pixel value difference pair $\mathbf{d}^{i,j}(\mathbf{u}) = [\mathbf{U}_{i,j} - \mathbf{U}_{i,j+1}, \mathbf{U}_{i,j} - \mathbf{U}_{i+1,j}]^T$, using a 2-dimensional generalized Gaussian distribution. Targeting at median filtering forensic detectors in [KF10, PBF10, Yua11] (see Sections 3.2.2.1 and 3.2.2.3), a noise attack algorithm is thereafter proposed for median filtering anti-forensic purposes:

$$\mathbf{d}^{i,j}(\tilde{\mathbf{x}}) = \mathbf{d}^{i,j}(\mathbf{y}) + \mathbf{n}^{i,j}, \tag{3.18}$$

where $\mathbf{n}$ is the attacking noise. The goal is to bring the distribution of $\mathbf{d}^{i,j}(\tilde{\mathbf{x}})$ close to the estimated one of the original image pixel value difference $\mathbf{d}^{i,j}(\mathbf{x})$. The distribution of $\mathbf{n}^{i,j}$ is obtained based on the following two assumptions: the convolution of the distributions of $\mathbf{d}^{i,j}(\mathbf{y})$ and $\mathbf{n}^{i,j}$ yields the distribution of $\mathbf{d}^{i,j}(\mathbf{x})$; and the parameters of the two distributions of $\mathbf{d}^{i,j}(\mathbf{y})$ and $\mathbf{d}^{i,j}(\mathbf{x})$ have linear correlations.

3.2.3.3 Dang-Nguyen *et al.*'s Method

Dang-Nguyen *et al.* [DN+13] proposed another noise attack to the MF image, targeting at forensic detectors in [KF10, Cao+10, Yua11] (see Sections 3.2.2.1-3.2.2.3). The given image is firstly divided into non-overlapping blocks. The major noise attack is performed in relatively complex blocks, whose variances are greater than a certain threshold T. If the complex block

does not contribute to Yuan's $\mathbf{f}^{DBM}$ and $\mathbf{f}^{OBC}$ features [Yua11], the block is then dithered with random noise uniformly distributed in $[-7,-3] \cup [3,7]$. At last, the whole image is attacked by random noise uniformly distributed in $[0,1]$.

3.2.4 Summary

For the sake of conciseness, we henceforth use symbols for referring to the detectors considered in our work on median filtering anti-forensics, which are summarized in Table 3.3. Similar to Table 3.1, we mainly name the detectors directly after the feature value name for the scalar-based median filtering forensic detectors. As to the SVM-based detectors, the superscript 'S' together with the dimensionality of the feature vector is used to indicate that they are SVM-based. Note that, if the same (group of) authors have contributed more than one SVM-based detectors, we choose the latest (also the strongest) one for forensic testing.

Table 3.3: Median filtering forensic detectors.

K_K	Kirchner and Fridrich's [KF10] first-order pixel value difference histogram based detector;
$\hat{K}_K$	the block-based version of detector K_K [KF10];
K_C	Cao *et al.*'s [Cao+10] first-order pixel value difference based detector;
K_Y	Yuan's [Yua11] merged discriminating feature based detector;
K_{SPAM}^{S686}	the 686-dimensional SPAM feature based detector [PBF10];
K_{MFF}^{S44}	Yuan's [Yua11] 44-dimensional MFF feature based detector;
K_{GLF}^{S56}	Chen *et al.*'s [Che+12, CNH13] 56-dimensional GLF feature based detector;
K_{AR}^{S10}	Kang *et al.*'s [Kan+12, Kan+13] 10-dimensional MFRAR feature based detector;
K_{LTP}^{S220}	Zhang *et al.*'s [Zha+14] 220-dimensional MFLTP feature based detector.

From Section 3.2.3, we can see that the state-of-the-art median filtering anti-forensic methods use image processing operations (*e.g.*, convolution filtering [FB12]), or noise injection [WSL13, DN+13], in order to remove telltale artifacts from an MF image. The resulting anti-forensic MF image not necessarily achieves a good forensic undetectability against the forensic detectors listed in Table 3.3, and normally suffers from low visual quality.

In Chapter 7, we follow another research line for median filtering anti-forensics, leveraging on image deconvolution. We hope to create anti-forensic MF images with a better forensic undetectability against the detectors listed in Table 3.3 as well as a higher visual quality of the anti-forensic MF image, compared with the state-of-the-art anti-forensic MF images listed in Table 3.4[11].

[11] In [DN+13], Dang-Nguyen *et al.* show that their median filtering anti-forensic method outperforms Fontani and Barni's [FB12]. In Chapter 7, we refrain from comparing with Fontani and Barni's pioneer anti-forensic MF image [FB12], which is therefore not included in Table 3.4.

Table 3.4: Notations for original, MF, and state-of-the-art anti-forensic MF images.

$\mathcal{I}$	original, intact image;
$\mathcal{M}$	MF image;
$\mathcal{F}_W^M$	Wu *et al.*'s [Sta+10b, SL11] anti-forensic MF image created from $\mathcal{M}$ using the dithering operation (see Section 3.2.3.2);
$\mathcal{F}_D^M$	Dang-Nguyen *et al.*'s [DN+13] anti-forensic MF image created from $\mathcal{M}$ using the noise injection based method (see Section 3.2.3.3).

Chapter 4

Total Variation Based JPEG Anti-Forensics

Contents

4.1 Introduction and Motivation **48**
4.2 Performance Analysis of Scalar-Based JPEG Detectors **49**
4.2.1 Quantization Table Estimation Based Detector 49
4.2.2 Other Scalar-Based JPEG Forensic Detectors 51
4.3 JPEG Anti-Forensics via TV-Based Deblocking **53**
4.3.1 JPEG Deblocking Using Constrained TV-Based Minimization 53
4.3.2 De-Calibration 56
4.4 Experimental Results **56**
4.4.1 Parameter Settings 56
4.4.2 Comparison and Analysis 58
4.5 Summary **61**

In this chapter, we focus on removing the JPEG blocking artifacts at the anti-forensic level. To this end, we propose a constrained TV-based minimization problem, whose cost function consists of a TV term and a TV-based blocking measurement term. The resulting anti-forensic JPEG image successfully fools the forensic methods detecting JPEG blocking, and other advanced JPEG forensic detectors. The calibration-based detector is also defeated by conducting a further feature value minimization. Experimental results show that the proposed method outperforms the state-of-the-art JPEG anti-forensic methods in a better forensic undetectability and a higher visual quality of the processed image.

A paper describing the proposed method was published in an international conference [Fan+13a]. The Matlab code of the method is freely shared online and can be downloaded from: `http://www.gipsa-lab.grenoble-inp.fr/~wei.fan/documents/TV-AFJPG-ICASSP13.tar.gz`.

4.1 Introduction and Motivation

For JPEG anti-forensic purposes, we need to consider both the quantization artifacts in the DCT domain and the blocking artifacts in the spatial domain. In practice, we find it extremely difficult to conduct a single-step attack to defeat multiple JPEG forensic detectors that work in different domains, while keeping a high image visual quality. Therefore, in this thesis, we consider removing JPEG artifacts alternatively in the spatial and in the DCT domains. A similar strategy is adopted in Stamm *et al.*'s JPEG anti-forensic method [Sta+10a, Sta+10b, SL11], where the DCT-domain quantization artifacts and the spatial-domain blocking artifacts are handled separately in different domains.

For JPEG post-processing in image restoration, the quantization artifacts in the DCT domain is not the main concern. Its main objective is to remove the spatial-domain JPEG blocking artifacts for visual quality enhancement purposes [LY04, ADF05, Zha+08]. Unfortunately, these JPEG deblocking methods may not be able to create post-processed JPEG images which are able to pass the examination of JPEG blocking detectors. However, by integrating some additional anti-forensic terms, these methods can contribute in the JPEG blocking artifacts removing task at the anti-forensic level. Based on the above analysis, here we choose to firstly focus on the spatial-domain JPEG blocking artifacts. We leave the task of DCT-domain quantization artifacts removing in Chapter 5. Following the research line of conducting image anti-forensics leveraging on image restoration, in this chapter, we devise a JPEG anti-forensic method based on TV regularization.

The concept of Total Variation (*i.e.*, TV for short) regularization was pioneered by Rudin *et al.* [ROF92] in 1992. Ever since then, it has been widely used in image restoration, for various applications such as denoising, deconvolution, inpainting, *etc.* Take denoising as an example, the integral of the absolute gradient of the noised signal is high, in other words, the noised signal has high TV. Therefore, the signal restoration can be conducted by reducing the TV of the noised signal, usually with a signal fidelity constraint with respect to the noised signal. Compared with traditional techniques, *e.g.*, linear smoothing, and median filtering, the TV-based denoising is not only simple and effective in noise removal but also able to preserve details such as edges.

For JPEG image, the blocking artifacts in the spatial domain are presented in the form of pixel value discontinuities around the 8×8 block borders. This kind of artifacts leads to high TV of the JPEG image. Therefore, JPEG deblocking can be performed by minimizing a TV-based energy function [ADF05]. Traditionally, its main objective is to improve the visual quality of the JPEG image. In this chapter, we add another TV-based JPEG blocking measurement term into the TV-based minimization cost function. By solving the new minimization problem, we may need to sacrifice some image quality of the JPEG image. However, we hope to create anti-forensic JPEG images which are able to fool existing scalar-based JPEG forensic detectors.

The remainder of this chapter is organized as follows. Section 4.2 experimentally analyzes the forensic performance of 8 existing scalar-based JPEG forensic detectors listed in Table 3.1.

In Section 4.3, we describe the proposed TV-based deblocking method for JPEG anti-forensics. Experimental results with comparisons with state-of-the-art JPEG anti-forensic methods are presented in Section 4.4. At last, Section 4.5 summarizes this chapter.

4.2 Performance Analysis of Scalar-Based JPEG Detectors

In this section, we experimentally analyze the performance of the 8 scalar-based JPEG detectors. As to the other 2 SVM-based JPEG detectors listed in Table 3.1, we leave it to be discussed in Chapter 5.

4.2.1 Quantization Table Estimation Based Detector

Fan and De Queiroz's [FD03] JPEG forensic detector K_F^Q (see Section 3.1.4 and Table 3.1) classifies a given image as not compressed, if all the estimated quantization steps are either 1 or "undetermined" [Sta+10a]. It is the targeted quantization artifacts detector of the JPEG anti-forensic method proposed in [SL11]. However, as pointed out by Böhme and Kirchner [BK13], we lack knowledge on the false positive rate of K_F^Q [FD03].

In order to investigate the reliability of K_F^Q [FD03], we estimated the quantization table of $10,000$ genuine, uncompressed 512×512 PGM images from BOSSBase dataset [BFP11] (see the end of Section 2.3.1 for more descriptions and discussions concerning this dataset). Table 4.1 lists the false positive rate P_{FP} of all the 10 archives of BOSSBase. Surprisingly, P_{FP} can reach as high as 61.50% for the 6-th archive. The average rate 25.22% also indicates that K_F^Q is not very reliable to determine whether an image has been JPEG compressed.

Furthermore, we found that among all the "non-1" and "non-undetermined" estimated entries by K_F^Q [FD03], 3 takes a large portion, as shown in the P_3 column in Table 4.1. Figure 4.1-(a) is an example DCT histogram from a never compressed image, which has no comb-like quantization artifacts but is detected as quantized by 3 according to K_F^Q. Here we go back to review the MLE of quantization table [FD03]. For the sake of simplicity, we only analyze why K_F^Q picks 3 instead of 1 in this case. Based on the Eq. (14) of [FD00] and the Eq. (15) of [FD03], for quantization step $q = 1$, the log-likelihood is:

$$L(1) = N \times w(0,1) + N \times \log 1, \tag{4.1}$$

while the log-likelihood for $q = 3$ is:

$$L(3) = p_0 \times N \times w(0,3) + (1 - p_0) \times N \times w(1,3) + N \times \log 3, \tag{4.2}$$

where N is the total number of blocks used in estimation, and $w(i,q)$ is an even function of i defined in [FD00] (more details can be found in [FD00, FD03]). We denote p_0 as the percentage of coefficients which are the integer multiples of 3. K_F^Q chooses 3 instead of 1

Table 4.1: P_{FP} and P_3 for detector K_F^Q [FD03] on BOSSBase dataset [BFP11]. P_{FP} is calculated as the portion of images which are mistakenly classified as JPEG compressed among all the original images under examination. P_3 is the fraction of the number of estimated quantization steps being 3 over the number of the estimated "non-1" and "non-undetermined" quantization steps.

Archive	$P_{FP}(\%)$	P_3
01	18.70	$1653/1959 = 84.38\%$
02	12.40	$957/1118 = 85.60\%$
03	4.50	$175/187 = 93.58\%$
04	5.70	$111/125 = 88.80\%$
05	10.50	$986/1141 = 86.42\%$
06	61.50	$9243/11033 = 83.78\%$
07	17.50	$1084/1155 = 93.85\%$
08	43.90	$5655/6670 = 84.78\%$
09	33.20	$5062/5895 = 85.87\%$
10	44.30	$6780/7720 = 87.82\%$
Average	25.22	87.49%

when $L(3) > L(1)$. This happens when $p_0 > 67.28\%$. For Figure 4.1-(a), among all 4083[12] DCT coefficients in use for quantization step estimation, there are 2797 coefficients which are integer multiples of 3, namely $p_0 = 68.50\%$. Hence K_F^Q detects that it is quantized by 3. This frequently happens in the high-frequency subbands for relatively smooth images.

Another interesting point is that K_F^Q outputs 1 for some DCT histograms which are obviously not from genuine, uncompressed images. Figure 4.1-(b) is from a post-processed JPEG image. It is obvious that the comb-like quantization artifacts still remain, but K_F^Q fails to estimate the correct quantization step.

When K_F^Q is tested on UCIDTest dataset (see Section 2.3.1 for more details concerning this dataset) with 1000 UCID images [SS04]. The false positive rate is $P_{FP} = 16/1000 = 1.6\%$, and $P_3 = 15/16 = 94.74\%$. This low false positive rate is due to the fact that most UCID images contain a large portion of highly textured regions, which decreases the occurrences of 3 in the quantization step estimation. Nevertheless the weakness of K_F^Q is exposed when tested on the dataset BOSSBase where images are relatively smooth (also see the discussion in the end of Section 2.3.1).

From the above analysis of detector K_F^Q, it can be seen that it is not a very reliable forensic detector to determine whether an image has been previously JPEG compressed. However, in our JPEG anti-forensic work, we still consider the outputs of K_F^Q trying to decrease the

[12] For an image with size of 512×512, there should be 4096 DCT coefficients in each subband. However, for estimating the quantization step [FD03], the 8×8 block with equal value for all the 64 pixels is excluded in the MLE. Therefore, here in this example DCT histogram, there are only 4083 instead of 4096 DCT coefficients used in the estimation.

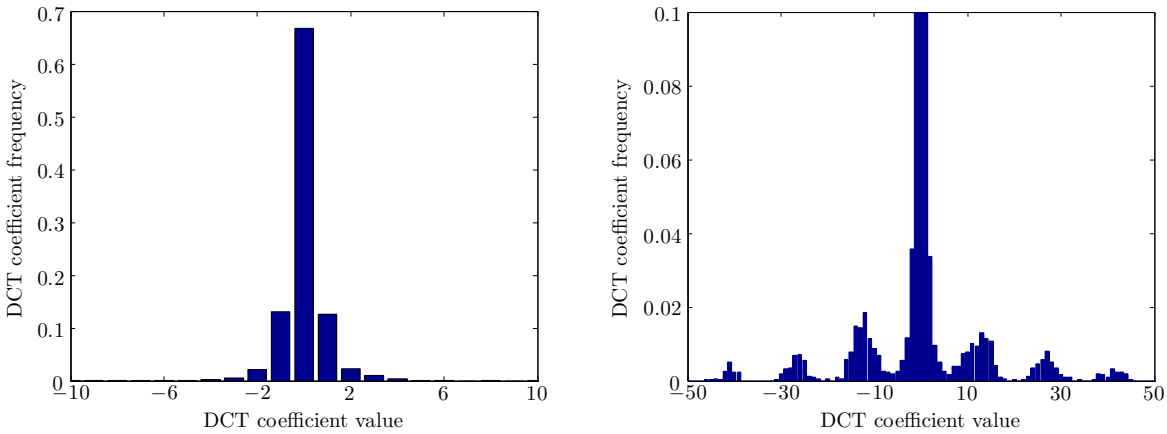


Figure 4.1: (a) is an example unquantized DCT histogram which is detected as quantized by 3 according to K_F^Q [FD03], while the quantization step is detected as 1 (namely never quantized) for (b), which is from a post-processed JPEG image.

occurrences of 3 in the anti-forensic JPEG image on UCID dataset [SS04], so as to prevent the processed image from getting over-smoothed.

4.2.2 Other Scalar-Based JPEG Forensic Detectors

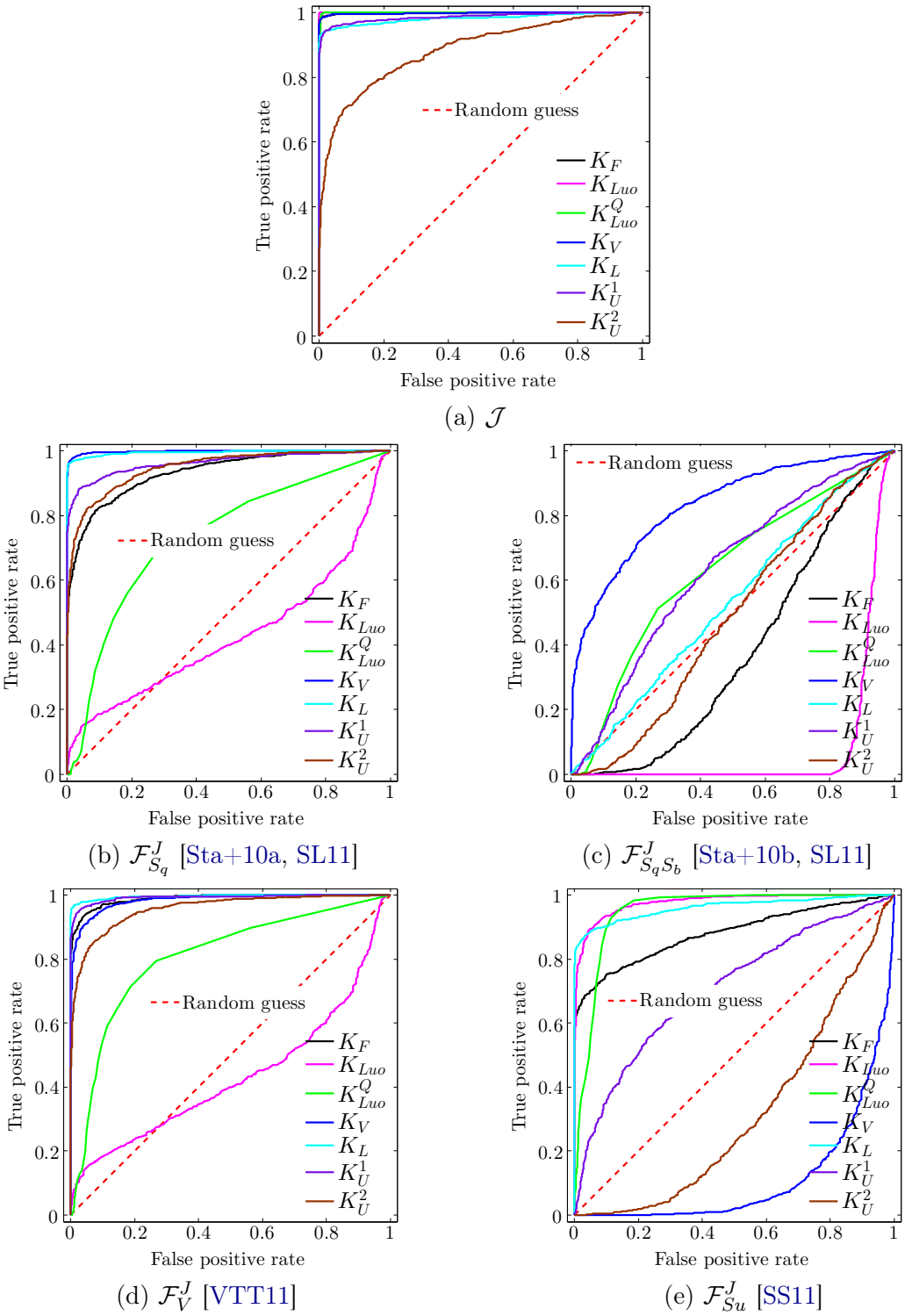
Besides the JPEG forensic detectors descried in Sections 3.1.4, 3.1.6 and 3.1.7, here we further propose to build a family of measures, for detecting JPEG blocking artifacts:

$$K_U^p = \left| B_{gr}^p(\mathbf{U}) - B_{gr}^p(\mathbf{U}_{cal}) \right|, \tag{4.3}$$

where B_{gr}^p is the gradient aware blockiness [UW08], which is the normalized ℓ_p norm of the weighted gradient computed from each group of four adjacent pixel values across 8×8 block borders (see [UW08] for details). Note that the parameter p can vary to build a family of different measures. In this thesis, we use $p = 1$ and $p = 2$ to build two JPEG blocking artifacts detectors K_U^1 and K_U^2, respectively.

Besides the analysis of the quantization table estimation based detector K_F^Q in Section 4.2.1, in this section, we study the other scalar-based JPEG forensic detectors. Here, we report the ROC curves obtained by the (state-of-the-art anti-forensic) JPEG images listed in Table 3.2 against the 7 other scalar-based JPEG forensic detectors listed in Table 3.1.

Shown in Figure 4.2 are ROC curves achieved by 7 scalar-based JPEG forensic detectors examining 5 kinds of (state-of-the-art anti-forensic) JPEG images on UCIDTest dataset. In Figure 4.2-(a), all the 7 ROC curves are all close to the upper-left point $(0, 1)$ in the ROC space (the performance of detector K_U^2 is slightly worse compared with others). It indicates that all these JPEG forensic detectors perform well to detect JPEG compression without any anti-forensics applied.



(a) $\mathcal{J}$

(b) $\mathcal{F}^J_{S_q}$ [Sta+10a, SL11]

(c) $\mathcal{F}^J_{S_q S_b}$ [Sta+10b, SL11]

(d) $\mathcal{F}^J_V$ [VTT11]

(e) $\mathcal{F}^J_{Su}$ [SS11]

Figure 4.2: ROC curves achieved by 5 kinds of (state-of-the-art anti-forensic) JPEG images against 7 scalar-based JPEG forensic detectors. Results are obtained on UCIDTest dataset.

Detectors K_V [Val+11, VTT13] and K_L [LB11] are proposed aiming at detecting the anti-forensic JPEG image $\mathcal{F}^J_{S_q}$ [Sta+10a, SL11]. Therefore, from Figure 4.2-(b), we can see that both the two ROC curves respectively achieved by K_V and K_L are close to the upper-left point of the ROC space, which implies a good forensic detectability. Valensize *et al.* [VTT11] improved the dithering based anti-forensic method of Stamm *et al.* [Sta+10a, SL11], by employing the JND criterion to control the distortion introduced by the anti-forensic dithering operation. The resulting anti-forensic JPEG image $\mathcal{F}^J_V$ [VTT11] is able to achieve higher visual quality (see Table 4.2 for relevant quality comparison) but about the same level of forensic undetectability, compared with $\mathcal{F}^J_{S_q}$ [Sta+10a, SL11].

The anti-forensic JPEG image $\mathcal{F}^J_{S_qS_b}$ [Sta+10b, SL11] is obtained by processing $\mathcal{F}^J_{S_q}$ [Sta+10a, SL11] using median filtering combined with white Gaussian noise addition. The ROC curves in Figure 4.2-(c) show that median filtering performs very well in disguising JPEG blocking artifacts, against detectors K_F [FD03], K^1_U and K^2_U. Compared with other state-of-the-art anti-forensic JPEG images, $\mathcal{F}^J_{S_qS_b}$ [Sta+10b, SL11] is able to drag the ROC curves the closest to the random guess. However, the ROC curves achieved by K_{Luo} [LHQ10] and K_V [Val+11, VTT13] are still quite far away from the random guess. Moreover, Böhme and Kirchner [BK13] point out that $\mathcal{F}^J_{S_qS_b}$ [Sta+10b, SL11] has not been examined by median filtering forensic detectors yet. In Chapter 7, we will further discuss this issue.

From Figure 4.2-(e), it can be seen that Sutthiwan and Shi's [SS11] anti-forensic JPEG image $\mathcal{F}^J_{Su}$ can be detected by most of the 7 JPEG forensic detectors in consideration. The exceptions are K^1_U and K^2_U, which indicates that the blocking artifacts are well suppressed by the SAZ operation, yet still remain detectable by K_F [FD03].

4.3 JPEG Anti-Forensics via TV-Based Deblocking

For JPEG anti-forensic purposes, we need to remove from the JPEG image the quantization artifacts in the DCT domain as well as the blocking artifacts in the spatial domain. Unlike [Sta+10a, Sta+10b, SL11], here we firstly focus on the blocking artifacts removal, leveraging on the widely used TV in the image restoration field.

4.3.1 JPEG Deblocking Using Constrained TV-Based Minimization

In image restoration, researchers have investigated the problem of removing JPEG blocking artifacts. However, their efforts mainly focused on improving the image visual quality [LY04, ADF05, Zha+08], especially for highly compressed images. Anti-forensics should take into account both the forensic undetectability and the perceptual quality. For JPEG deblocking purposes, we hereby propose a variational approach to minimize a TV-based energy function consisting of a TV term and a TV-based blocking measurement term.

Inspired by [ADF05], which aims to improve the visual quality of JPEG images compressed at low bit-rates by solving a constrained and weighted TV minimization problem, for any given

image $\mathbf{U}$ of size $H \times W$, we first define the TV term as:

$$\mathrm{TV}_b(\mathbf{U}) = \sum_{1 \le i \le H, 1 \le j \le W} \nu_{i,j}, \tag{4.4}$$

with the variation at location (i, j) defined as:

$$\nu_{i,j} = \sqrt{(\mathbf{U}_{i-1,j} + \mathbf{U}_{i+1,j} - 2\mathbf{U}_{i,j})^2 + (\mathbf{U}_{i,j-1} + \mathbf{U}_{i,j+1} - 2\mathbf{U}_{i,j})^2}, \tag{4.5}$$

where $\mathbf{U}_{i,j}$ is the value of the (i, j)-th pixel of image $\mathbf{U}$.

In order to remove the statistical traces of JPEG blocking artifacts, we define a second term which measures the JPEG blocking. The idea is very simple: it assumes that if there is no JPEG compression, statistically the energy sum of the pixel value variation along the block borders should be close to that within the block. In other words, the energy sum shall statistically remain the same no matter where the 8×8 block starts. Hence, we divide all the pixels in the image into two sets according to their positions in the block. The pixel classification strategy is illustrated in Figure 4.3. Pixels at shaded locations are put into set $\mathcal{A}$, while the others are put into set $\mathcal{B}$. Based on this, the second energy term is defined as:

$$C(\mathbf{U}) = \left| \sum_{\mathbf{U}_{i,j} \in \mathcal{A}} \nu_{i,j} - \sum_{\mathbf{U}_{i,j} \in \mathcal{B}} \nu_{i,j} \right|. \tag{4.6}$$

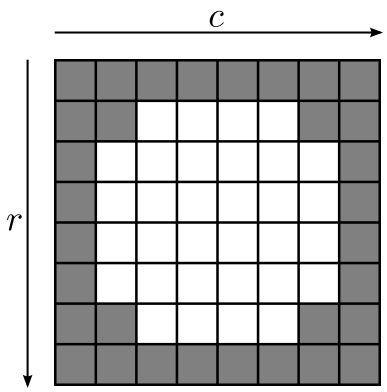


Figure 4.3: Pixel classification according to its position in the 8×8 pixel value block.

The rationale behind this simple idea of using TV and pixel classification (according to Figure 4.3) to conduct JPEG deblocking can be backed by experimental results shown in Figure 4.4. We can see that for original UCIDTest images (see Section 2.3.1 for more descriptions about the datasets), the values of $C(\mathbf{U})/\mathrm{TV}_b(\mathbf{U})$ (here, the dividing by $\mathrm{TV}_b(\mathbf{U})$ is for normalization purposes, since the TV varies largely for different images) are close to 0. After JPEG compression, this ratio value largely increases. Therefore, we hope to remove the blocking artifacts from a given JPEG image by minimizing the two energy terms described in Eqs. (4.4) and (4.6).

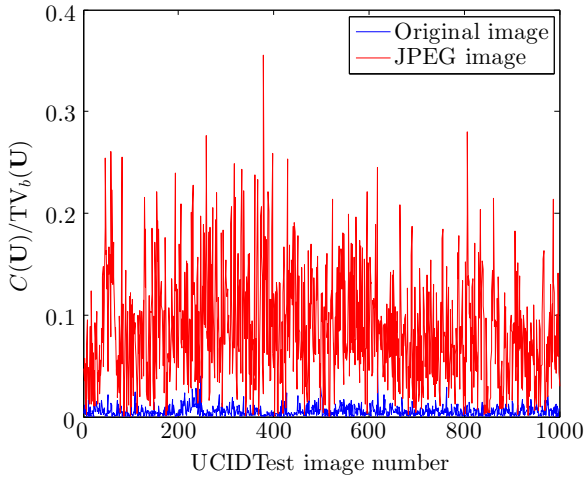


Figure 4.4: TV-based blocking measurement test conducted on original images and JPEG images, respectively. Results are obtained on UCIDTest dataset.

We also adopt a similar, yet more flexible constraint (controlled by parameter μ, a small positive number) than that in [ADF05], with the objective to achieve a good quality of the processed image. Denote $\mathbb{M}$ as the set of integers within the range $[0, 255]$. We define the constraint image space $\mathcal{S}$ as:

$$\mathcal{S} = \left\{ \mathbf{U} \in \mathbb{M}^{H\times W} \middle| (\mathbf{DU})^l_{r,c} \in \left[(k^l_{r,c} - \mu)Q_{r,c}, (k^l_{r,c} + \mu)Q_{r,c} \right]; r, c = 1, 2, \cdots, 8; l = 1, 2, \cdots, L \right\}, \tag{4.7}$$

where $k^l_{r,c} = (\mathcal{Q}(\mathbf{DI}))^l_{r,c}$ are the quantized DCT coefficients, which can be read from the JPEG image $\mathcal{J}$ using the Matlab JPEG toolbox [Sal03]. We set this constraint space to make sure that the DCT coefficients of the processed image are within the same quantization bins (if $\mu \leq 0.5$), or in the same or neighboring bins (if $\mu > 0.5$), as those of the JPEG image $\mathcal{J}$, so as to ensure an acceptable quality of the processed image.

The final constrained TV-based minimization problem is:

$$\mathbf{U}^* = \arg\min_{\mathbf{U}\in\mathcal{S}} E(\mathbf{U}) = \arg\min_{\mathbf{U}\in\mathcal{S}} \left(\mathrm{TV}_b(\mathbf{U}) + \alpha C(\mathbf{U}) \right), \tag{4.8}$$

where $\alpha > 0$ is a regularization parameter, balancing the two energy terms. It is easy to demonstrate that $E(\mathbf{U})$ is a convex function (though not differentiable) and $\mathcal{S}$ is a convex set [ADF05, BV04]. The optimization problem can be solved using the projected subgradient method (see Section 2.4.1 for more descriptions) leading to the iteration:

$$\mathbf{U}^{(k+1)} = P_{\mathcal{S}} \left(\mathbf{U}^{(k)} - t_k \times g(\mathbf{U}^{(k)}) \right), \tag{4.9}$$

where $\mathbf{U}^{(k)}$ is the processed image at the k-th iteration (note that, $\mathbf{U}^{(0)}$ is the given JPEG image), $t_k > 0$ is the step size, $g(\mathbf{U})$ is a subgradient of $E(\mathbf{U})$, and $P_{\mathcal{S}}$ is the projection operator onto space $\mathcal{S}$ that will be explained in detail later.

4.3.2 De-Calibration

In practice, the TV-based JPEG deblocking method is able to generate anti-forensic JPEG images which can well fool the 8 scalar-based JPEG forensic detectors listed in Table 3.1 except the calibration-based one K_L [LB11] (see Section 3.1.6). In fact the calibrated feature value, *i.e.*, K_L of Eq. (3.10), has also been significantly decreased. However, for genuine, uncompressed images, this feature value is highly condensed in an interval of very small values. It is hard to further decrease this value by performing deblocking on JPEG images, while keeping a good visual quality.

In this section, we will directly optimize an energy function which is very close to Eq. (3.10) for *de-calibration* purposes. The minimization problem is formulated as:

$$\mathbf{U}^* = \arg\min_{\mathbf{U}} \sum_{k=1}^{28} \Big| \mathrm{var}(\mathbf{D}_k \mathbf{U}) - \mathrm{var}(\mathbf{D}_k \mathbf{U}_{cal}) \Big|, \tag{4.10}$$

which can also be solved using the subgradient method (see Section 2.4.1 for more descriptions).

4.4 Experimental Results

Our large-scale forensic tests are conducted on UCIDTest dataset (see Section 2.3.1 for more descriptions about the datasets) with 1000 original, uncompressed UCID images [SS04]. From each uncompressed image $\mathcal{I}$, a JPEG image $\mathcal{J}$ is obtained by compressing $\mathcal{I}$ with a random quality factor chosen from $\{50, 51, \cdots, 95\}$. Thereafter, one kind of post-processed JPEG image and two kinds of anti-forensic JPEG images are thereafter created from $\mathcal{J}$:

- $\mathcal{J}_A$, created from $\mathcal{J}$ with the application of Alter *et al.*'s deblocking method [ADF05];
- $\hat{\mathcal{F}}_0^J$, created from $\mathcal{J}$ with the application of the proposed TV-based JPEG deblocking method described in Section 4.3.1;
- $\mathcal{F}_0^J$, created from $\hat{\mathcal{F}}_0^J$ with the application of the proposed de-calibration operation described in Section 4.3.2.

In this section, the above three kinds of images will be compared with (state-of-the-art anti-forensic) JPEG images listed in Table 3.2 in terms of both image quality and forensic undetectability against existing JPEG forensic detectors listed in Table 3.1.

4.4.1 Parameter Settings

Instead of waiting for the convergence of the optimization problem in Eq. (A.4), we have a different strategy to select the candidate deblocked image, which may not be the solution

$\mathbf{U}^*$ of the minimization problem. The selection is guided by the JPEG blocking signature measure K_F [FD03] in Eq. (3.6). Indeed, in practice we found that for uncompressed images, the output of K_F has a smaller standard deviation than another blocking signature K_U^p in Eq. (4.3). Its detection ability is also proven to be stronger than K_U^p when tested on JPEG images (see Figure 4.2-(a)). Therefore, although it is difficult to include K_F in our optimization framework as it is histogram based, it would be reasonable to use K_F to guide the selection of the deblocked result. Experimentally, we run 50 iterations, and choose the resulting image giving the smallest K_F value as the final result. This gives us satisfying deblocked JPEG image $\hat{\mathcal{F}}_0^J$ which is able to achieve a good forensic undetectability against all the three blocking artifacts detectors K_F, K_U^1 and K_U^2.

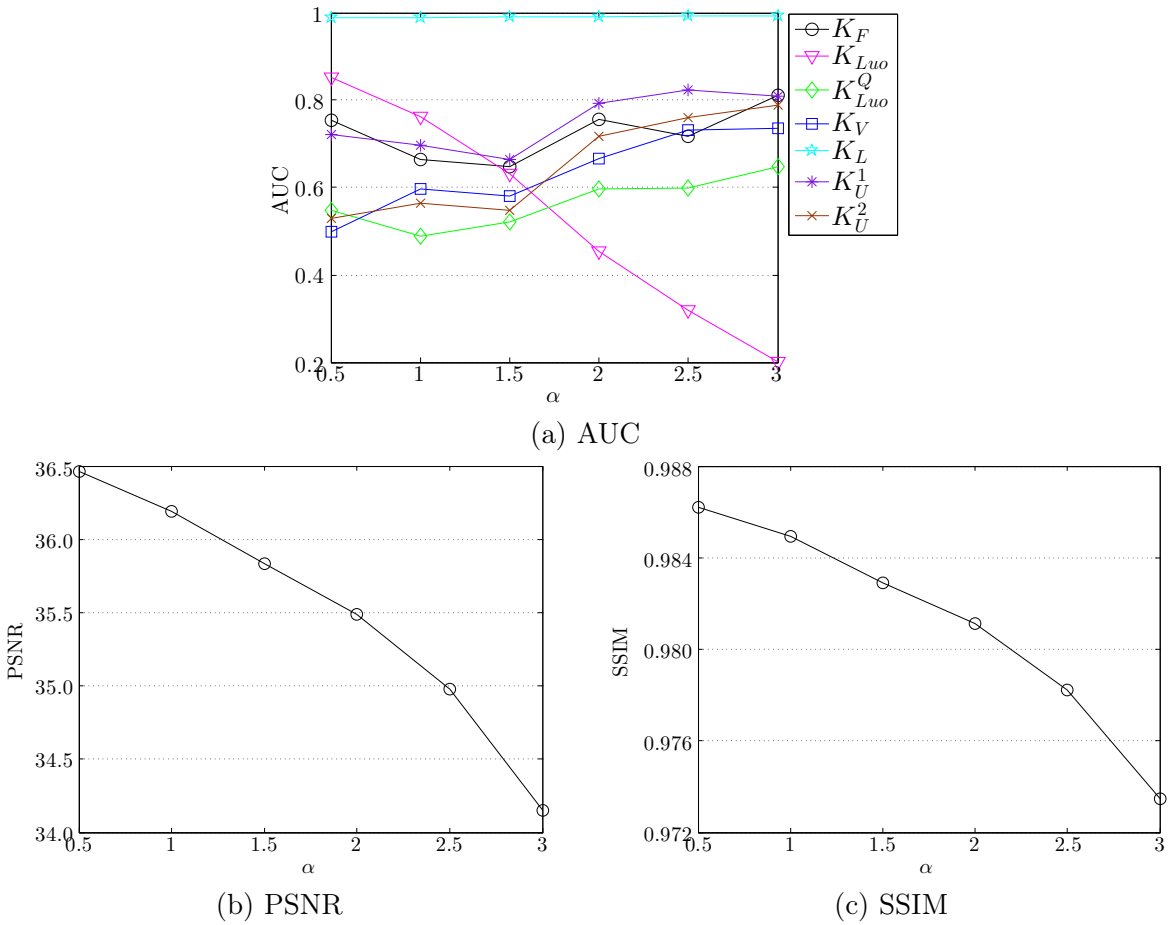
Furthermore, the regularization parameter α in Eq. (A.4), μ in Eq. (4.7), and the step size t_k in Eq. (4.9) are parameters that we can adjust. Before the discussion about the setting of α, we discuss about settings for μ and t_k first.

As to the setting of the convex set $\mathcal{S}$, here we set $\mu = 1.5$, which strictly constrains the processed DCT coefficient to stay within the same (or the neighboring) quantization bin of its original value. If μ is set to be too small, the resulting anti-forensic JPEG image will have obvious DCT-domain quantization artifacts. On the other hand, if μ is set to be too big, the resulting anti-forensic JPEG image will suffer from low visual quality. By setting $\mu = 1.5$, a good tradeoff can be achieved. The projection operator $P_{\mathcal{S}}$ works as follows: once a DCT coefficient under processing goes outside the constrained range, it will be modified back to a random value uniformly distributed within the original quantization bin. In the spatial domain the resulting pixel values will at last be rounded and truncated to integers in the range $[0, 255]$. In fact, the $P_{\mathcal{S}}$ operator can be taken as a modified QCS projection which is often used in JPEG image post-processing [YGK95, RS05, SC07].

For the step size t_k in Eq. (4.9), it is set to $t_k = 1/k$ $(k = 1, 2, \cdots, 50)$ at the k-th iteration, following [ADF05].

The regularization parameter α is mainly for balancing the two energy terms $\mathrm{TV}_b(\mathbf{U})$ and $C(\mathbf{U})$. We need to consider the AUC values achieved by the detectors as well as the visual quality of the anti-forensic JPEG image, for choosing a proper value of α. In order to well tune this parameter, here we conduct the experiments on UCIDTest92 (see Section 2.3.1 for details), with α varying in $\{0.5, 1.0, 1.5, 2.0, 2.5, 3.0\}$. For each α with value in this set, a set of anti-forensic JPEG images $\hat{\mathcal{F}}_0^J$ are generated and tested against the 7 scalar-based JPEG forensic detectors. Figure 4.5-(a) plots the AUC curves as a function of the value of α for different JPEG detectors. Since we take special care of the calibration-based JPEG forensic detector K_L [LB11] in a subsequent processing, here we mainly consider the remaining 6 detectors. The image quality variation can also be seen in Figure 4.5-(b) and -(c) for evaluation metrics PSNR (see Section 2.2.2.1) and SSIM (see Section 2.2.2.2) with the original image $\mathcal{I}$ as the reference, respectively. It can be seen that, the greater the value of α is, the poorer the quality of the resulting image $\hat{\mathcal{F}}_0^J$ will be. When $\alpha = 1.5$, the anti-forensic JPEG image $\hat{\mathcal{F}}_0^J$ is able to achieve a good tradeoff between forensic undetectability and visual quality.

For the de-calibration operation, as we almost directly minimize the calibrated feature



(a) AUC

(b) PSNR

(c) SSIM

Figure 4.5: The variation trend of the forensic undetectability against 7 scalar-based JPEG forensic detectors and image quality for $\hat{\mathcal{F}}_0^J$ under different settings of α (see Eq. (A.4)). Results are obtained on UCIDTest92 dataset.

value, we are able to obtain very small K_L values when converging to $\mathbf{U}^*$ in Eq. (A.5). In order to fool the detector, a random threshold for each image is drawn from the distribution of the calibrated feature values for genuine, uncompressed images, and the iteration stops once the K_L value goes below it.

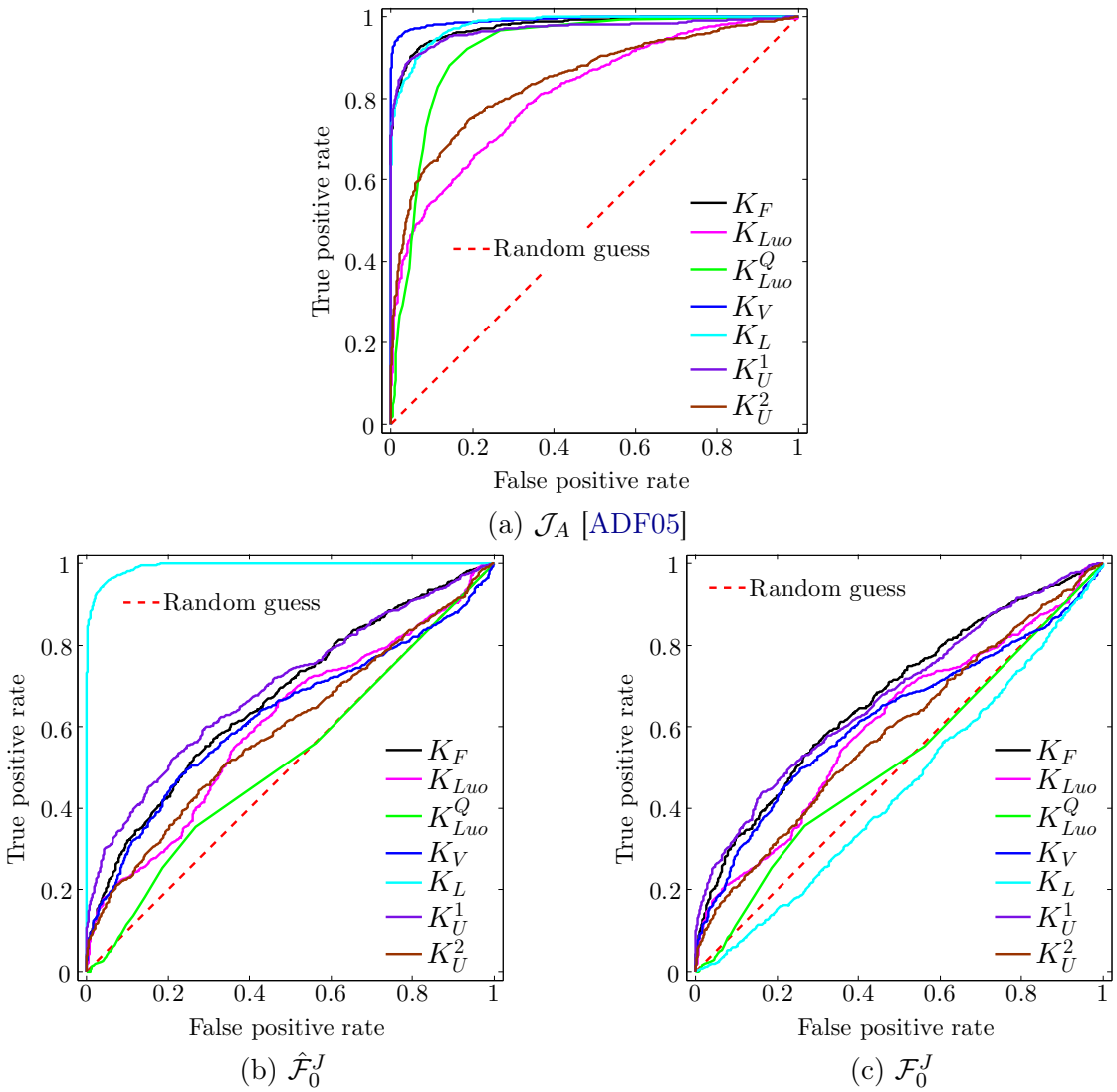
4.4.2 Comparison and Analysis

Table 4.2 reports the average PSNR and SSIM values, for $\mathcal{F}_{S_q}^J$ [Sta+10a, SL11], $\mathcal{F}_{S_qS_b}^J$ [Sta+10b, SL11], $\mathcal{F}_V^J$ [VTT11], $\mathcal{F}_{Su}^J$ [SS11], $\mathcal{J}_A$ [ADF05], $\hat{\mathcal{F}}_0^J$, and $\mathcal{F}_0^J$, respectively. The image quality evaluation metric values are calculated using the original image $\mathcal{I}$ as the reference, on UCIDTest dataset. The ROC curves achieved by different kinds of images against 7 scalar-based JPEG forensic detectors are plotted in Figure 4.2 (in Section 4.2.2) and Figure 4.6.

Apparently, fooling detectors is not the goal of Alter *et al.*'s work [ADF05], as their main focus is to improve the image perceptual quality. Note that in Table 4.2, the average PSNR

Table 4.2: Image quality comparison of different kinds of (anti-forensic/post-processed) JPEG images, with the original image $\mathcal{I}$ as the reference. Results are obtained on UCIDTest dataset.

	$\mathcal{J}$	$\mathcal{F}^J_{S_q}$	$\mathcal{F}^J_{S_qS_b}$	$\mathcal{F}^J_V$	$\mathcal{F}^J_{Su}$	$\mathcal{J}_A$	$\hat{\mathcal{F}}^J_0$	$\mathcal{F}^J_0$
PSNR	37.0999	33.4061	30.4591	33.2890	31.6552	35.8541	35.5070	35.4814
SSIM	0.9919	0.9756	0.9509	0.9802	0.9719	0.9884	0.9844	0.9843



(a) $\mathcal{J}_A$ [ADF05]

(b) $\hat{\mathcal{F}}^J_0$

(c) $\mathcal{F}^J_0$

Figure 4.6: ROC curves achieved by different kinds of (anti-forensic/post-processed) JPEG images against 7 scalar-based JPEG forensic detectors. Results are obtained on UCIDTest dataset.

and SSIM values of $\mathcal{J}_A$ are both lower than those of $\mathcal{J}$. The reason may be that the parameter setting in [ADF05] is optimized for low bit-rate compression, but not for the JPEG compression with relatively high quality factor in $\{50, 51, \cdots, 95\}$.

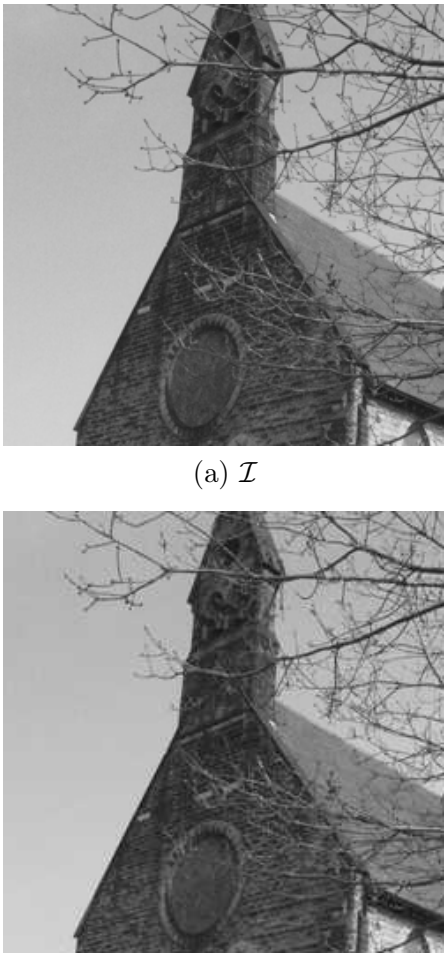
As analyzed in Section 4.2.2, $\mathcal{F}_{S_q}$ [Sta+10a, SL11] and $\mathcal{F}_V$ [VTT11] have similar anti-forensic performance against the 7 scalar-based JPEG forensic detectors. Both of them can be well detected by the JPEG blocking detectors K_F [FD03], K_U^1 and K_U^2, as well as the detectors K_V [Val+11, VTT13] and K_L [LB11] which are especially designed to attack the dithering based JPEG anti-forensic method [Sta+10a, SL11]. As an improved version of $\mathcal{F}_{S_q}$ by employing the JND criterion to control the distortion in the spatial domain, we can see from Table 4.2 that $\mathcal{F}_V$ achieves a higher average SSIM value than $\mathcal{F}_{S_q}$. The median filtering based JPEG deblocking method [Sta+10b, SL11] improves the forensic undetectability [Sta+10a, SL11], however, from Table 4.2, we can see that the cost of the resulting anti-forensic JPEG image $\mathcal{F}_{S_qS_b}$ [Sta+10b, SL11] is to lose 2.95 dB of PSNR value and 0.0247 of SSIM value on average. As to the anti-forensic JPEG image $\mathcal{F}_{Su}^J$ generated using Sutthiwan and Shi's [SS11] SAZ attack, it can be well detected by most of the detectors meanwhile suffers from a relatively low visual quality.

From the ROC curves plotted in Figure 4.6-(b) and the average PSNR and SSIM values listed in Table 4.2 for $\hat{\mathcal{F}}_0^J$, it can be seen that the proposed TV-based JPEG deblocking method is able to well fool the JPEG blocking detectors K_F [FD03], K_U^1 and K_U^2 while keeping a good visual quality of the processed image. Besides, Luo *et al.*'s [LHQ10] two JPEG forensic detectors K_{Luo} and K_{Luo}^Q are also well fooled. Moreover, $\hat{\mathcal{F}}_0^J$ is also able to fool another advanced detector K_V [Val+11, VTT13]. The reason may be that the TV term of Eq. (4.4) suppresses the unnatural noises that can be detected by K_V. After processing $\hat{\mathcal{F}}_0^J$ with the de-calibration operation (see Section 4.3.2), the resulting final anti-forensic image $\mathcal{F}_0^J$ is able to drag the 7 ROC curves the closest to the random guess, compared with the state-of-the-art anti-forensic JPEG images. By comparing the average PSNR and SSIM values achieved by $\hat{\mathcal{F}}_0^J$ and $\mathcal{F}_0^J$ listed in Table 4.2, it can be seen that the de-calibration operation does not harm the image quality of $\hat{\mathcal{F}}_0^J$, but can successfully defeat Lai and Böhme's [LB11] calibration-based detector K_L.

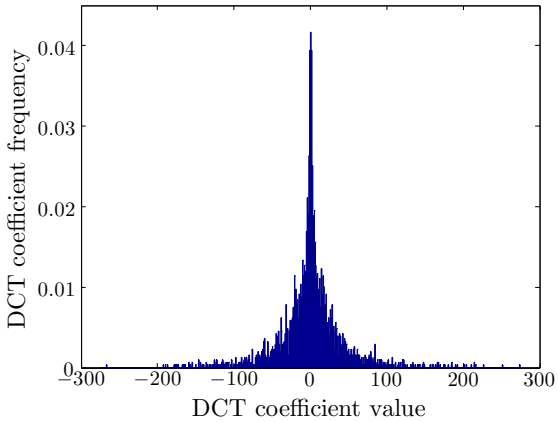
Among all the anti-forensic JPEG images in consideration, $\mathcal{F}_0^J$ generated using the proposed JPEG anti-forensic method is able to achieve a better forensic undetectability as well as a higher image quality, compared with the state-of-the-art anti-forensic JPEG images. Compared with Stamm *et al.*'s [Sta+10a, Sta+10b, SL11] $\mathcal{F}_{S_qS_b}^J$, besides a better anti-forensic performance, $\mathcal{F}_0^J$ has a gain of 5.02 dB of PSNR value and 0.0334 of SSIM value on average.

Figures 4.7-4.9 show some example results obtained from a UCID image [SS04]. The JPEG image $\mathcal{J}$ is obtained by compressing the original image $\mathcal{I}$ with quality factor 50. Compared with Stamm *et al.*'s anti-forensic JPEG images $\mathcal{F}_{S_q}^J$ and $\mathcal{F}_{S_qS_b}^J$, our anti-forensic JPEG image $\mathcal{F}_0^J$ better preserves the image details such as textures and edges. As pointed out by Valensize *et al.* [VTT11], the dithering operation [Sta+10a, SL11] introduces extra noise into the anti-forensic image $\mathcal{F}_{S_q}^J$. Though the median filtering based deblocking operation [Sta+10b, SL11] can mitigate the spatial-domain noise introduced by the dithering signal, it can still be noticed at the smooth areas of the image, *e.g.*, the sky area in Figure 4.9-(a) (please refer to the electronic version for a better visibility).

Figure 4.9-(e) show an example close-up image of $\mathcal{F}_0^J$, it can be seen that it has a better

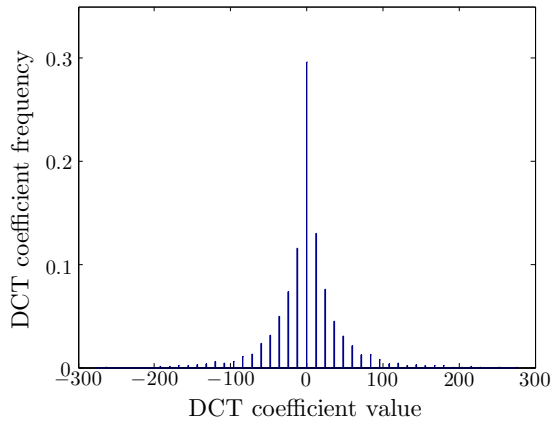


(a) $\mathcal{I}$



(b) $\mathcal{I}$

(c) $\mathcal{J}$



(d) $\mathcal{J}$

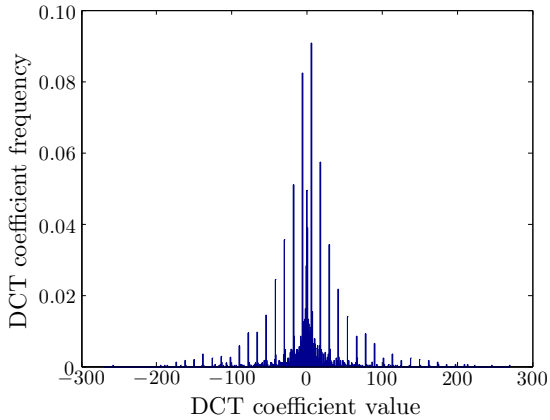
Figure 4.7: Example results (close-up images) and DCT histograms of the $(2, 2)$ subband, for $\mathcal{I}$ and $\mathcal{J}$, respectively.

visual quality than $\mathcal{F}^J_{S_q}$ and $\mathcal{F}^J_{S_q S_b}$ (see Figures 4.9-(a) and -(c)). Moreover, it can be seen from Figure 4.9-(f) that the shape of the DCT histogram of $\mathcal{F}^J_0$ has been improved, though in our JPEG deblocking method no specially designed DCT smoothing is conducted. According to the quantization table estimation based detector K^Q_F [FD03], 85.70% of $\mathcal{F}^J_0$ images are classified as never JPEG compressed.

4.5 Summary

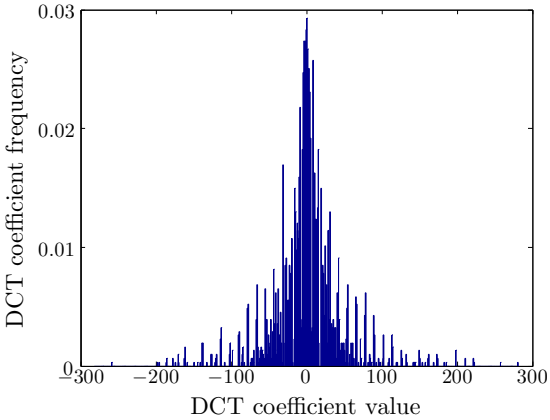
In this chapter, a new JPEG anti-forensic method is proposed. It is based on removing the spatial-domain blocking artifacts via minimizing a TV-based energy function. Experimental results show that it is able to not only defeat the blocking artifacts detectors but also fool other existing JPEG forensic detectors. As to the advanced JPEG forensic detector K_L [LB11], a

(a) $\mathcal{J}_A$ [ADF05]



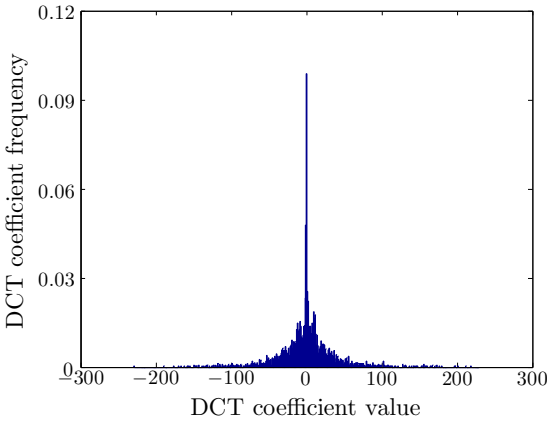
(b) $\mathcal{J}_A$ [ADF05]

(c) $\mathcal{F}_V^J$ [VTT11]



(d) $\mathcal{F}_V^J$ [VTT11]

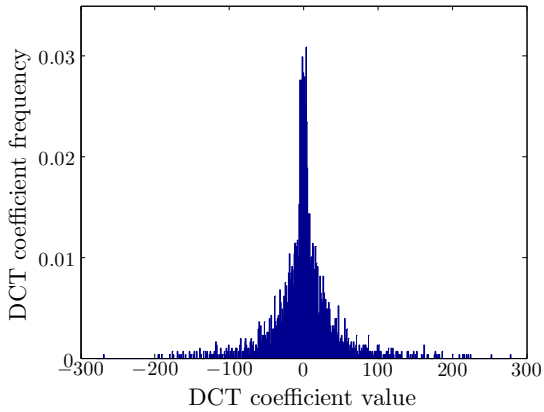
(e) $\mathcal{F}_{Su}^J$ [SS11]



(f) $\mathcal{F}_{Su}^J$ [SS11]

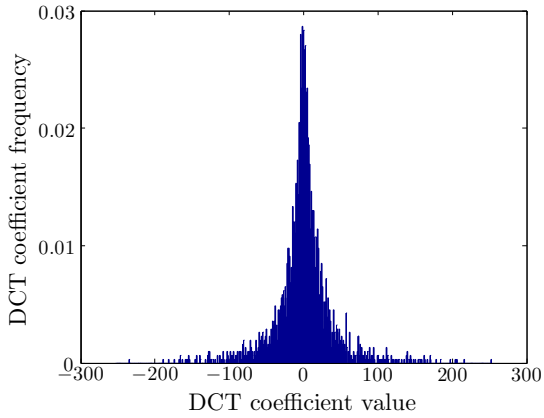
Figure 4.8: Example results (close-up images) and DCT histograms of the $(2, 2)$ subband, for $\mathcal{J}_A$ [ADF05], $\mathcal{F}_V^J$ [VTT11] and $\mathcal{F}_{Su}^J$ [SS11], respectively.

(a) $\mathcal{F}^J_{S_q}$ [Sta+10a, SL11]



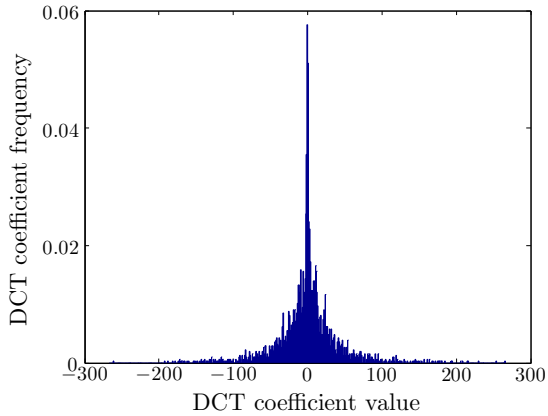
(b) $\mathcal{F}^J_{S_q}$ [Sta+10a, SL11]

(c) $\mathcal{F}^J_{S_qS_b}$ [Sta+10b, SL11]



(d) $\mathcal{F}^J_{S_qS_b}$ [Sta+10b, SL11]

(e) $\mathcal{F}^J_0$



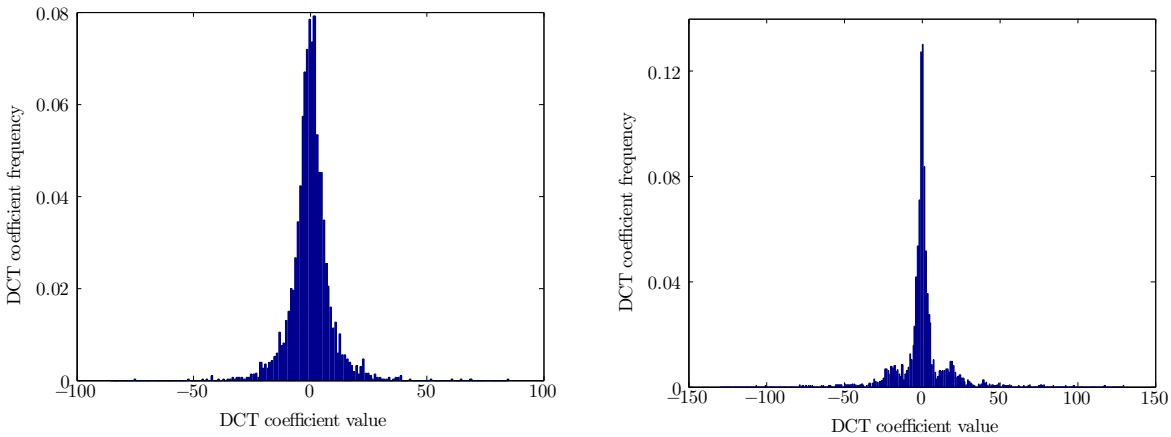
(f) $\mathcal{F}^J_0$

Figure 4.9: Example results (close-up images) and DCT histograms of the $(2, 2)$ subband, for $\mathcal{F}^J_{S_q}$ [Sta+10a, SL11], $\mathcal{F}^J_{S_qS_b}$ [Sta+10b, SL11] and $\mathcal{F}^J_0$, respectively.

calibration-based minimization problem is proposed to slightly modify the deblocked JPEG image but successfully reduce the calibrated feature of K_L into the normal range.

Though the proposed method in this chapter does not explicitly handle with the DCT-domain quantization artifacts removal task, the DCT histogram of the resulting anti-forensic image is to some extent smoothed. Experimental results show that most of the anti-forensic JPEG image $\mathcal{F}_0^J$ is able to pass off as never compressed under the examination of Fan and De Queiroz's [FD03] quantization table estimation based detector K_F^Q. Besides, Luo *et al.*'s [LHQ10] quantization step estimation based detector K_{Luo}^Q is also fooled.

Figure 4.10 compares two example DCT histograms of $\mathcal{F}_{S_qS_b}^J$ [Sta+10b, Sta+10a, SL11] and $\mathcal{F}_0^J$. We can see that the comb-like DCT-domain quantization artifacts still to some extent exist in $\mathcal{F}_0^J$. This usually happens in the mid-frequency subbands of $\mathcal{F}_0^J$. This weakness of $\mathcal{F}_0^J$ may potentially be used by advanced JPEG forensic detectors which can expose such kind of artifacts. In order to cope with the DCT-domain quantization artifacts removal after the JPEG image is processed by the proposed TV-based deblocking operation, we will propose a perceptual DCT histogram smoothing method in Chapter 5. Moreover, the SVM-based JPEG forensic detectors K_{Li}^{S100} [CS08, LLH12] and K_P^{S686} [PBF10] will also be tested.



(a) $\mathcal{F}_{S_qS_b}^J$ [Sta+10b, Sta+10a, SL11]

(b) $\mathcal{F}_0^J$

Figure 4.10: Example DCT histograms of the $(4, 4)$ subband, for $\mathcal{F}_{S_qS_b}^J$ [Sta+10a, Sta+10b, SL11] and $\mathcal{F}_0^J$, respectively.

Chapter 5

JPEG Anti-Forensics with Perceptual DCT Histogram Smoothing

Contents

5.1 Introduction and Motivation . . . 67
5.2 Proposed JPEG Anti-Forensics . . . 69
5.2.1 First-Round TV-Based Deblocking . . . 69
5.2.2 Perceptual DCT Histogram Smoothing . . . 71
5.2.2.1 Disadvantages of Global Laplacian Model . . . 71
5.2.2.2 Adaptive Local Dithering Signal Model . . . 72
5.2.2.3 DCT Histogram Mapping . . . 76
5.2.2.4 Necessity of First-Round JPEG Deblocking . . . 78
5.2.3 Second-Round TV-Based Deblocking . . . 79
5.2.3.1 Parameter Settings . . . 79
5.2.3.2 DCT Coefficient Perturbation . . . 80
5.2.4 De-Calibration . . . 80
5.3 Experimental Results of JPEG Anti-Forensics . . . 81
5.3.1 Comparing Anti-Forensic Dithering Methods . . . 81
5.3.2 Against JPEG Forensic Detectors . . . 85
5.3.3 Computation Cost . . . 87
5.4 Hiding Traces of Double JPEG Compression Artifacts . . . 88
5.4.1 Hiding Traces of Aligned Double JPEG Compression . . . 90
5.4.2 Hiding Traces of Non-Aligned Double JPEG Compression . . . 92
5.4.3 Fooling JPEG Artifacts Based Image Forgery Localization . . . 93
5.5 Summary . . . 94
5.A Appendix: The p.m.f. of the Dithering Signal Using the Laplacian Model . . . 98
5.B Appendix: The Constraints Used for Modeling the DCT Coefficients 99

This chapter proposes a JPEG anti-forensic method, which aims at removing from a given image the footprints left by JPEG compression, in both the spatial domain and the DCT domain. With reasonable loss of image quality, the proposed method can defeat existing forensic detectors that attempt to identify traces of the image JPEG compression history or JPEG anti-forensic processing. In the proposed framework, firstly because of a TV-based deblocking operation, the partly recovered DCT information is thereafter used to build an

adaptive local dithering signal model which is able to bring the DCT histogram of the processed image close to that of the original one. Then a perceptual DCT histogram smoothing is carried out by solving a simplified assignment problem, where the cost function is established as the total perceptual quality loss due to the DCT coefficient modification. The second-round deblocking and de-calibration operations successfully bring the image statistics that are used by the JPEG forensic detectors to the normal status. Experimental results show that the proposed method outperforms the state-of-the-art methods in a better tradeoff between the JPEG forensic undetectability and the visual quality of processed images. Moreover, the application of the proposed anti-forensic method in disguising double JPEG compression artifacts is proven to be feasible by experiments.

A paper describing the proposed method was published in an international journal [Fan+14]. The Matlab code of the method is freely shared online and can be downloaded from: http://www.gipsa-lab.grenoble-inp.fr/~wei.fan/documents/AFJPG-TIFS14.tar.gz.

5.1 Introduction and Motivation

In Chapter 4, a JPEG anti-forensic method based on TV-based deblocking is proposed. Though the method is initially designed to remove spatial-domain blocking artifacts from a JPEG image, experimental results show that it also perturbs the DCT coefficients to the extent that the deblocked JPEG image $\mathcal{F}_0^J$ is able to fool existing quantization artifacts detectors. However, it is very difficult to further smooth the DCT histogram of the JPEG image by simply performing deblocking in the spatial domain while keeping a high visual quality of the processed image. As shown in Figure 4.10-(b) and discussed in Section 4.5, though the comb-like gaps in the DCT histogram of the JPEG image are to some extent filled, the quantization artifacts still exist in $\mathcal{F}_0^J$ in a relatively mild way. These artifacts may not be detectable under the examination of existing quantization artifacts detectors K_F^Q [FD03], K_{Luo} [LHQ10], and K_{Luo}^Q [LHQ10]. Yet, we cannot exclude the possibility of the emergence of other JPEG forensic detectors targeting at this weakness. Therefore, in order to design reliable JPEG anti-forensics, it is necessary to fill the remaining gaps in the DCT histogram of the deblocked JPEG image processed by the TV-based deblocking method presented in Chapter 4.

Here, we adopt a similar strategy to Stamm *et al.*'s JPEG anti-forensic method [Sta+10a, Sta+10b, SL11] which handles DCT-domain quantization artifacts and spatial-domain blocking artifacts separately in different domains. Also following the above discussion, in this chapter, we propose a four-step JPEG anti-forensic method, as illustrated in Figure 5.1 and as summarized in the following:

- The first step is the TV-based deblocking in the spatial domain (to be described in Section 5.2.1). This is the same method with that described in Chapter 4, yet with different parameter settings. Besides the removal of JPEG blocking artifacts, another purpose of this step is to partly and plausibly fill gaps in the DCT histogram, so as to facilitate the following step of explicit histogram smoothing. Experimentally, it is necessary and beneficial to conduct this first-round deblocking, especially for a better histogram restoration in the high-frequency subbands where all DCT coefficients are quantized to 0 in the JPEG image (relevant results will be presented in Section 5.2.2.4). After the first step, the image generated from the JPEG image $\mathcal{J}$ is denoted as $\hat{\mathcal{F}}_b^J$.

- We found that in the deblocked image $\hat{\mathcal{F}}_b^J$, the comb-like DCT quantization artifacts are no longer as obvious as those in the JPEG image $\mathcal{J}$ (an example is shown in Figure 5.3-(c)). Under the hypothesis that the partly recovered DCT-domain information is reliable, the next step naturally goes to further filling the remaining gaps in the DCT histogram. This leads us to the construction of an adaptive local model for the DCT coefficient distribution, with which a perceptual histogram mapping method is thereafter proposed to modify the DCT coefficients while minimizing the total SSIM value loss (to be described in Section 5.2.2). We denote the intermediate image after applying the perceptual DCT histogram smoothing as $\hat{\mathcal{F}}_{bq}^J$.

- The removal of the DCT quantization artifacts is at the cost of introducing a small amount of unnatural noise and blocking artifacts in the spatial domain to the output

image $\hat{\mathcal{F}}^J_{bq}$, despite that we have tried to minimize the image quality loss. Hence, we move to the spatial domain again and conduct a second-round TV-based deblocking and regularization (to be described in Section 5.2.3). The resulting intermediate image is denoted as $\hat{\mathcal{F}}^J_{bqb}$.

- At last, $\hat{\mathcal{F}}^J_{bqb}$ is processed by the *de-calibration* operation (to be described in Section 5.2.4), which is the same with that described in Chapter 4, to generate our anti-forensic JPEG image $\mathcal{F}^J$.

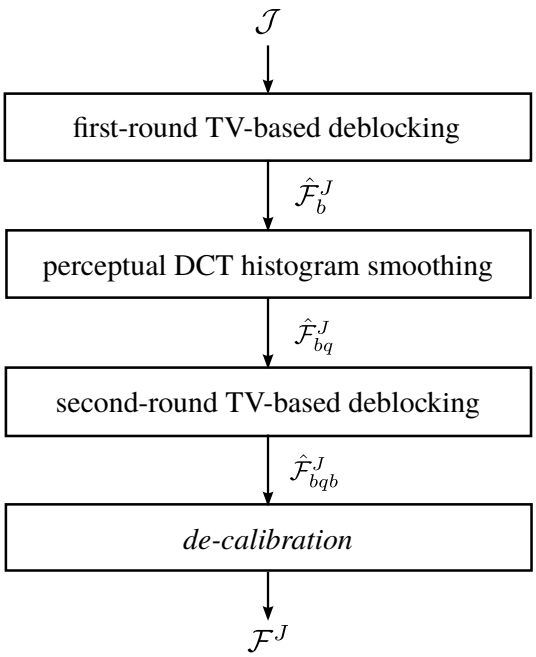


Figure 5.1: The proposed anti-forensic JPEG image creation process for $\mathcal{F}^J$.

By processing a JPEG image alternatively in the spatial domain and in the DCT domain, we hope to create an improved version of anti-forensic JPEG image which achieves better forensic undetectability than $\mathcal{F}^J_0$, which is generated by the deblocking based JPEG anti-forensic method in Chapter 4. Meanwhile, by introducing a visual quality control during the proposed DCT histogram smoothing operation, we hope a high quality of the processed image can be ensured.

As discussed in Section 1.3.1 and as illustrated in Figure 1.3, one important motivation/application of JPEG anti-forensics is to disguise double JPEG compression artifacts. In this chapter, we will also apply the proposed JPEG anti-forensic method in three scenarios of double JPEG compression, to prove the possibility of creating anti-forensic double JPEG compressed images.

The remainder of this chapter is organized as follows. The proposed JPEG anti-forensic method is described in Section 5.2. Section 5.3 shows some experimental results with some comparisons with the prior art as well as the method proposed in Chapter 4. In Section 5.4, the application of the proposed JPEG anti-forensic method is proven to be practical in creating

anti-forensic double JPEG compressed images, with experimental comparisons with the state-of-the-art methods. Finally, we summarize this chapter in Section 5.5.

5.2 Proposed JPEG Anti-Forensics

5.2.1 First-Round TV-Based Deblocking

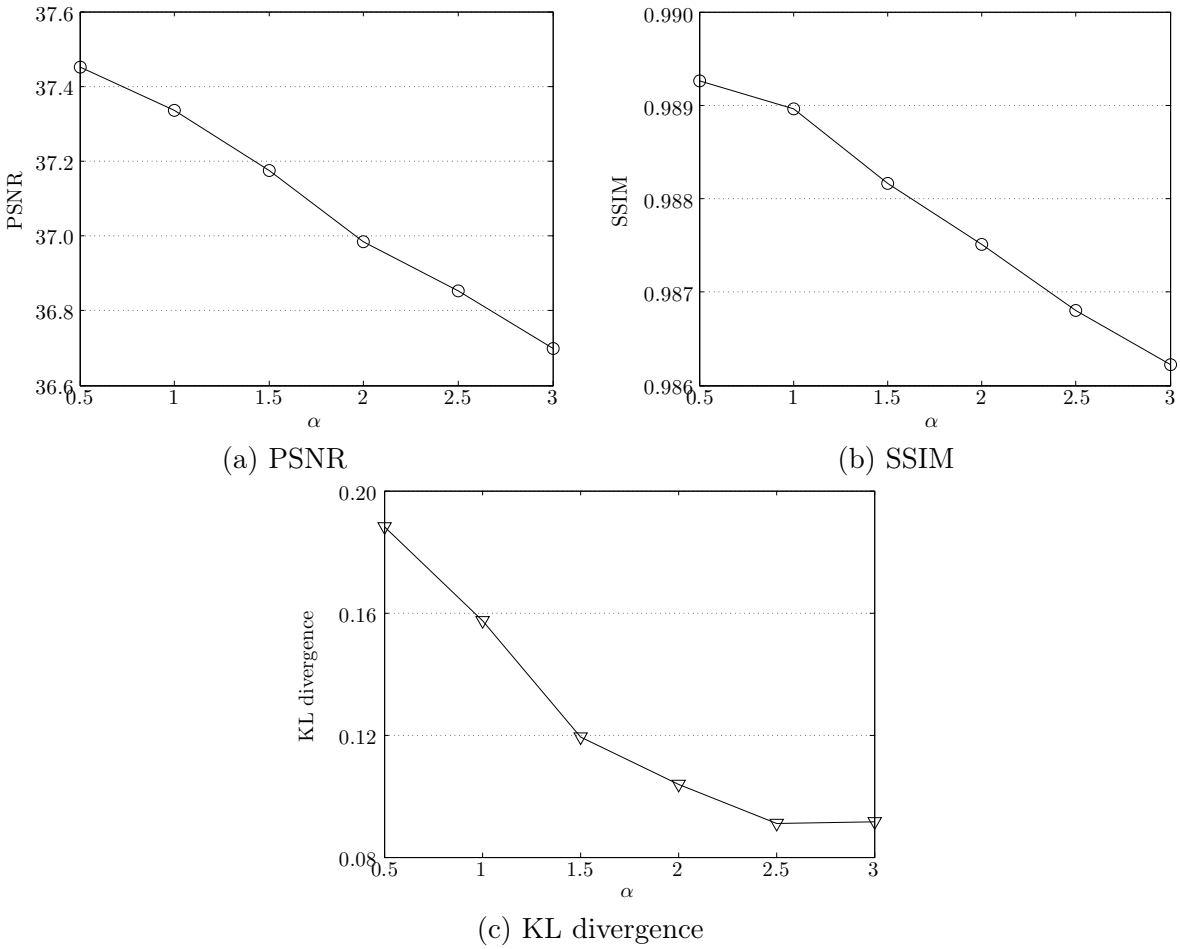
According to the flowchart of the anti-forensic JPEG image creation shown in Figure 5.1, the first step is TV-based deblocking. It is basically what is described in Section 4.3.1, yet in practice with some different parameter settings. In the proposed TV-based JPEG deblocking method, μ in Eq. (4.7), the regularization parameter α in Eq. (A.4), and the step size t_k in Eq. (4.9) are parameters that we can adjust.

For the setting of the convex set $\mathcal{S}$, we set $\mu = 1.5$ in Chapter 4. Compared with a relatively small μ value, it is for better scattering the DCT coefficients in the DCT histogram, so that the comb-like quantization artifacts are less likely to appear. Compared with a relatively big μ value, it is for the image quality consideration of the processed image, because the DCT coefficient is constrained not go too far away from its original quantization bin. The effectiveness of this setting is proven by the good forensic undetectability of $\hat{\mathcal{F}}_0^J$ against the quantization artifacts detectors K_F^Q [FD03], K_{Luo} [LHQ10] and K_{Luo}^Q [LHQ10] with a good image quality (see Table 4.2 and Figure 4.6). In practice, we found that this setting works well for low-frequency and high-frequency subbands, yet the DCT-domain quantization artifacts still exist in the mid-frequency DCT subbands (see an example in Figure 4.10-(b)).

Different from the setting in Chapter 4, here we set $\mu = 0.5$ for $\mathcal{S}$ in order to generate $\hat{\mathcal{F}}_b^J$. This setting strictly constrains the processed DCT coefficient to stay within the same quantization bin as its original value. Besides the visual quality control, this is also favorable for the perceptual DCT histogram smoothing to be described in Section 5.2.2, where the processed DCT coefficient is also constrained to stay within its original quantization bin. The projection operator $P_{\mathcal{S}}$ works as follows: once a DCT coefficient under processing goes outside the quantization bin of its original value, it will be modified back to a random value uniformly distributed within the quantization bin. In the spatial domain, the resulting pixel values will at last be rounded and truncated to integers in the range $[0, 255]$.

In order to select a good value for α in Eq. (A.4), we create a set of intermediate images $\hat{\mathcal{F}}_b^J$ on UCIDTest92 dataset (see Section 2.3.1 for more descriptions about the datasets) for comparison. The processed images $\hat{\mathcal{F}}_b^J$ are generated from the JPEG images for different values of α ranging from 0.5 to 3 with step 0.5. As the first intermediate image during our anti-forensic JPEG image creation, the output of the JPEG forensic detectors is not taken into account for comparison now. We hope $\hat{\mathcal{F}}_b^J$ to have a good image quality and to recover as much DCT-domain information as possible for the consideration of the further perceptual DCT histogram smoothing. From Figure 5.2-(a) to -(c), the average PSNR, SSIM, and KL divergence values are respectively compared for different values of α. Note that all the metric

values are computed using the original image as the reference. We can see that the image quality, as well as the KL divergence, decreases as α increases. When $\alpha > 2.5$, the curve of the average KL divergence trends to be flat. We therefore only consider $\alpha \leq 2.5$, as a high value of α will degrade too much the quality of the processed image, with limited improvement on the KL divergence. In the end, we choose $\alpha = 1.5$ as it appears to have a good tradeoff between the DCT histogram restoration quality (which will contribute to the perceptual DCT histogram smoothing described in the Section 5.2.2) and the quality of the processed image.



(a) PSNR (b) SSIM

(c) KL divergence

Figure 5.2: The quality and KL divergence change of the image $\hat{\mathcal{F}}_b^J$ obtained by the first-round TV-based deblocking for different values of α. The original image is used as the reference for computing the metric values. Results are obtained on UCIDTest92 dataset.

As to the step size, we set $t_k = 1/k$ at the k-th iteration, still following [ADF05].

Similar to the setting in Chapter 4, instead of waiting for the convergence of the optimization problem, we select the candidate deblocked image guided by the blocking signature measure K_F in Eq. (3.6). Experimentally, we run 50 iterations, and choose the resulting image giving the smallest K_F value as the final result. This provides us with a satisfying intermediate image $\hat{\mathcal{F}}_b^J$.

5.2.2 Perceptual DCT Histogram Smoothing

After JPEG image $\mathcal{J}$ has been processed using the TV-based deblocking method, the gaps in the DCT domain have been partly filled in the obtained image $\hat{\mathcal{F}}_b^J$ (an example DCT histogram is shown in Figure 5.3-(c)). We have confirmed the effectiveness of the TV-based deblocking in fooling existing JPEG forensic detectors in Chapter 4, yet with a different parameter setting of μ in Eq. (4.7). However, the periodicity of the DCT histogram in $\hat{\mathcal{F}}_b^J$ may still exist, which might be utilized by JPEG forensic detectors. This issue will be further discussed in the application of JPEG anti-forensics in Section 5.4.2. In order to achieve a better forensic undetectability, it is necessary to fill the gaps left in the DCT histogram of $\hat{\mathcal{F}}_b^J$. In this section, we propose a perceptual DCT histogram smoothing method. The partly recovered information in the DCT domain of $\hat{\mathcal{F}}_b^J$ will help us to build an adaptive local dithering signal model based on both the Laplacian distribution and the uniform distribution for a better goodness-of-fit, compared to the global Laplacian distribution.

5.2.2.1 Disadvantages of Global Laplacian Model

Constructing the DCT histogram is a common practice for studying the image statistics in the DCT domain. In this thesis, all the DCT histograms are constructed using *integers* as the bin centers. For subband (r, c) of a generic image $\mathbf{U}$, the normalized DCT histogram is therefore constructed as:

$$H_{r,c}^{\mathbf{U}}(k) = \frac{1}{L}\sum_{l=1}^{L}\delta\left(\text{round}\left((\mathbf{DU})_{r,c}^{l}\right) - k\right), \quad k \in \mathbb{Z} \tag{5.1}$$

where δ is the indicator function: $\delta(x) = 1$ if and only if $x = 0$, otherwise $\delta(x) = 0$.

For modeling the DCT coefficients in AC components, the Laplacian distribution is the dominant choice balancing the model simplicity and the fidelity to real data [LG00], which is also the basic assumption of Stamm *et al.*'s dithering-based JPEG anti-forensic method [Sta+10a, SL11]. Figure 5.3-(a) shows an example DCT histogram from a genuine, uncompressed UCID [SS04] image, and the fitting result using the discrete Laplacian distribution:

$$P(Y = y) = \begin{cases} 1 - e^{-\lambda/2} & \text{if} \quad y = 0 \\ e^{-\lambda|y|}\sinh(\lambda/2) & \text{if} \quad y \in \mathbb{Z}, y \neq 0 \\ 0 & \text{otherwise,} \end{cases} \tag{5.2}$$

where $\sinh(\cdot)$ is the hyperbolic sine function, and the parameter λ can be estimated using MLE [PR99]. From Figure 5.3-(a), we can see that the Laplacian distribution may not always be a good model for a precise description of real data.

Furthermore, as known, the kurtosis (different from the *excess kurtosis* including a -3 term) of the Laplacian distribution is a constant 6. We calculated the kurtosis of all the AC components for each UCID image [SS04], and 93.68% of them have values higher than 6. The average kurtosis is 19.99, much higher than 6, which indicates that the actual distribution of

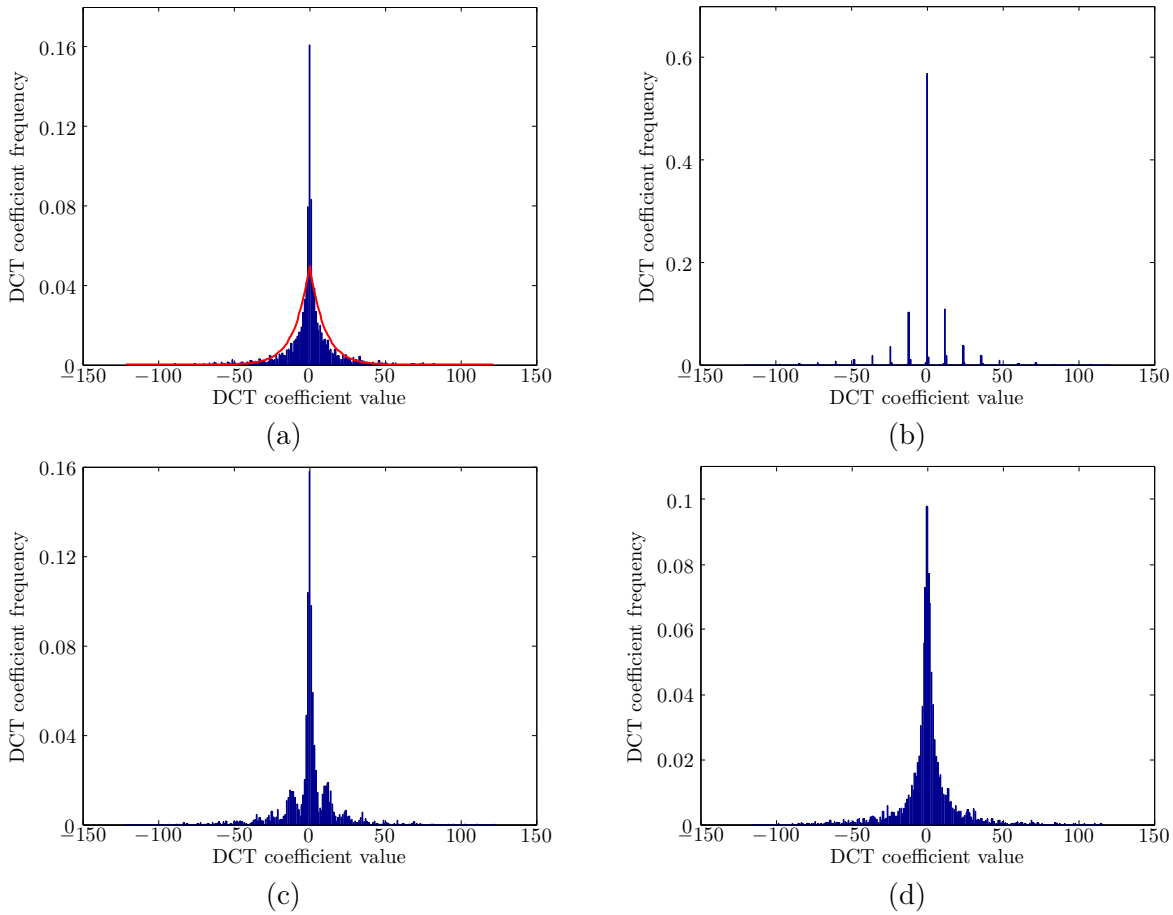


Figure 5.3: (a) is the DCT histogram of subband $(2, 2)$ from an example genuine, uncompressed UCID image [SS04], and the red curve is the fitting result using the discrete Laplacian distribution model. Then the image is JPEG compressed with quality factor 50, and the proposed JPEG anti-forensic method is applied. (b), (c), and (d) are the corresponding DCT histograms of (a) in the JPEG image, after the first-round TV-based deblocking, and after the adaptive local dithering signal is injected (as described in Section 5.2.2), respectively.

DCT coefficients usually has a much higher peak. This also partly explains the fitting problem of the DCT histogram using the Laplacian model in Figure 5.3-(a).

Moreover, Robertson and Stevenson [RS05] pointed out that for quantized DCT coefficients that are observed to be zero, *i.e.*, the DCT coefficients in the quantization bin 0 satisfying $(\mathcal{Q}(\mathbf{DU}))^l_{r,c} = 0$, the Laplacian model indeed works well. However, for the other DCT coefficients in the quantization bin $b \neq 0$ satisfying $(\mathcal{Q}(\mathbf{DU}))^l_{r,c} = b$, it appears that the uniform model fits better to the real data than the Laplacian model.

5.2.2.2 Adaptive Local Dithering Signal Model

Based on the above analysis, we hope to build a model having a better goodness-of-fit than the global Laplacian model. Comparing an example histogram shown in Figure 5.3-(c) with

that in -(b), we notice that the DCT-domain information has been partly recovered in $\hat{\mathcal{F}}_b^J$, with the help of which we will be able to build an adaptive local dithering signal model.

We still use $\mathbf{U}$ to denote a generic image pixel value matrix, however $\mathbf{U}$ now contains pixel values of the processed JPEG image $\hat{\mathcal{F}}_b^J$ using the TV-based deblocking method in Section 5.2.1. For a given DCT subband (r, c) of $\hat{\mathcal{F}}_b^J$, with $\mathbf{Q}_{r,c}$, the DCT coefficients $(\mathbf{DU})_{r,c}^l$ (generically denoted as Y') are quantized and then dequantized to obtain coefficients $(\mathcal{Q}^{-1}(\mathcal{Q}(\mathbf{DU})))_{r,c}^l$ (generically denoted as Y). The goal is to add a dithering signal N in such a way that the dithered DCT coefficient:

$$Z = Y + N \tag{5.3}$$

has no comb-like DCT quantization artifacts while its distribution is to some extent close to that of Y'. In this section, we are to find a proper distribution for the dithering signal N.

We firstly consider the AC component. Without special explanation, the DCT coefficients mentioned in this section are all from the AC component subbands. Inspired by Robertson and Stevenson's work about the suitability of the Laplacian model and the uniform model for different parts of the DCT histogram [RS05], we propose an *adaptive local* dithering signal model based on the combination of the Laplacian distribution and the uniform distribution, with the appropriate parameter tuned in each quantization bin.

We denote $B_{r,c}^- = \min\left((\mathcal{Q}(\mathbf{DU}))_{r,c}^l\right)$ as the non-empty quantization bin with the smallest bin center value, whereas $B_{r,c}^+ = \max\left((\mathcal{Q}(\mathbf{DU}))_{r,c}^l\right)$ as the non-empty quantization bin with the largest bin center value. We build the dithering signal model through one quantization bin by another, starting from quantization bin $b = 0$.

Given quantization bin b, we try to seek for parameter λ_b of the Laplacian distribution by solving the following constrained weighted least-squares fitting problem:

$$\lambda_b = \underset{\lambda_b^- \leq \lambda \leq \lambda_b^+}{\arg\min} \sum_{k=B_{r,c}^- \mathbf{Q}_{r,c} - \lfloor \frac{\mathbf{Q}_{r,c}}{2} \rfloor}^{B_{r,c}^+ \mathbf{Q}_{r,c} + \lfloor \frac{\mathbf{Q}_{r,c}}{2} \rfloor} w_k \times \left(H_{r,c}^{\mathbf{U}}(k) - P(Y = k)\right)^2, \tag{5.4}$$

where $H_{r,c}^{\mathbf{U}}$ and P are defined in Eqs. (5.1) and (5.2), respectively. The fitting problem in Eq. (5.4) means, that we wish to find a local Laplacian distribution which is still close to the corresponding distribution in $\hat{\mathcal{F}}_b^J$. We set $w_k = \left(|\text{round}(\frac{k}{\mathbf{Q}_{r,c}}) - b| + 1\right)^{-1}$ as the weight for the deduction of λ_b, which emphasizes the importance of the DCT coefficients from the current quantization bin b for the fitting. We also studied some other settings of w_k, *e.g.*, the same function as the one used above but with different powers, the Gaussian function, *etc.* We found that in practice different settings of w_k have minor impact on the histogram restoration quality, and that the current setting yields slightly better results. Moreover, λ_b^- and λ_b^+ in Eq. (5.4) are the lower and upper bounds of the parameter λ. If λ_b^- and λ_b^+ are well defined, then the fitting problem can be established and λ_b *can be found* by solving Eq. (5.4); otherwise, the fitting problem cannot be established and we say that λ_b *cannot be found*. Before we describe how the searching of the two bounds λ_b^- and λ_b^+ is performed, we first explain the models in use for each quantization bin b.

If the parameter λ_b can be found for quantization bin b by solving a well-defined fitting problem Eq. (5.4), the Laplacian model will be used. In this case, we follow Stamm *et al.*'s dithering signal model [Sta+10a]. The distribution of the dithering signal N is given by (replacing λ by the actual value of λ_b):

$$P(N=n|Y=0)=\begin{cases} c_0 e^{-\lambda|n|} & \text{if} \quad -\frac{\mathbf{Q}_{r,c}}{2}<n<\frac{\mathbf{Q}_{r,c}}{2} \\ 0 & \text{otherwise,} \end{cases} \tag{5.5}$$

$$P(N=n|Y=y,y>0)=\begin{cases} c_1 e^{-\lambda n} & \text{if} \quad -\frac{\mathbf{Q}_{r,c}}{2}\leq n<\frac{\mathbf{Q}_{r,c}}{2} \\ 0 & \text{otherwise,} \end{cases} \tag{5.6}$$

$$P(N=n|Y=y,y<0)=\begin{cases} c_1 e^{\lambda n} & \text{if} \quad -\frac{\mathbf{Q}_{r,c}}{2}<n\leq\frac{\mathbf{Q}_{r,c}}{2} \\ 0 & \text{otherwise,} \end{cases} \tag{5.7}$$

with $c_0=\frac{\lambda}{2}(1-e^{-\lambda\mathbf{Q}_{r,c}/2})^{-1}$ and $c_1=\lambda e^{-\lambda\mathbf{Q}_{r,c}/2}(1-e^{-\lambda\mathbf{Q}_{r,c}})^{-1}$. Let P_m^o and P_m^e, two functions of λ, denote the probability mass function (p.m.f.) of the *rounded* dithering signal when $\mathbf{Q}_{r,c}$ is an odd number and an even number, respectively. The domain of the p.m.f. is the integer set $\{-\lfloor\frac{\mathbf{Q}_{r,c}}{2}\rfloor,-\lfloor\frac{\mathbf{Q}_{r,c}}{2}\rfloor+1,\cdots,\lfloor\frac{\mathbf{Q}_{r,c}}{2}\rfloor\}$. For the sake of brevity, here we omit the equations of the p.m.f., which however can be found in Appendix 5.A. The p.m.f. will be used later for the searching of λ_b^- and λ_b^+ of Eq. (5.4).

As to the quantization bin b, where λ_b cannot be found, the uniform model will be used instead. In this case, the dithering signal N is generated according to:

$$P(N=n|Y=y)=\begin{cases} \frac{1}{\mathbf{Q}_{r,c}} & \text{if} \quad n\in\mathcal{N} \\ 0 & \text{otherwise,} \end{cases} \tag{5.8}$$

where $\mathcal{N}=(-\frac{\mathbf{Q}_{r,c}}{2},\frac{\mathbf{Q}_{r,c}}{2})$ when $y=0$, $\mathcal{N}=[-\frac{\mathbf{Q}_{r,c}}{2},\frac{\mathbf{Q}_{r,c}}{2})$ when $y>0$, and $\mathcal{N}=(-\frac{\mathbf{Q}_{r,c}}{2},\frac{\mathbf{Q}_{r,c}}{2}]$ when $y<0$.

Now, we go back to the searching of the bounds λ_b^- and λ_b^+ used in Eq. (5.4). We start from the center of the DCT histogram, *i.e.*, quantization bin $b=0$. We set $\lambda_b^-=10^{-3}$ according to Valenzise *et al.*'s statement [VTT11] that the parameter λ of the Laplacian model usually takes values between 10^{-3} and 1 for natural images. The constraint used in the searching is based on the observation that in the distribution of DCT coefficients, the probability decreases when the coefficient magnitude increases. As an example and as illustrated in Figure 5.4, when $\mathbf{Q}_{r,c}$ is an odd number, we constrain that the probability of DCT coefficient falling in the leftmost integer bin $k=-\lfloor\frac{\mathbf{Q}_{r,c}}{2}\rfloor$ (or the rightmost integer bin $k=\lfloor\frac{\mathbf{Q}_{r,c}}{2}\rfloor$) of the quantization bin $b=0$ should be no smaller than either that in the rightmost integer bin of the quantization bin $b=-1$ or that in the leftmost integer bin of the quantization bin $b=1$. For the moment, in the neighboring quantization bins -1 and 1, the DCT coefficients are assumed to follow a uniform distribution. Then λ_b^+ is determined by solving Eq. (5.9), where M_0,M_{-1},M_1 are respectively the approximate probabilities of DCT coefficient falling in quantization bins $0,-1,1$, which are estimated directly from the constructed histogram. The searching for λ_b^+ can be done using a numerical method. A set of numbers are uniformly sampled from

the interval $[10^{-3}, 1]$. Given each number in this set as the parameter λ, P_m^o is calculated. Therefore, λ_b^+ is chosen as the largest number satisfying constraints in Eq. (5.9). When $\mathbf{Q}_{r,c}$ is an even number, the procedure to find λ_b^+ is similar yet slightly different. Detailed information can be found in Appendix 5.B.

$$\lambda_b^+ = \underset{10^{-3} \leq \lambda \leq 1}{\arg\max}\ \lambda, \text{ subject to: } P_m^o\left(N = \lfloor \frac{\mathbf{Q}_{r,c}}{2} \rfloor | Y = 0\right) \times M_0 \geq \max\left(\frac{M_{-1}}{\mathbf{Q}_{r,c}}, \frac{M_1}{\mathbf{Q}_{r,c}}\right) \quad (5.9)$$

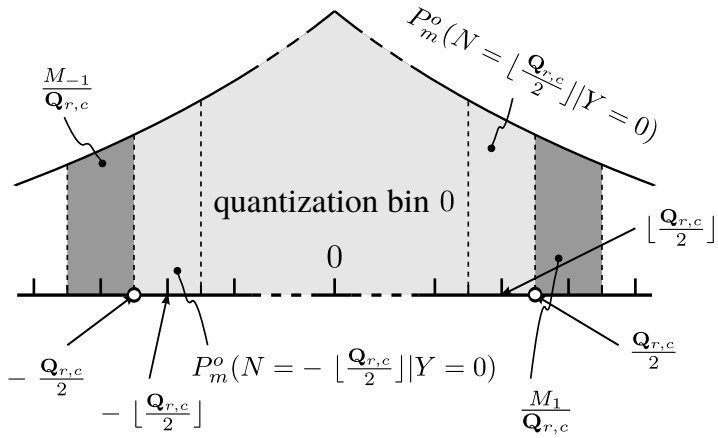


Figure 5.4: Illustration for the constraint used for the searching of λ_b^+ when $\mathbf{Q}_{r,c}$ is an odd number: The probability for the leftmost (or the rightmost) integer bin of the quantization bin $b = 0$ should be no smaller than either that of the rightmost integer bin in the quantization bin $b = -1$ or that of the leftmost integer bin in the quantization bin $b = 1$.

For quantization bins $b = 1, 2, \cdots, B_{r,c}^+$, we adopt a similar procedure to obtain λ_b^- and λ_b^+. The constraint is that the probability of coefficients falling in the leftmost integer bin of quantization bin $b \geq 1$ should be no bigger than that in the rightmost integer bin of quantization bin $b - 1$, meanwhile the probability of coefficients falling in the rightmost integer bin of quantization bin b should be no smaller than that in the leftmost integer bin of quantization bin $b + 1$. The distribution in the quantization bin $b - 1$ is already defined at that moment, and for the quantization bin $b + 1$ we assume the uniform distribution. Then λ_b^- and λ_b^+ are chosen as the smallest and largest number in $[10^{-3}, \lambda_{b-1}]$ satisfying the constraint, respectively. However, if λ_b^- and λ_b^+ cannot be found, the uniform model in Eq. (5.8) will be used as the dithering signal model for the current and following quantization bin(s). For quantization bins $b = -1, -2, \cdots, B_{r,c}^-$, a similar searching procedure for the values of λ_b^- and λ_b^+ is applied (see Appendix 5.B for details).

When we have well defined λ_b^- and λ_b^+ values, the parameter λ_b of the Laplacian model is obtained by solving the minimization problem in Eq. (5.4), and the dithering signal N can be thereafter generated according to Eqs. (5.5)-(5.7); otherwise, the uniform model in Eq. (5.8) is used. Algorithm 5.1 summarizes the proposed adaptive local dithering procedure for the AC components.

Algorithm 5.1 Adaptive local dithering procedure for AC components.

1: **require:** Y
2: **initialization:** $Z = Y$
3: **for** $b = 0, 1, 2, \cdots, B^{+}_{r,c}, -1, -2, \ldots, B^{-}_{r,c}$ **do**
4: Search for λ^{+}_{b} and λ^{-}_{b} (*e.g.*, using Eq. (5.9))
5: **if** λ^{+}_{b} and λ^{-}_{b} are well defined **then**
6: $\lambda_b \leftarrow$ Eq. (5.4)
7: Generate N using Eq. (5.5)/(5.6)/(5.7)
8: $Z \leftarrow Z + N$
9: **else**
10: Generate N using Eq. (5.8)
11: $Z \leftarrow Z + N$
12: **end if**
13: **end for**
14: **return** Z

Although we adopt Stamm *et al.*'s [Sta+10a, SL11] dithering signal model, note that their model is rather *global* and our model is *local*. In Stamm *et al.*'s model, once the parameter λ of the Laplacian distribution is estimated for a subband, it will be used for all the quantization bins of this subband. However, in our model, it is considered *locally* for each quantization bin b. We tune parameter λ_b of the Laplacian model for different quantization bins if it can be found; otherwise, the uniform model will be used instead. Moreover, in Stamm *et al.*'s method, the MLE [PR99] is used to estimate λ from the JPEG image. However, in the proposed method, λ_b for each quantization bin is obtained via a weighted least-squares fitting using the post-processed image $\hat{\mathcal{F}}^{J}_{b}$ where the DCT-domain information is partly recovered. We will show in Section 5.3.1 that our method leads to a better restoration of the DCT histogram of the original, uncompressed image.

Because there is no general model for the DC component, we use the uniform model for all the quantization bins to generate the dithering signal N according to Eq. (5.8).

For each quantization bin, the dithering signal generated by numerical sampling of a given probability distribution function can reproduce the natural, fine-grained details in the DCT histogram. An example result is shown in Figure 5.3-(d).

5.2.2.3 DCT Histogram Mapping

Now, we can generate the dithered signal Z using the adaptive local dithering signal model. However, we cannot use Z directly as the altered DCT coefficients by adding the dithering signal N *randomly* to Y, without any consideration of the image spatial-domain information: the processed image will suffer from low visual quality as Stamm *et al.*'s anti-forensic JPEG image does [Sta+10a, SL11]. A different strategy is adopted here. We will try to move the distribution of Y' (*i.e.*, the distribution of DCT coefficients of $\hat{\mathcal{F}}^{J}_{b}$) towards that of Z, while minimizing the introduced distortion in the spatial domain, by solving an assignment problem

whose cost function is defined as the total perceptual quality loss.

We still consider the DCT coefficients in each quantization bin *individually*. In this section, all the DCT coefficients mentioned are in a single quantization bin b of subband (r, c). A classical assignment problem can be established as follows. Let O^b denote the set of DCT coefficients in quantization bin b from Y' which are to be modified, and T^b is used to denote the set of target DCT coefficient values from Z $(= Y + N)$ falling in the quantization bin b. O^b and T^b are of equal size. The weight function $W : O^b \times T^b \to \mathbb{R}$ is defined as the SSIM value loss due to the coefficient modification, compared with the currently achieved processed image from the last solved assignment problem (or $\hat{\mathcal{F}}_b^J$ at the very beginning). Our goal is to find a bijection $f : O^b \to T^b$ such that the cost function:

$$\sum_{o \in O^b} W(o, f(o)), \tag{5.10}$$

is minimized. This problem can be solved using the well-known Hungarian algorithm[13] [Kuh55] (see Section 2.4.2 for more descriptions of this algorithm). The solution of the assignment problem can be found in $O(D^3)$ time, where D is the dimensionality of the problem. If D is small, the problem however can be solved within a reasonable time. The setting of D will be further discussed in Section 5.3.1.

In order to save the computation cost, we hereby propose three strategies to simplify the building and the solving of this assignment problem. Firstly, not all but only part of the DCT coefficients in Y' are to be modified, so that the dimensionality of the assignment problem can be largely reduced. For choosing the DCT coefficients to put into O^b, first we compare the *unnormalized* DCT histogram (using *integers* as bin centers) of Y', denoted as h_o^b, and that of Z, denoted as h_t^b. Integers $b\mathbf{Q}_{r,c} - \lfloor \frac{\mathbf{Q}_{r,c}}{2} \rfloor, b\mathbf{Q}_{r,c} - \lfloor \frac{\mathbf{Q}_{r,c}}{2} \rfloor + 1, \cdots, b\mathbf{Q}_{r,c} + \lfloor \frac{\mathbf{Q}_{r,c}}{2} \rfloor$ are all the possible rounded DCT coefficient values in h_o^b and h_t^b. It is obvious that the dithering process in Section 5.2.2.2 ensures that the two histograms in comparison have the same number of DCT coefficients. We compute their difference histogram as:

$$h_d^b(k) = h_o^b(k) - h_t^b(k), k = b\mathbf{Q}_{r,c} - \lfloor \frac{\mathbf{Q}_{r,c}}{2} \rfloor, \cdots, b\mathbf{Q}_{r,c} + \lfloor \frac{\mathbf{Q}_{r,c}}{2} \rfloor. \tag{5.11}$$

In order to reduce the dimensionality of the assignment problem, the first strategy is that we do not modify the coefficients in h_o^b that are already in the *correct* integer bin with respect to h_t^b. More precisely, for each integer bin k with $h_d^b(k) \leq 0$, the corresponding DCT coefficients are left unchanged; and for each integer bin k with $h_d^b(k) > 0$, instead of putting all the $h_o^b(k)$ coefficients in O^b, we only put $h_d^b(k)$ $(< h_o^b(k))$ coefficients into O^b while leaving the other $h_t^b(k)$ coefficients unchanged. In general, this will yield suboptimal results, however the advantage is that it will largely reduce the computation cost with still satisfactory final results. In order to control the distortion in the spatial domain, it is better to choose the DCT coefficient whose corresponding spatial-domain 8×8 block is less sensitive to noise for being altered. Using a similar strategy with [BFT12], here we adopt the SSIM index [Wan+04] to compute

[13] We use the Matlab code downloaded from: http://www.mathworks.fr/matlabcentral/fileexchange/20652-hungarian-algorithm-for-linear-assignment-problems-v2-3.

a similarity map between the currently achieved image and $\hat{\mathcal{F}}_b^J$. The DCT coefficients whose spatial-domain 8×8 block has a higher SSIM index value are chosen to be modified and put into set O^b. Note that at the very beginning of the procedure, the SSIM index cannot be computed because the currently achieved image is exactly $\hat{\mathcal{F}}_b^J$. In this case, we calculate the local variance instead. We also have to reduce the dimensionality of the target value set T^b so that O^b and T^b are of equal size. To this end, for each k with $h_d^b(k) < 0$, in Z we *randomly* choose $-h_d^b(k)$ (> 0) values among the DCT coefficients whose rounded values are k, and put these values into T^b.

The second strategy to simplify the building of the assignment problem in Eq. (5.10) is to speed up the calculation of the weight function W. Normally, for each DCT coefficient o in O^b, all the possible modifications to values in T^b should be enforced to calculate the cost. Yet, this can be simplified by only computing the SSIM value loss when the DCT coefficient o in O^b is changed to a value in T^b which is the farthest from o. The linear interpolation is afterwards used to estimate the cost for all the other possible modifications in T^b.

The last strategy is to randomly split the simplified assignment problem into several smaller ones of lower dimensionality which can be solved in a reasonable time. This will be further discussed with experimental results in Section 5.3.1, where we also show that the adoption of all these strategies will not impair the visual quality of the obtained image, despite the fact that these strategies lead to suboptimal solutions.

After solving a simplified assignment problem for each quantization bin, we are able to smooth the DCT histogram with minimum introduced distortion in the spatial domain. The created intermediate image is denoted as $\hat{\mathcal{F}}_{bq}^J$.

5.2.2.4 Necessity of First-Round JPEG Deblocking

As one may have noticed, it is possible to perform the perceptual DCT histogram smoothing described in this section directly on the JPEG image, without the application of the first-round TV-based deblocking described in Section 5.2.1. Here, $\hat{\mathcal{F}}_q^J$ denotes the image obtained using the proposed perceptual DCT histogram smoothing directly from the JPEG image $\mathcal{J}$. For comparing $\hat{\mathcal{F}}_{bq}^J$ and $\hat{\mathcal{F}}_q^J$, we conduct a test on UCIDTest92 dataset.

PSNR, SSIM are adopted as the image quality evaluation metrics (see Section 2.2.2 for more descriptions). In order to give a quantitative evaluation of the DCT histogram, we use the KL divergence (see Section 2.2.3 for more descriptions) as the difference measure between the histogram of the uncompressed image and the one of the processed image. A smaller value of KL divergence means a better resemblance between the two compared histograms. Table 5.1 reports the average PSNR, SSIM, and KL divergence values of $\hat{\mathcal{F}}_{bq}^J$ and $\hat{\mathcal{F}}_q^J$. All the metric values are computed using the original uncompressed image $\mathcal{I}$ as the reference. The difference between the two kinds of KL divergences is that the first one is averaged over the subbands where not all the DCT coefficients are quantized to 0 in the original JPEG image, whereas the other is averaged over the rest of the subbands. The lower KL divergence value of $\hat{\mathcal{F}}_{bq}^J$ than

that of $\hat{\mathcal{F}}_q^J$ demonstrates that with the partly recovered DCT-domain information in $\hat{\mathcal{F}}_b^J$ we are able to achieve a more accurate DCT histogram restoration, especially for the subbands where all the DCT coefficients are quantized to 0 in JPEG images.

Table 5.1: Image quality and KL divergence (with $\mathcal{I}$ as the reference) comparison after the perceptual DCT histogram smoothing, with and without the first-round TV-based deblocking. Results are obtained on UCIDTest92 dataset.

	PSNR	SSIM	KL divergence-1	KL divergence-2
$\hat{\mathcal{F}}_{bq}^J$	36.4419	0.9862	0.0904	0.1054
$\hat{\mathcal{F}}_q^J$	36.1427	0.9874	0.1316	0.2622

Moreover, $\hat{\mathcal{F}}_b^J$ also helps us to reduce the dimensionality of the simplified assignment problem during the DCT histogram mapping described in Section 5.2.2.3. The reason is that in the first strategy of simplifying the assignment problem, more DCT coefficients in $\hat{\mathcal{F}}_b^J$ will already be in the correct integer bin than those in the JPEG image.

Furthermore, $\hat{\mathcal{F}}_{bq}^J$ achieves a slightly higher PSNR value, but slightly lower SSIM value, than $\hat{\mathcal{F}}_q^J$. Considering the results of both metrics, the two kinds of images have comparable visual qualities. Another advantage of the TV-based deblocking is the removal of JPEG blocking artifacts. It is therefore necessary to conduct the first-round TV-based deblocking for a better tradeoff between the visual quality and the histogram restoration quality of the processed image.

5.2.3 Second-Round TV-Based Deblocking

In the perceptual DCT histogram smoothing, although we have tried to modify the DCT coefficients while minimizing the spatial-domain distortion, there must be some unnatural noise and blocking artifacts introduced in $\hat{\mathcal{F}}_{bq}^J$. Hence, we focus on the spatial domain again and propose to apply the second-round TV-based deblocking and regularization.

5.2.3.1 Parameter Settings

Similar to Section 5.2.1, the deblocking procedure is basically the same as that described in Section 4.3.1, yet with some modifications to the parameter setting. Since the JPEG blocking artifacts presented in $\hat{\mathcal{F}}_{bq}^J$ are not as serious as those in $\mathcal{J}$, hence we lower the parameters α (see Eq. (A.4)) and t_k (see Eq. (4.9)) for a milder JPEG deblocking. We set $\alpha = 0.9$, and the step size $t_k = 1/(k+1)$ at the k-th iteration. As to the setting of the convex set $\mathcal{S}$, here we set $\mu = 1.5$ (see Eq. (4.7)), which constrains that the processed DCT coefficient should stay within the same or the neighboring quantization bins as its original value. Once a processed DCT coefficient goes outside of the constrained range, the projection operator $P_{\mathcal{S}}$ modifies its value back to a random value uniformly distributed in the original quantization bin. This can avoid strong DCT histogram shape modification by the TV-based deblocking and prevent

the emergence of new DCT quantization artifacts. Using these empirical parameter settings, experimentally we can achieve satisfactory results considering both the forensic undetectability and the visual quality of the processed image.

5.2.3.2 DCT Coefficient Perturbation

We also observed that in practice, the TV-based deblocking might interfere with the output of the quantization table estimation based detector K_F^Q [FD03], especially in the high-frequency subbands. The image tends to be over-smoothed by the TV-based regularization. As analyzed in Section 4.2.1, the quantization table estimation based detector will detect one DCT subband as quantized by 3 instead of 1, when the probability of DCT coefficients whose rounded values are integer multiples of 3 (denoted as p_0) is higher than 67.28%. This frequently happens in the high-frequency subbands of relatively smooth images, which has a high frequency of DCT coefficients with rounded value 0.

In order to tackle this problem, we propose to introduce a slight perturbation to the DCT coefficients in the high-frequency subbands which have a high value of p_0 during each optimization iteration. Considering the influence of the subgradient method, the DCT coefficients projection of $P_{\mathcal{S}}$, and the pixel value rounding and truncation in the spatial domain, for a given DCT subband, if $p_0 > 60\%$, we modify part of the DCT coefficients whose rounded values are 0 to the integer bins $-5, -4, \cdots, -1, 1, 2, \cdots, 5$. The DCT coefficient whose corresponding spatial-domain block has a higher tolerance of distortion (measured by the SSIM index comparing the currently achieved image and the deblocked image in the last subgradient iteration, or $\hat{\mathcal{F}}_{bq}^J$ at the very beginning) will be modified first to a DCT coefficient which has a bigger rounded amplitude. The modification constrains that the relative probability of integer bins $-5, -4, \cdots, -1, 1, 2, \cdots, 5$ should stay unchanged. The modification stops once p_0 reaches 50%.

In order to avoid over-smoothing of the image, a random threshold for each image is drawn from the distribution of the K_F values for genuine, uncompressed images, and the iteration stops once the K_F value in Eq. (3.6) decreases below it. Otherwise, if this cannot be achieved within 30 iterations, the resulting intermediate image with the smallest K_F output is chosen as the final result. The created image after this step is denoted as $\hat{\mathcal{F}}_{bqb}^J$.

5.2.4 De-Calibration

For $\hat{\mathcal{F}}_{bqb}^J$, all the existing detectors seems to be well fooled except the calibrated feature based detector K_L [LB11]. Based on the same consideration presented in Section 4.3.2, we perform the de-calibration operation to the intermediate image $\hat{\mathcal{F}}_{bqb}^J$. The method is the same as that described in Section 4.3.2. In order to avoid tedious repetition, here we refrain from re-presenting it. After de-calibration, the anti-forensic JPEG image $\mathcal{F}^J$ is obtained.

5.3 Experimental Results of JPEG Anti-Forensics

Besides the state-of-the-art anti-forensic JPEG images listed in Table 3.2 and $\mathcal{F}_0^J$ created in Chapter 4, our improved anti-forensic JPEG image and two kinds of intermediate results are created from the JPEG image $\mathcal{J}$:

- $\mathcal{F}^J$, with the application of the proposed four-step JPEG anti-forensic method;
- $\hat{\mathcal{F}}_{bq}^J$, with the application of the proposed first-round TV-based deblocking and the perceptual DCT histogram smoothing method, the maximum dimensionality is set to be 200 when solving the assignment problem;
- $\hat{\mathcal{F}}_{bq}'$, the creation of which is the same as $\hat{\mathcal{F}}_{bq}^J$, except that there is no limit set for the maximum dimensionality when solving the assignment problem.

5.3.1 Comparing Anti-Forensic Dithering Methods

Figure 5.5 shows the SSIM value (with the original image $\mathcal{I}$ as the reference) comparison when tested on the "Lena" image. Note that the difference between the results shown here and those shown in [VTT11] is probably due to the different versions of "Lena" images in use and to the different parameter settings of SSIM index. However, the trend of the curves is in accordance with each other, and the results confirm the conclusion of [VTT11] that it is not easy to conceal the traces of JPEG compression without serious image quality loss. According to Figure 5.5, the perceptual anti-forensic dither of Valenzise *et al.* [VTT11] is able to achieve a higher SSIM value of the processed image than Stamm *et al.*'s anti-forensic dither [Sta+10a, SL11]. However, the proposed DCT histogram smoothing method outperforms both of them [Sta+10a, SL11, VTT11] in terms of SSIM value. Figure 5.7 shows the close-up images of Lena. In the relatively smooth area of the image, *e.g.*, the shoulder and the face of Lena, we can see that less noise is introduced in the spatial domain of $\hat{\mathcal{F}}_{bq}^J$ compared with $\mathcal{F}_{S_q}^J$ and $\mathcal{F}_V^J$.

The average PSNR and SSIM values of large-scale test on UCIDTest dataset for $\hat{\mathcal{F}}_{bq}^J$ are 35.9926 dB and 0.9872, respectively, which are noticeably higher than those of $\mathcal{F}_{S_q}^J$ and of $\mathcal{F}_V^J$ (see Table 5.4). It demonstrates that the proposed perceptual DCT histogram smoothing can help us to achieve a higher image quality of the processed image than the state-of-the-art anti-forensic dithering methods [Sta+10a, SL11, VTT11].

In the proposed DCT histogram smoothing method, we have to solve an assignment problem for each quantization bin of each subband. As mentioned in Section 5.2.2.3, the computation cost is $O(D^3)$ when using the Hungarian algorithm [Kuh55], where D is the dimensionality of the assignment problem. Obviously, the problem solving might be impractical when D is too large. A possible solution is that we randomly split a single assignment problem into several smaller ones. From Figure 5.5, we can see that the problem splitting barely affects the image quality of the processed image. However, it largely saves the computation cost to

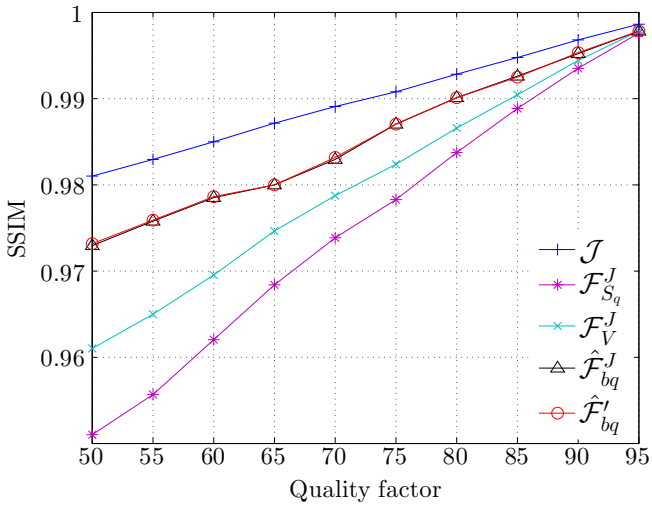


Figure 5.5: SSIM values achieved by $\mathcal{J}$, $\mathcal{F}^J_{S_q}$ [Sta+10a, SL11], $\mathcal{F}^J_V$ [VTT11], $\hat{\mathcal{F}}^J_{bq}$, and $\hat{\mathcal{F}}'_{bq}$, with $\mathcal{I}$ as the reference, when tested on the "Lena" image. The $\hat{\mathcal{F}}^J_{bq}$ plot is almost under that of $\hat{\mathcal{F}}'_{bq}$.

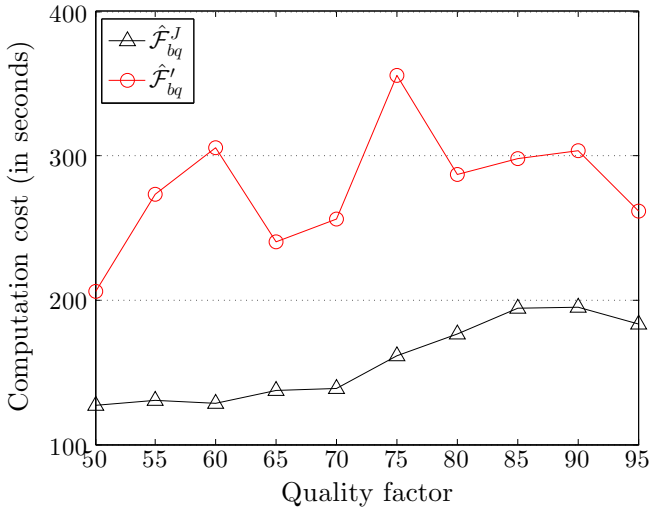
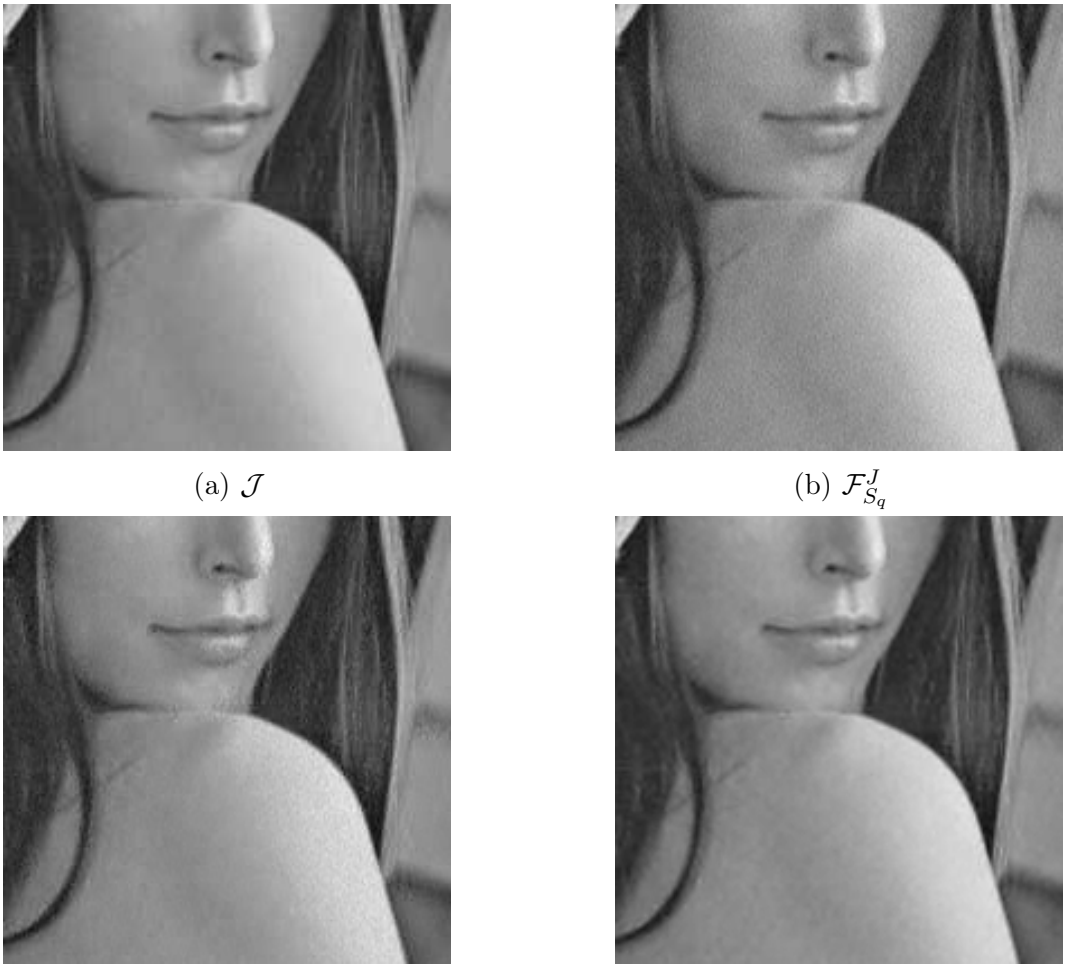


Figure 5.6: Time taken to create $\hat{\mathcal{F}}^J_{bq}$, and $\hat{\mathcal{F}}'_{bq}$ from the JPEG compressed "Lena" image with different quality factors. By reducing the dimensionality of the assignment problem, the computation cost is largely reduced to conduct the perceptual DCT histogram smoothing.

conduct the perceptual DCT histogram smoothing procedure, as demonstrated by the test on the "Lena" image, as reported in Figure 5.6. In the large-scale test, in order to speed up the simulation, we always randomly split the assignment problem into several smaller ones of maximum dimensionality of 200.

With experiments carried out on UCIDTest dataset, we compute the KL divergence value between $\mathcal{I}$ and $\mathcal{F}^J_{S_q}$, and that between $\mathcal{I}$ and $\hat{\mathcal{F}}^J_{bq}$ for all DCT subbands. The *difference* between these two KL divergence values is reported in Table 5.2. We notice that during the dithering process, the subband where all the coefficients are quantized to 0 is left untouched in [Sta+10a,



(a) $\mathcal{J}$ (b) $\mathcal{F}^J_{S_q}$

(c) $\mathcal{F}^J_V$ (d) $\hat{\mathcal{F}}^J_{bq}$

Figure 5.7: Example results (close-up images) of $\hat{\mathcal{F}}^J_{bq}$ compared with $\mathcal{J}$, $\mathcal{F}^J_{S_q}$ [Sta+10a, SL11], and $\mathcal{F}^J_V$ [VTT11], where $\mathcal{J}$ is compressed with quality factor 50. Their SSIM values (with $\mathcal{I}$ as the reference) are: (a) 0.9809, (b) 0.9509, (c) 0.9610, and (d) 0.9731. We can see that less noise is introduced in $\hat{\mathcal{F}}^J_{bq}$ especially in the relatively smooth area of the image. Results are obtained on the "Lena" image.

SL11]. To be fair for $\mathcal{F}^J_{S_q}$, these subbands were not counted in the comparison. From Table 5.2, we can see that $\hat{\mathcal{F}}^J_{bq}$ performs consistently better than $\mathcal{F}^J_{S_q}$. Similar results are obtained when compared with Valenzise *et al.*'s anti-forensic JPEG images $\mathcal{F}^J_V$ [VTT11], because the dithering model used in [VTT11] is exactly the same as in [Sta+10a] in the DCT domain. We can conclude that the proposed adaptive local dithering model has a better restoration capability of the original DCT histogram than the one based on the global Laplacian model [Sta+10a, VTT11].

We also compare the KL divergence value between $\mathcal{I}$ and $\mathcal{F}^J_0$ obtained using the anti-forensic method presented in Chapter 4, and that between $\mathcal{I}$ and $\mathcal{F}^J$ for all DCT subbands. The *difference* between these two KL divergence values is reported in Table 5.3. In this

Table 5.2: The difference of the KL divergence between $\mathcal{I}$ and $\mathcal{F}^J_{S_q}$ [Sta+10a, SL11], and that between $\mathcal{I}$ and $\hat{\mathcal{F}}^J_{bq}$ for all 64 DCT subbands. The average difference value over all subbands is 0.0552 with standard deviation 0.0249. For a fair comparison for $\mathcal{F}^J_{S_q}$, the subbands, whose DCT coefficients are all quantized to 0 in the JPEG image $\mathcal{J}$, are not counted. Results are obtained on UCIDTest dataset.

r \ c	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
1	0.0001	0.0065	0.0118	0.0278	0.0493	0.0634	0.0663	0.0656
2	0.0042	0.0166	0.0229	0.0363	0.0447	0.0565	0.0504	0.0369
3	0.0161	0.0208	0.0291	0.0442	0.0573	0.0665	0.0634	0.0497
4	0.0200	0.0317	0.0409	0.0470	0.0658	0.0802	0.0553	0.0446
5	0.0357	0.0395	0.0522	0.0678	0.0764	0.0930	0.0927	0.0856
6	0.0441	0.0383	0.0642	0.0610	0.0726	0.0769	0.0806	0.0957
7	0.0538	0.0442	0.0678	0.0595	0.0879	0.0809	0.0948	0.0975
8	0.0619	0.0545	0.0697	0.0528	0.0927	0.0880	0.0854	0.0722

comparison, all the DCT subbands of all the images in UCIDTest dataset are considered, no matter whether there are some DCT subbands whose coefficients are all quantized to 0 in the corresponding JPEG image $\mathcal{J}$. We can see that the JPEG anti-forensic method proposed in this chapter has improved the DCT histogram recovery performance compared to the method presented in Chapter 4. We can conclude that the perceptual DCT histogram smoothing method proposed in Section 5.2.2 is able to better shape the DCT histogram towards that of the original image.

Table 5.3: The difference of the KL divergence between $\mathcal{I}$ and $\mathcal{F}^J_0$, and that between $\mathcal{I}$ and $\mathcal{F}^J$ for all 64 DCT subbands. The average difference value over all subbands is 0.0160 with standard deviation 0.0139. All the subbands are considered, no matter whether the DCT coefficients are all quantized to 0 in the JPEG image $\mathcal{J}$. Results are obtained on UCIDTest dataset.

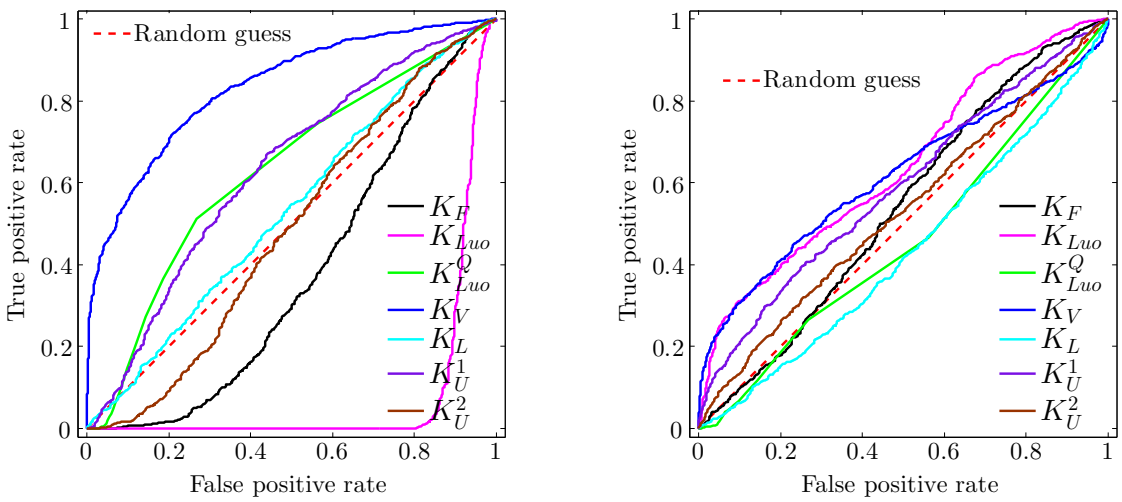
r \ c	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
1	-0.0040	0.0002	0.0047	0.0035	0.0072	0.0135	0.0243	0.0464
2	0.0045	-0.0043	-0.0002	0.0053	0.0127	0.0275	0.0248	0.0451
3	0.0023	-0.0033	0.0037	0.0106	0.0136	-0.0042	0.0001	0.0292
4	0.0060	0.0013	0.0094	0.0132	0.0258	0.0186	0.0288	0.0395
5	0.0035	0.0088	0.0113	0.0254	0.0121	0.0071	0.0128	0.0278
6	0.0109	0.0202	-0.0065	0.0193	0.0138	0.0139	0.0143	0.0291
7	0.0287	0.0298	-0.0036	0.0231	0.0132	0.0151	0.0130	0.0297
8	0.0435	0.0487	0.0243	0.0408	0.0287	0.0281	0.0294	0.0035

5.3.2 Against JPEG Forensic Detectors

Table 5.4 reports the AUC values achieved by different kinds of (anti-forensic) JPEG images against 7 scalar-based forensic detectors listed in Table 3.1. Besides, the last two columns of Table 5.4 also list the average PSNR and SSIM values for the comparison of the visual quality of processed image with the original uncompressed image $\mathcal{I}$ as the reference. Moreover, Figure 5.8 plots the ROC curves obtained by Stamm *et al.*'s [Sta+10a, Sta+10b, SL11] anti-forensic JPEG image $\mathcal{F}^J_{S_qS_b}$ and ours $\mathcal{F}^J$.

Table 5.4: From the 2nd to the 8th columns, the AUC values for different kinds of images when tested against different scalar-based JPEG forensic detectors are listed; the image quality (with $\mathcal{I}$ as the reference) comparison is reported in the last 2 columns. Results are obtained on UCIDTest dataset.

	K_F	K_{Luo}	K^Q_{Luo}	K_V	K_L	K^1_U	K^2_U	PSNR	SSIM
$\mathcal{J}$	0.9991	1.0000	0.9996	0.9976	0.9811	0.9860	0.8840	37.0999	0.9919
$\mathcal{F}^J_{S_q}$	0.9332	0.4328	0.7328	0.9977	0.9946	0.9633	0.9483	33.4061	0.9756
$\mathcal{F}^J_{S_qS_b}$	0.3783	0.0806	0.6288	0.8337	0.5338	0.6309	0.4854	30.4591	0.9509
$\mathcal{F}^J_V$	0.9889	0.4330	0.8066	0.9834	0.9958	0.9916	0.9574	33.2890	0.9802
$\mathcal{F}^J_{Su}$	0.8802	0.9772	0.9475	0.1115	0.9610	0.7052	0.3149	31.6552	0.9719
$\mathcal{F}^J_0$	0.6756	0.6046	0.5194	0.6210	0.4490	0.6772	0.5880	35.4814	0.9843
$\hat{\mathcal{F}}^J_b$	0.7590	0.4830	0.4354	0.8542	0.9588	0.6813	0.5650	36.7405	0.9891
$\hat{\mathcal{F}}^J_{bq}$	0.9170	0.4050	0.5244	0.8435	0.9874	0.8081	0.6531	35.9926	0.9872
$\mathcal{F}^J$	0.5398	0.6425	0.4598	0.6159	0.4344	0.5894	0.5317	35.9855	0.9866



(a) $\mathcal{F}^J_{S_qS_b}$ [Sta+10a, Sta+10b, SL11] (b) $\mathcal{F}^J$

Figure 5.8: ROC curves achieved by $\mathcal{F}^J$ and $\mathcal{F}^J_{S_qS_b}$ [Sta+10a, Sta+10b, SL11] against JPEG forensic detectors. Results are obtained on UCIDTest dataset.

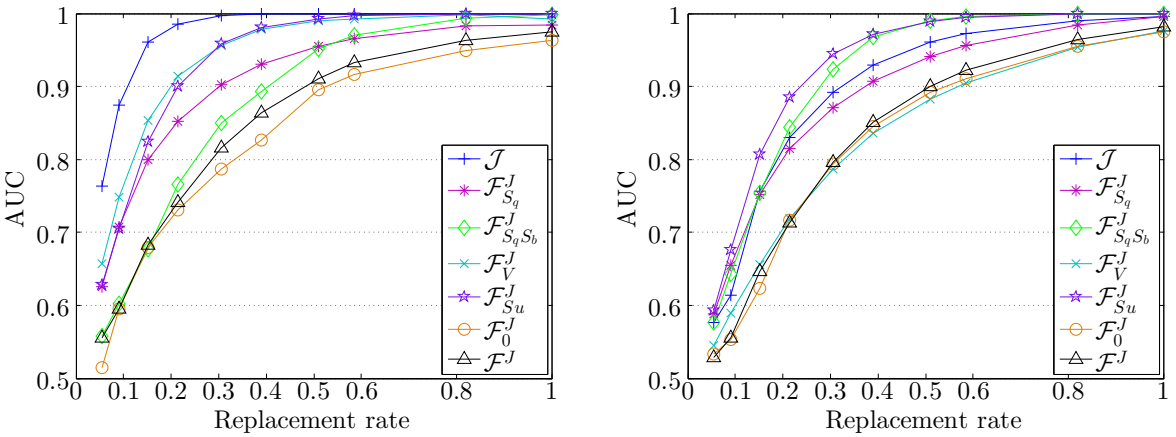
From both Table 5.4 and the ROC curves shown in Figure 5.8, we can see that our anti-

forensic JPEG image has a better forensic undetectability compared with Stamm *et al.*'s $\mathcal{F}^J_{S_qS_b}$. The other version of Stamm *et al.*'s anti-forensic JPEG image $\mathcal{F}^J_{S_q}$ processed by the DCT histogram smoothing can be well detected by the JPEG forensic detectors which were designed targeting at it [Val+11, LB11]. The anti-forensic JPEG image $\mathcal{F}^J_V$ created by the perceptual anti-forensic dithering [VTT11] has a similar anti-forensic performance as $\mathcal{F}^J_{S_q}$, while achieving a higher SSIM value on average. The median filtering based deblocking [Sta+10b] improves the undetectability of the anti-forensic JPEG image $\mathcal{F}^J_{S_qS_b}$ against forensic detectors, but with a PSNR value loss of 6.64 dB on average, compared to JPEG image $\mathcal{J}$ (see the last 2 columns of Table 5.4).

Figure 5.8-(b) shows that the proposed method succeeds in fooling all the 3 JPEG blocking detectors, yet through the minimization of a different TV-based blocking measure in Eq. (4.6). Our method is also capable of fooling the advanced detector K_V [Val+11, VTT13]. The reason might be that the TV term in Eq. (4.4) manages to suppress the unnatural noises utilized by this detector. Furthermore, the calibration-based detector K_L [LB11] is defeated by the *de-calibration* operation. Even though the quantization table estimation based detector K^Q_F is proven to be not very reliable in Section 4.2.1, we still test it on our anti-forensic JPEG images $\mathcal{F}^J$ on UCIDTest dataset, 93.70% of which can be passed off as never JPEG compressed (making the true positive rate be 6.30%). Our method successfully fools existing detectors, at the cost of a slightly lower visual quality than the JPEG compressed image: 1.11 dB of PSNR loss and 0.0053 of SSIM value loss on average. Compared to Stamm *et al.*'s anti-forensic JPEG image $\mathcal{F}^J_{S_qS_b}$ [Sta+10a, Sta+10b, SL11], our method achieves a better tradeoff between the forensic undetectability and the visual quality of processed images: the average PSNR has been improved by 5.53 dB and 0.0357 of SSIM gain has been achieved; meanwhile our method achieves a better overall forensic undetectability.

Concerning the SVM-based detectors K^{S100}_{Li} [CS08, LLH12], and K^{S686}_P [PBF10] which are initially designed for steganalysis, we follow the experimental setup described in Section 2.2.1.2. The AUC value as a function of the image replacement rate for different kinds of images, is shown in Figure 5.9. We can see that our anti-forensic JPEG image $\mathcal{F}^J$ does not perform the best when tested by K^{S100}_{Li} and K^{S686}_P. However, unlike $\mathcal{F}^J_V$ or $\mathcal{F}^J_{S_qS_b}$, the performance of $\mathcal{F}^J$ is quite stable and in general quite satisfactory when compared with other anti-forensic images. Meanwhile, $\mathcal{F}^J_0$ created using the method described in Chapter 4 slightly outperforms $\mathcal{F}^J$. This is understandable, as we explicitly smooth DCT histograms for creating $\mathcal{F}^J$, which may lead to more changes in image statistics. When the replacement rate is about 0.10, the performance of our anti-forensic JPEG image is quite good, with a relatively low AUC value of the SVM-based detectors. In this case, we can safely replace a 112×160 block in a 384×512 UCID image. This is enough for many image forgery creation scenarios, *e.g.*, replacing the head of one person in the picture. We remain reserved on whether the proposed JPEG anti-forensic method is able to disguise a whole JPEG image as uncompressed, as it can still be well detected using machine learning based forensic methods. However, it is still highly applicable in various JPEG anti-forensic scenarios, *e.g.*, image splicing, and disguising double JPEG compression artifacts (see results in Section 5.4).

Figure 5.10 shows the anti-forensic JPEG images created from an example JPEG image



(a) K_{Li}^{S100} [CS08, LLH12] (b) K_P^{S686} [PBF10]

Figure 5.9: The AUC value as a function of image replacement rate for different kinds of images, when tested using the SVM-based detectors. Results are obtained by testing on UCIDTE dataset and training on UCIDTR dataset.

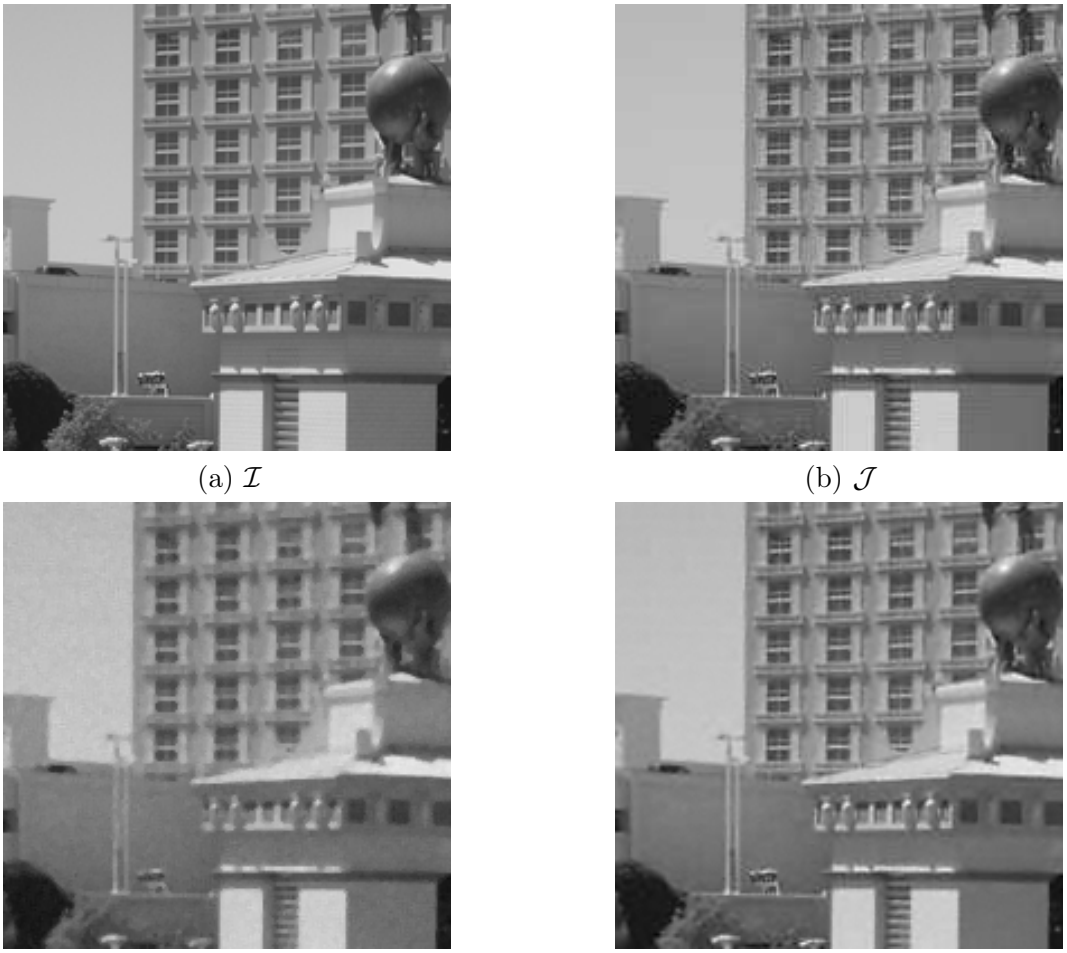
with quality factor 50. As expected, Figure 5.10-(d) processed using the proposed JPEG anti-forensic method better preserves the image details and the edges than -(c). It can also be observed that even after the deblocking operation [Sta+10b, SL11], the spatial-domain noise introduced by the dithering signal [Sta+10a, SL11] can still be noticed in the smooth areas of the image in -(c). Some example DCT histograms of the anti-forensic JPEG image in Figure 5.10-(d) are shown in Figure 5.11: no noticeable artifacts appear.

5.3.3 Computation Cost

Table 5.5: Comparison of the average time taken to create different kinds of anti-forensic JPEG images. Results are obtained on UCIDTest92 dataset.

	$\mathcal{F}^J_{S_q}$	$\mathcal{F}^J_{S_qS_b}$	$\mathcal{F}^J_V$	$\mathcal{F}^J_{Su}$	$\mathcal{F}^J_0$	$\mathcal{F}^J$
in second(s)	0.0503	0.0879	0.4380	0.0201	7.0815	176.8205

Table 5.5 records the average time taken for the generation of different anti-forensic JPEG images on UCIDTest92 dataset, using Matlab R2012b, on a PC with 24G RAM and 2.80GHz CPU. We can see that the proposed JPEG anti-forensic method requires around 3 minutes to create an anti-forensic JPEG image on average, using the unoptimized Matlab code. The bottleneck of the computation cost lies in the perceptual DCT histogram smoothing, as the computation cost is $O(D^3)$ using the Hungarian algorithm. However, the reduction of the dimensionality of the assignment problem discussed in Section 5.3.1 can effectively lower the computation cost. Apparently, the proposed method is more complex and more computationally demanding compared to other state-of-the-art JPEG anti-forensic methods. However, the benefit is a better tradeoff between the forensic undetectability and the image quality as shown in Section 5.3.2. Moreover, in practice, usually the forger does not need to create a



(a) $\mathcal{I}$ (b) $\mathcal{J}$

(c) $\mathcal{F}^J_{S_q S_b}$ [Sta+10a, Sta+10b, SL11] (d) $\mathcal{F}^J$

Figure 5.10: Example results (close-up images) of $\mathcal{F}^J$ compared with $\mathcal{I}$, $\mathcal{J}$, and $\mathcal{F}^J_{S_q S_b}$ [Sta+10a, Sta+10b, SL11], where $\mathcal{J}$ is compressed with quality factor 50. Our anti-forensic JPEG image $\mathcal{F}^J$ has a better image quality compared with $\mathcal{F}^J_{S_q S_b}$.

large amount of image forgeries. It is therefore acceptable to take several minutes to generate an anti-forensic JPEG image.

5.4 Hiding Traces of Double JPEG Compression Artifacts

As JPEG is one of the most commonly used image compression formats today, it is a very likely scenario that a forger alters something in a JPEG image and re-saves it using the JPEG format again. As explained in Section 1.3.1, this will lead to double JPEG compression artifacts which may be utilized by forensic investigators. An investigator may wish to tell whether an image has been double JPEG compressed (exposing aligned or non-aligned double JPEG compression artifacts), and furthermore which part of the image has been altered (forgery localization via

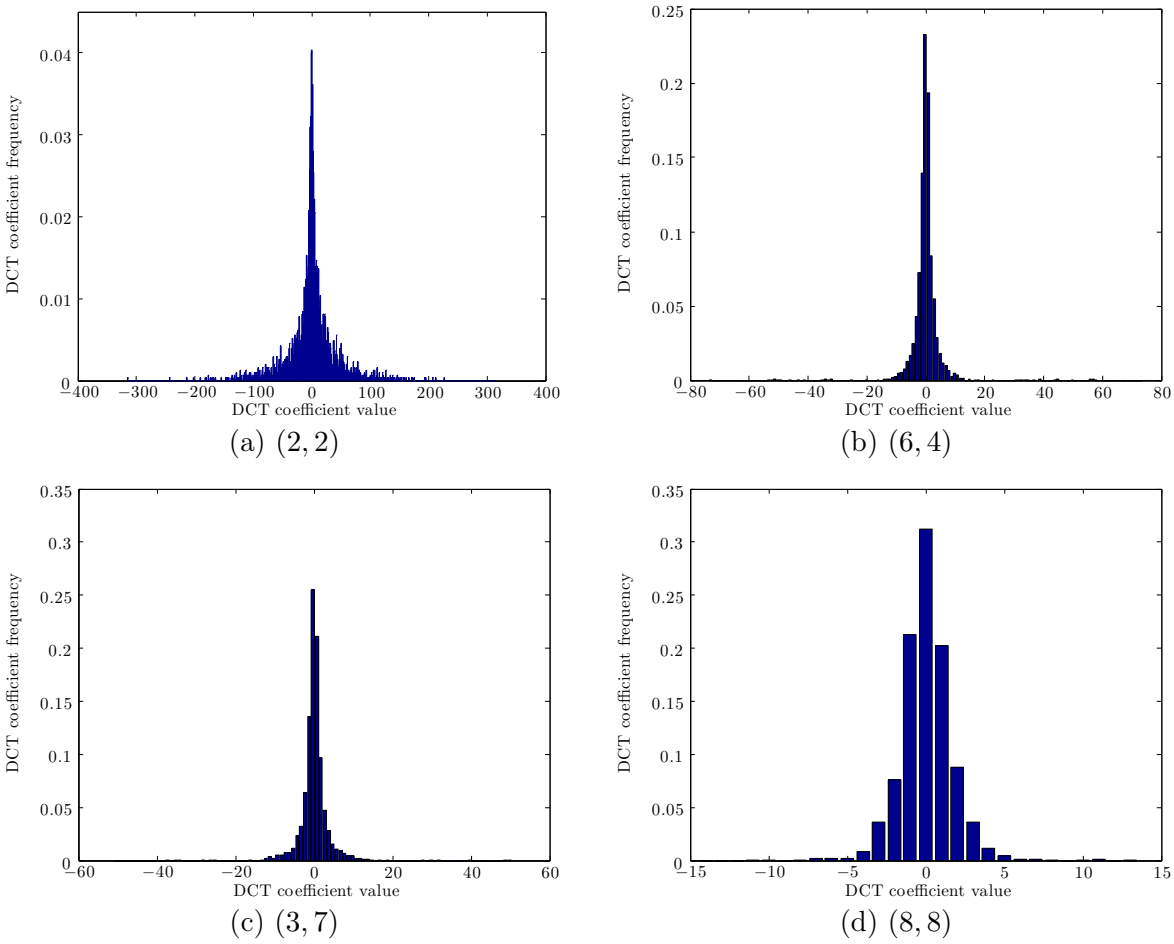


Figure 5.11: Example DCT histograms of different subbands of an anti-forensic JPEG image $\mathcal{F}^J$ shown in Figure 5.10-(d). No noticeable comb-like DCT quantization artifacts appear in the histograms.

analysis of double JPEG compression artifacts). In the following, we show how our JPEG anti-forensic method can be used to deceive three state-of-the-art image forensic algorithms [PF08, BP12a, BP12b] that take advantage of double JPEG compression artifacts.

In this section, single JPEG compressed images are only JPEG compressed with quality factor QF_2, while double JPEG compressed images are firstly JPEG compressed with quality factor QF_1 and JPEG compressed again with quality factor QF_2. Image cropping or content modification may occur between the two compressions. For anti-forensic testing, the JPEG anti-forensic methods in [Sta+10a, Sta+10b, SL11, VTT11, SS11] and the proposed JPEG anti-forensic methods in both Chapter 4 and this chapter are applied on the JPEG image after the first compression with QF_1. Then the anti-forensic JPEG image is unchanged, cropped by a random grid shift, or partly altered. Finally the resulting image is JPEG compressed again with QF_2 to create anti-forensic double JPEG compressed image.

We conduct large-scale tests on UCID corpus [SS04], while following the experimental

settings in the original papers of double JPEG detectors [PF08, BP12a, BP12b]. Every dataset contains double JPEG compressed images first with QF_1 and then with QF_2 between which there might be JPEG anti-forensics applied, cropping, or partial image replacement according to different scenarios. In Section 5.4.1 and Section 5.4.2, each dataset also includes single JPEG compressed images with QF_2. In order to avoid tedious repetition later, we hereby explain the naming rule of the image datasets we will use in this section. Unless otherwise stated, the dataset created from UCIDTE or UCIDTest100 (see Section 2.3.1 for more descriptions about the datasets) is written in bold letters. The end of the name of the dataset '-***R***' indicates which kind of (or no) JPEG anti-forensics is applied to the single JPEG compressed image with QF_1 before further processing. It can the followings:

- '-**T**': no JPEG anti-forensics is applied;
- '-**S**': Stamm *et al.*'s dithering based JPEG anti-forensic method [Sta+10a, SL11] is applied;
- '-**SS**': Stamm *et al.*'s dithering [Sta+10a, SL11] and median filtering based JPEG anti-forensic method [Sta+10b, SL11] is applied;
- '-**V**': Valenzise *et al.*'s perceptual dithering based JPEG anti-forensic method [VTT11] is applied;
- '-**Su**': Sutthiwan and Shi's SAZ attack [SS11] is applied;
- '-$\mathbf{F}_0$': our previously proposed TV-based anti-forensic method in Chapter 4 is applied;
- '-**F**': the proposed four-step JPEG anti-forensic method in this chapter is applied.

5.4.1 Hiding Traces of Aligned Double JPEG Compression

In [PF08], Pevný and Fridrich proposed a method using the SVM with feature vectors formed by DCT histograms in the low-frequency subbands to classify single and double JPEG compressed images. For constructing the feature vector, they consider 9 low-frequency AC subbands, and for each of them a 16-bin histogram is computed. The 144-dimensional feature vectors are then fed to an SVM.

In order to train the SVM-based A-DJPG (Aligned Double JPEG) compression detector [PF08], each image in UCIDTR is firstly JPEG compressed with the primary quality factor QF_1 and then compressed again with the secondary quality factor $QF_2 \neq QF_1$ to create the A-DJPG compressed images. Here, $QF_1 \in \{50, 56, 63, 69, 81, 88, 94\}$, and $QF_2 = 75$, following [PF08]. The single JPEG compressed images are created by JPEG compressing the original uncompressed images with QF_2. Then we have $7 \times 500 + 500 = 4000$ images for training the detector, using the LIBSVM [CL11].

Each UCIDTE image is firstly JPEG compressed with QF_1, and then compressed again with QF_2 to create the A-DJPG compressed images. During the two JPEG compressions,

anti-forensic operations may occur. Meanwhile, each UCIDTE image is JPEG compressed once with QF_2 for creating the single JPEG compressed image. For forensic testing, we create 7 datasets each of which has 4000 images as well. The name of the dataset follows the pattern **A-DJPG-*R***.

The ROC curves in Figure 5.12 show that the proposed method successfully masks the presence of double JPEG compression artifacts. The AUC value of the A-DJPG compression detector [PF08] is decreased to 0.4832 for **A-DJPG-F** from 0.9274 for the training dataset and 0.9152 for **A-DJPG-T**. Other state-of-the-art JPEG anti-forensic methods [Sta+10a, Sta+10b, VTT11, SS11] as well as the one proposed in Chapter 4 achieve similar anti-forensic performance, however with higher image quality loss than the proposed method in this chapter, as reported in Table 5.6. Among different anti-forensic double JPEG compressed images, our anti-forensic images have the highest visual quality (with the uncompressed image as the reference).

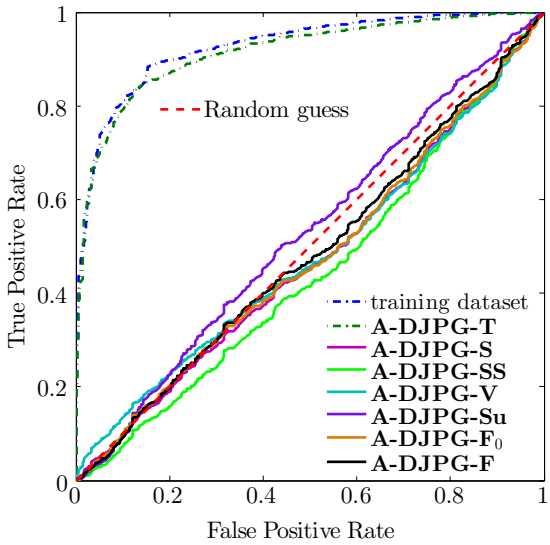


Figure 5.12: ROC curves achieved on **A-DJPG-*R*** against the SVM-based A-DJPG compression detector [PF08]. Results are obtained by testing on UCIDTE dataset and training on UCIDTR dataset. All the JPEG anti-forensic methods in consideration are able to move the ROC curves close to the random guess line.

Table 5.6: Image quality (with the uncompressed image as the reference) comparison of (anti-forensic) double JPEG compressed images created from UCIDTE for different datasets **A-DJPG-*R***. **A-DJPG-F** achieves the best image quality among different anti-forensic double JPEG compressed images in comparison.

	-T	**-S**	**-SS**	**-V**	**-Su**	**-F$_0$**	**-F**
PSNR	35.0916	32.8974	30.4112	32.8090	31.0414	34.1906	34.4417
SSIM	0.9879	0.9690	0.9463	0.9762	0.9664	0.9800	0.9821

5.4.2 Hiding Traces of Non-Aligned Double JPEG Compression

Bianchi and Piva [BP12a] analyzed the statistical change in the DCT coefficients of the DC component, after a JPEG image is compressed again with a non-aligned 8×8 grid. A simple but powerful threshold detector is constructed [BP12a], which is based on measuring the non-uniformity of a suitably defined integer periodicity map of the DC coefficients.

We compress each UCIDTE image with QF_1, then the JPEG image is cropped with a random shift $(i, j) \neq (0, 0)$, with $0 \leq i, j \leq 7$. At last the cropped image is JPEG compressed again with QF_2 to create the NA-DJPG (Non-Aligned Double JPEG) compressed image. Meanwhile each UCIDTE image is JPEG compressed once with QF_2 for creating the single JPEG compressed image. Anti-forensic operation may occur after the first JPEG compression with QF_1. For forensic testing, we create 7 datasets whose names follow the pattern **NA-DJPG-*R***. Here, as suggested in [BP12a], $QF_1 \in \{50, 53, 56, 59, 63, 66, 69, 72, 75, 78, 81, 84, 88, 91, 94\}$ and $QF_2 \in \{50, 53, 56, 59, 63, 66, 69, 72, 75, 78, 81, 84, 88, 91, 94, 97\}$, yielding in total $15 \times 16 = 240$ different quality factor combinations for creating NA-DJPG compressed images. Therefore, we have $15 \times 16 \times 500 + 16 \times 500 = 128000$ images for each dataset.

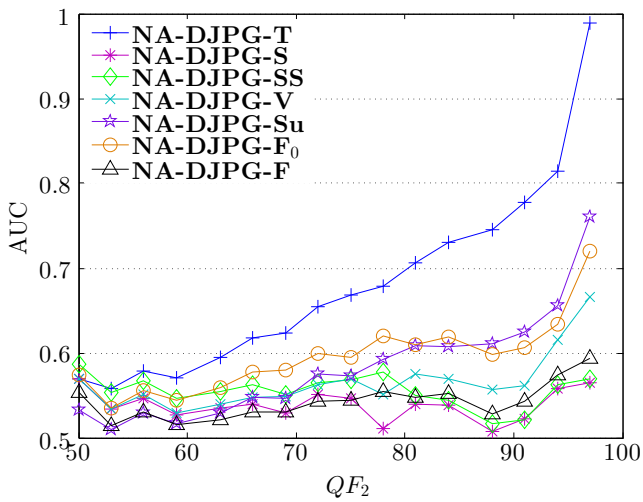


Figure 5.13: Average AUC value of the NA-DJPG compression detector [BP12a] as a function of QF_2, when tested on **NA-DJPG-*R***, created from UCIDTE. The improvement of forensic undetectability of **NA-DJPG-F** over **NA-DJPG-F**$_0$ shows the necessity of conducting the DCT histogram smoothing.

For each quality factor pair (QF_1, QF_2), the single JPEG compressed images together with their corresponding (anti-forensic) double JPEG compressed images are tested using the NA-DJPG detector [BP12a]. Then the AUC value can be computed for different kinds of images. Figure 5.13 shows the average AUC value over quality factor QF_1, under a fixed value of QF_2. We can see that $\mathcal{F}^J_{S_q}$ [Sta+10a, SL11], $\mathcal{F}^J_{S_q S_b}$ [Sta+10b, SL11], both of which have DCT histograms explicitly smoothed, keep a good forensic undetectability against this NA-DJPG detector. Although our previous JPEG anti-forensic method described in Chapter 4 can successfully fool many existing JPEG forensic detectors (as shown in Figure 4.6 and

Table 5.4), the gaps in the DCT domain are not well filled (an example can be seen in Figure 4.10-(b)), which might be exposed by the NA-DJPG compression detector [BP12a]. Similarly, anti-forensic double JPEG compressed images created from $\mathcal{F}^J_{Su}$ [SS11] can also be detected, to some extent, by the NA-DJPG compression detector [BP12a], especially when QF_2 is high. By contrast, with the application of the proposed JPEG anti-forensic method, the AUC of the NA-DJPG detector [BP12a] is successfully kept close to 0.5 (random guess). This proves the necessity of an explicit DCT histogram smoothing for JPEG anti-forensics and the effectiveness of the proposed perceptual histogram smoothing method, because no integer periodicity can be detected by the NA-DJPG detector in the DCT histogram of our anti-forensic double JPEG compressed images.

Besides a good forensic undetectability, the proposed anti-forensic method is able to create anti-forensic NA-DJPG compressed images with the highest image quality among all 6 kinds of anti-forensic images (see results in Table 5.7).

Table 5.7: Image quality (with the uncompressed image as the reference) comparison of (anti-forensic) double JPEG compressed images created from UCIDTE for different datasets **NA-DJPG-*R***. **NA-DJPG-F** achieves the best image quality among different anti-forensic double JPEG compressed images in comparison.

	-T	**-S**	**-SS**	**-V**	**-Su**	**-F$_0$**	**-F**
PSNR	34.6098	32.7958	30.2825	32.4824	30.9379	33.8345	34.0929
SSIM	0.9319	0.8650	0.8487	0.8864	0.8898	0.9168	0.9222

5.4.3 Fooling JPEG Artifacts Based Image Forgery Localization

Bianchi and Piva [BP12b] derived a likelihood-map indicating the probability for each 8×8 block of being double JPEG compressed, under the hypothesis of the presence of A-DJPG or NA-DJPG compression artifacts in the tampered image.

We conduct the test on the 100 images in the UCIDTest100 dataset. Following [BP12b], we first compress each UCIDTest100 image with QF_1; the resulting image is partly replaced using the original uncompressed image, and then compressed again with QF_2. After the primary JPEG compression, image cropping and/or anti-forensics may occur. In total, $7 \times 6 = 42$ datasets are created considering 7 kinds of images and 6 different scenarios. We name the dataset as **LOC-*E*-DJPG-*K*/*L*-*R***, where the italic letters may change to represent different scenarios. ***E*** can be '**A**', or '**NA**'. '**NA**' indicates that before the second compression the image is cropped by a random shift $(i, j) \neq (0, 0), 0 \leq i, j \leq 7$; whereas '**A**' means there is no image cropping happening. ***K*/*L*** indicates how much portion of the image has undergone double JPEG compression, which also implies how much portion of the image has been replaced by the original uncompressed image before the second JPEG compression. When ***K*/*L*** is '**1/2**', it indicates the left half of the image is replaced using the original image; when ***K*/*L*** is '**15/16**', the central 1/16 portion of the image is replaced; when ***K*/*L*** is '**1/16**', the whole image except its central 1/16 portion is replaced. In all datasets, QF_1

and QF_2 are taken from $\{50, 56, 63, 69, 75, 81, 88, 94\}$ and $\{50, 56, 63, 69, 75, 81, 88, 94, 100\}$, respectively. Therefore each dataset has $8 \times 9 \times 100 = 7200$ images.

Figure 5.14 shows the average AUC value over all possible QF_1 (under a fixed value of QF_2) achieved by the detector [BP12b] for the different scenarios and different kinds of images. Note that the results shown here are computed from the standard map instead of the simplified map (see [BP12b] for details). The blue curves demonstrate the effectiveness of the detector when no JPEG anti-forensics is applied after the first compression. With the help of JPEG anti-forensics, the forgery localization detector [BP12b] can be well fooled. Our anti-forensic double JPEG compressed image again achieves the best image visual quality among all kinds of anti-forensic images (see Tables 5.8-5.13).

Table 5.8: Image quality (with the uncompressed image as the reference) comparison of (anti-forensic) double JPEG compressed images created from UCIDTest100 for different datasets **LOC-A-DJPG-15/16-*R***. **LOC-A-DJPG-15/16-F** achieves the best image quality among different anti-forensic double JPEG compressed images in comparison.

	-T	**-S**	**-SS**	**-V**	**-Su**	**-F$_0$**	**-F**
PSNR	35.5248	33.5597	30.8639	33.4702	31.5678	34.5200	34.8080
SSIM	0.9875	0.9744	0.9528	0.9786	0.9703	0.9798	0.9819

Table 5.9: Image quality (with the uncompressed image as the reference) comparison of (anti-forensic) double JPEG compressed images created from UCIDTest100 for different datasets **LOC-NA-DJPG-15/16-*R***. **LOC-NA-DJPG-15/16-F** achieves the best image quality among different anti-forensic double JPEG compressed images in comparison.

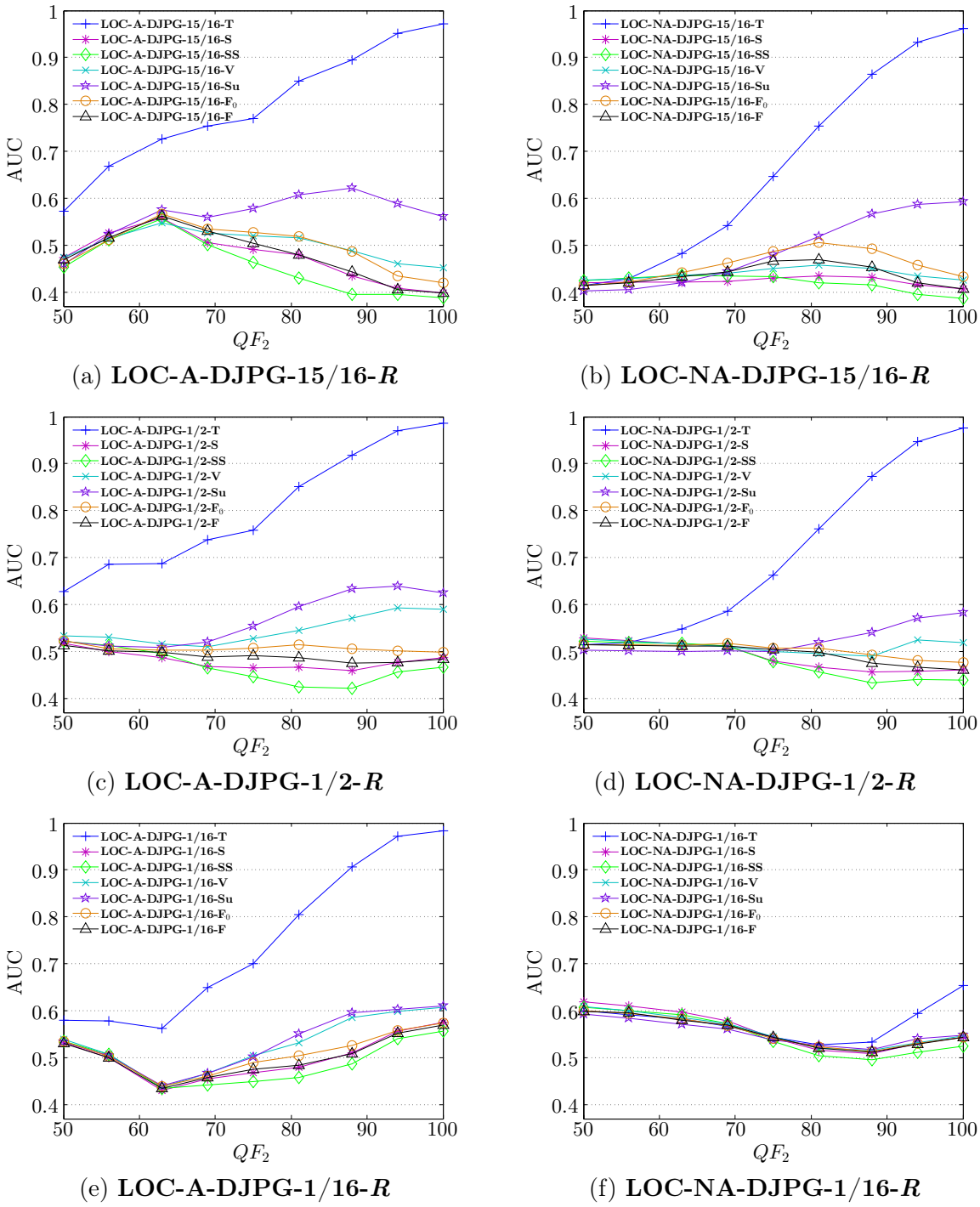
	-T	**-S**	**-SS**	**-V**	**-Su**	**-F$_0$**	**-F**
PSNR	35.1211	33.1837	30.8331	32.9189	31.4694	34.3396	34.5884
SSIM	0.9395	0.8776	0.8726	0.8972	0.9067	0.9265	0.9308

Table 5.10: Image quality (with the uncompressed image as the reference) comparison of (anti-forensic) double JPEG compressed images created from UCIDTest100 for different datasets **LOC-A-DJPG-1/2-*R***. **LOC-A-DJPG-1/2-F** achieves the best image quality among different anti-forensic double JPEG compressed images in comparison.

	-T	**-S**	**-SS**	**-V**	**-Su**	**-F$_0$**	**-F**
PSNR	36.3810	34.7737	32.7724	34.7288	33.2466	35.6653	35.8817
SSIM	0.9897	0.9829	0.9713	0.9852	0.9799	0.9855	0.9866

5.5 Summary

In this chapter, we propose a novel perceptual DCT histogram smoothing method to improve the TV-based deblocking based JPEG anti-forensic method described in Chapter 4. The proposed JPEG anti-forensic method designs four steps to alternatively remove the artifacts



(a) **LOC-A-DJPG-15/16-*R***

(b) **LOC-NA-DJPG-15/16-*R***

(c) **LOC-A-DJPG-1/2-*R***

(d) **LOC-NA-DJPG-1/2-*R***

(e) **LOC-A-DJPG-1/16-*R***

(f) **LOC-NA-DJPG-1/16-*R***

Figure 5.14: Average AUC value as a function of QF_2 of the forgery localization detector [BP12b], when tested on **LOC-*E*-DJPG-*K*/*L*-*R***, created from UCIDTest100. The black curves with triangles indicating AUC values around 0.5, prove the effectiveness of the proposed four-step JPEG anti-forensic method in masking double JPEG compression artifacts.

Table 5.11: Image quality (with the uncompressed image as the reference) comparison of (anti-forensic) double JPEG compressed images created from UCIDTest100 for different datasets **LOC-NA-DJPG-1/2-*R***. **LOC-NA-DJPG-1/2-F** achieves the best image quality among different anti-forensic double JPEG compressed images in comparison.

	-T	**-S**	**-SS**	**-V**	**-Su**	**-F$_0$**	**-F**
PSNR	36.1007	34.5186	32.7271	34.3632	33.1516	35.5217	35.7151
SSIM	0.9492	0.9169	0.9146	0.9272	0.9313	0.9423	0.9447

Table 5.12: Image quality (with the uncompressed image as the reference) comparison of (anti-forensic) double JPEG compressed images created from UCIDTest100 for different datasets **LOC-A-DJPG-1/16-*R***. **LOC-A-DJPG-1/16-F** achieves the best image quality among different anti-forensic double JPEG compressed images in comparison.

	-T	**-S**	**-SS**	**-V**	**-Su**	**-F$_0$**	**-F**
PSNR	38.1497	37.4751	36.5211	37.3986	36.8277	37.8649	37.9576
SSIM	0.9919	0.9913	0.9894	0.9915	0.9905	0.9914	0.9915

Table 5.13: Image quality (with the uncompressed image as the reference) comparison of (anti-forensic) double JPEG compressed images created from UCIDTest100 for different datasets **LOC-NA-DJPG-1/16-*R***. **LOC-NA-DJPG-1/16-F** achieves the best image quality among different anti-forensic double JPEG compressed images in comparison.

	-T	**-S**	**-SS**	**-V**	**-Su**	**-F$_0$**	**-F**
PSNR	38.0669	37.4034	36.4870	37.3130	36.7781	37.8137	37.9028
SSIM	0.9597	0.9564	0.9547	0.9574	0.9571	0.9588	0.9591

in the spatial domain and in the DCT domain. The resulting anti-forensic JPEG image is able to fool JPEG compression detectors and achieve a better image visual quality than state-of-the-art JPEG anti-forensic methods and the method proposed in Chapter 4. From the results against the NA-DJPG detector [BP12a], the necessity of histogram smoothing can be seen, as the anti-forensic double JPEG compressed image in **NA-DJPG-F$_0$** created using the method in Chapter 4 can be to some extent detected when QF_2 is relatively high (see Figure 5.13). This problem is successfully tackled by the explicit DCT histogram smoothing procedure proposed in Section 5.2.2.

Indeed, the proposed four-step JPEG anti-forensic method is heuristic, but effective. It is an interesting but very difficult problem to design a single-step attack to remove JPEG artifacts, as existing detectors work in both spatial and DCT domains. Furthermore, we have to consider the visual quality of the processed image. As known, SSIM is non-convex, which makes it hard to optimize. In order to drag the processed image out from the detection regions of *multiple* detectors working in *different domains*, and at the same time to keep a *high image quality* under the evaluation of *both PSNR and SSIM metrics*, in each of the four steps we consider a different optimization problem. Moreover, similar to Stamm *et al.*'s anti-forensic JPEG image creation process [Sta+10a, Sta+10b, SL11], the JPEG artifacts in the spatial

domain and the DCT domain are considered alternatively. With a local modification to the image considering multiple metrics, we are able to create anti-forensic JPEG images with a good tradeoff between the forensic undetectability and the image quality.

In Chapters 4-5, the proposed JPEG anti-forensic methods are inspired by the TV-based JPEG image post-processing [ADF05]. The TV can be considered as a simple but effective image prior. Besides, in the proposed TV-based JPEG deblocking framework, there is no likelihood term considering the JPEG compression process as in the MAP-based JPEG post-processing methods [RS05, SC07]. In the literature of image restoration, there exist more sophisticated image prior models [RS05, SC07, ZW11] than the TV, and spatial-domain compression noise model for JPEG compression [RS05]. In Chapter 6, still following the research line of designing image anti-forensics leveraging on image restoration, we will propose another JPEG anti-forensic method using a more advanced image prior than the TV and with a likelihood term modeling the JPEG compression process.

5.A Appendix: The p.m.f. of the Dithering Signal Using the Laplacian Model

Here, we only consider AC components of the image. The probability density function (p.d.f.) of the distribution of the dithering signal N is denoted as $P(N=n|Y=y)$. P can be easily computed (see Eqs. (5.5)-(5.7)) according to the sign of the quantized DCT coefficient Y. Here, the p.m.f. of the rounded dithering signal are calculated with domain defined as the integer set $\{-\lfloor\frac{\mathbf{Q}_{r,c}}{2}\rfloor, -\lfloor\frac{\mathbf{Q}_{r,c}}{2}\rfloor+1, \cdots, \lfloor\frac{\mathbf{Q}_{r,c}}{2}\rfloor\}$. Eqs. (5.12)-(5.13) below are the p.m.f. of the rounded dithering signal for quantization bin 0 when $\mathbf{Q}_{r,c}$ is an odd number and an even number, respectively. For quantization bin $b \neq 0$, the p.m.f. of the rounded dithering signal are listed in Eqs. (5.14)-(5.17) according to the sign and the parity of b.

$$P_m^o(N=n|Y=0) = \begin{cases} 2c_0\lambda^{-1}\left(1-e^{-\lambda/2}\right) & \text{if} \quad n=0 \\ -c_0\lambda^{-1}e^{-\lambda(n+1/2)}\left(1-e^{\lambda}\right) & \text{if} \quad n=1,\cdots,\lfloor\frac{\mathbf{Q}_{r,c}}{2}\rfloor \\ c_0\lambda^{-1}e^{\lambda(n+1/2)}\left(1-e^{-\lambda}\right) & \text{if} \quad n=-\lfloor\frac{\mathbf{Q}_{r,c}}{2}\rfloor,\cdots,-1 \\ 0 & \text{otherwise,} \end{cases} \tag{5.12}$$

$$P_m^e(N=n|Y=0) = \begin{cases} 2c_0\lambda^{-1}\left(1-e^{-\lambda/2}\right) & \text{if} \quad n=0 \\ -c_0\lambda^{-1}e^{-\lambda\mathbf{Q}_{r,c}/2}\left(1-e^{\lambda/2}\right) & \text{if} \quad n=-\frac{\mathbf{Q}_{r,c}}{2},\frac{\mathbf{Q}_{r,c}}{2} \\ -c_0\lambda^{-1}e^{-\lambda(n+1/2)}\left(1-e^{\lambda}\right) & \text{if} \quad n=1,\cdots,\frac{\mathbf{Q}_{r,c}}{2}-1 \\ c_0\lambda^{-1}e^{\lambda(n+1/2)}\left(1-e^{-\lambda}\right) & \text{if} \quad n=-\frac{\mathbf{Q}_{r,c}}{2}+1,\cdots,-1 \\ 0 & \text{otherwise,} \end{cases} \tag{5.13}$$

$$P_m^o(N=n|Y=y,y>0) = \begin{cases} -c_1\lambda^{-1}e^{-\lambda(n+1/2)}\left(1-e^{\lambda}\right) & \text{if} \quad n=-\lfloor\frac{\mathbf{Q}_{r,c}}{2}\rfloor,\cdots,\lfloor\frac{\mathbf{Q}_{r,c}}{2}\rfloor \\ 0 & \text{otherwise,} \end{cases} \tag{5.14}$$

$$P_m^e(N=n|Y=y,y>0) = \begin{cases} c_1\lambda^{-1}e^{\lambda\mathbf{Q}_{r,c}/2}\left(1-e^{-\lambda/2}\right) & \text{if} \quad n=-\frac{\mathbf{Q}_{r,c}}{2} \\ -c_1\lambda^{-1}e^{-\lambda\mathbf{Q}_{r,c}/2}\left(1-e^{\lambda/2}\right) & \text{if} \quad n=\frac{\mathbf{Q}_{r,c}}{2} \\ -c_1\lambda^{-1}e^{-\lambda(n+1/2)}\left(1-e^{\lambda}\right) & \text{if} \quad n=-\frac{\mathbf{Q}_{r,c}}{2}+1,\cdots,\frac{\mathbf{Q}_{r,c}}{2}-1 \\ 0 & \text{otherwise,} \end{cases} \tag{5.15}$$

$$P_m^o(N=n|Y=y,y<0) = \begin{cases} c_1\lambda^{-1}e^{\lambda(n+1/2)}\left(1-e^{-\lambda}\right) & \text{if} \quad n=-\lfloor\frac{\mathbf{Q}_{r,c}}{2}\rfloor,\cdots,\lfloor\frac{\mathbf{Q}_{r,c}}{2}\rfloor \\ 0 & \text{otherwise,} \end{cases} \tag{5.16}$$

$$P_m^e(N=n|Y=y,y<0) = \begin{cases} -c_1\lambda^{-1}e^{-\lambda\mathbf{Q}_{r,c}/2}\left(1-e^{\lambda/2}\right) & \text{if} \quad n=-\frac{\mathbf{Q}_{r,c}}{2} \\ c_1\lambda^{-1}e^{\lambda\mathbf{Q}_{r,c}/2}\left(1-e^{-\lambda/2}\right) & \text{if} \quad n=\frac{\mathbf{Q}_{r,c}}{2} \\ c_1\lambda^{-1}e^{\lambda(n+1/2)}\left(1-e^{-\lambda}\right) & \text{if} \quad n=-\frac{\mathbf{Q}_{r,c}}{2}+1,\cdots,\frac{\mathbf{Q}_{r,c}}{2}-1 \\ 0 & \text{otherwise.} \end{cases} \tag{5.17}$$

5.B Appendix: The Constraints Used for Modeling the DCT Coefficients

In Section 5.2.2.2, for building the *adaptive local* dithering signal model for AC components, we combine the Laplacian model and the uniform model together. A key point to establish the adaptive local model for the dithering signal is that we try to find an appropriate parameter λ_b of the Laplacian model for each quantization bin b. If we cannot find a valid value for λ_b, the uniform model is used instead for the current and following quantization bin(s). The parameter λ_b is derived by solving a constrained weighted least-squares fitting problem (see Eq. (5.4)), with λ_b bounded between λ_b^- and λ_b^+. In Section 5.2.2.2, we show how to search for the bound λ_b^+ (λ_b^- is set as 10^{-3}) for quantization bin 0 when $\mathbf{Q}_{r,c}$ is an odd number, using a numerical method. Here we explain how to search for the two bounds in other cases.

In the quantization bin 0 when $\mathbf{Q}_{r,c}$ is an even number

The empirical observation tells us that in the distribution of DCT coefficients of AC component, the probability of DCT coefficient decreases when the coefficient magnitude increases. Now we consider the quantization bin 0 when $\mathbf{Q}_{r,c}$ is an even number.

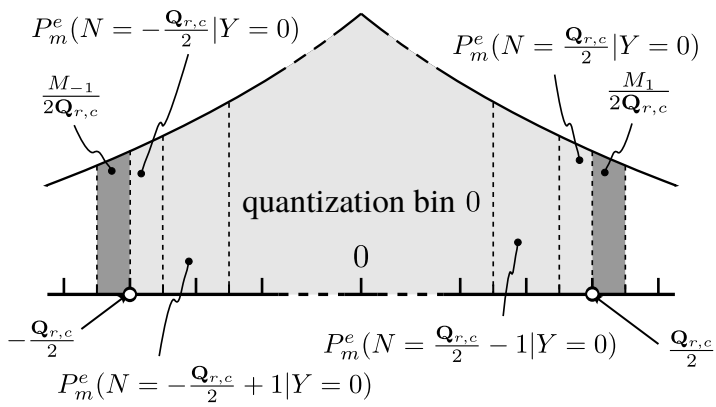


Figure 5.15: For the quantization bin 0, the probability of DCT coefficient falling in the leftmost (or rightmost) integer bin should be no smaller than either that in the rightmost integer bin of quantization bin -1 or that in the leftmost integer bin of quantization bin 1.

As illustrated in Figure 5.15, the probability of DCT coefficient falling in the leftmost (or rightmost) integer bin should be no smaller than either that in the rightmost integer bin of quantization bin -1 or that in the leftmost integer bin of quantization bin 1. For the moment, the neighboring quantization bins -1 and 1 are assumed to follow a uniform distribution. Similarly to the case in the quantization bin 0 when $\mathbf{Q}_{r,c}$ is an odd number, we still set

$\lambda_b^- = 10^{-3}$, whereas λ_b^+ can be found by solving:

$$\begin{aligned} \lambda_b^+ &= \underset{10^{-3}\leq\lambda\leq 1}{\arg\max}\ \lambda, \\ \text{subject to: } & P_m^e\left(N = \tfrac{\mathbf{Q}_{r,c}}{2} - 1 | Y = 0\right) \times M_0 \\ & \geq P_m^e\left(N = \tfrac{\mathbf{Q}_{r,c}}{2} | Y = 0\right) \times M_0 + \tfrac{1}{2}\max\left(\tfrac{M_{-1}}{\mathbf{Q}_{r,c}}, \tfrac{M_1}{\mathbf{Q}_{r,c}}\right) \end{aligned} \tag{5.18}$$

using a numerical method. Note that M_b ($b = B_{r,c}^-, B_{r,c}^- + 1, \cdots, B_{r,c}^+$) denotes the approximate probability that the DCT coefficient falls in quantization bin b.

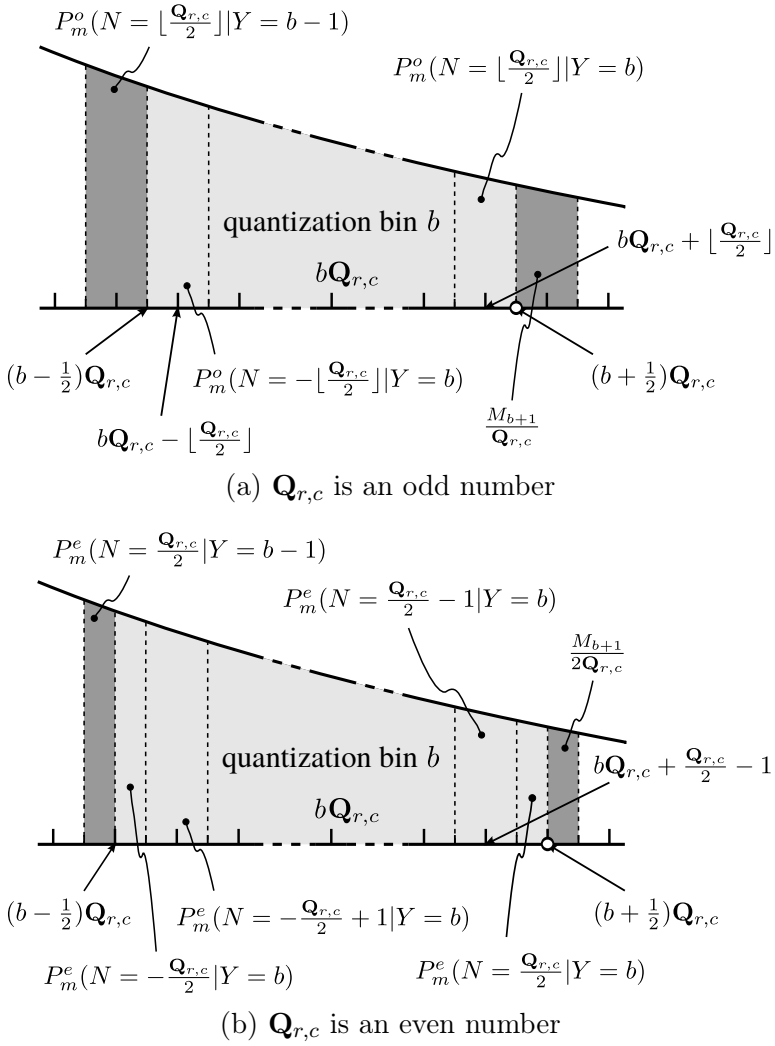
In the quantization bin $b > 0$

As illustrated in Figures 5.16-(a) and -(b), we consider a quantization bin $b > 0$, when $\mathbf{Q}_{r,c}$ is an odd number and an even number, respectively. The probability of DCT coefficient falling in the leftmost integer bin of quantization bin b should be no bigger than that in the rightmost integer bin of quantization bin $b-1$. Meanwhile, the probability of DCT coefficient falling in the rightmost integer bin of quantization bin b should be no smaller than that in the leftmost integer bin of quantization bin $b+1$. As the building of the dithering signal model is an iterative procedure, the distribution of the dithering signal N in the quantization bin $b-1$ has already been estimated in the last iteration. Hence, $P_m^o(N = n|Y = b-1)$ or $P_m^e(N = n|Y = b-1)$ is known. Moreover, for the quantization bin $b+1$, its dithering signal is assumed to follow a uniform distribution for the moment. Therefore, when $\mathbf{Q}_{r,c}$ is an odd number, the constraints that λ_b^- and λ_b^+ can be respectively found by solving:

$$\begin{aligned} \lambda_b^- &= \underset{10^{-3}\leq\lambda\leq\lambda_{b-1}}{\arg\min}\ \lambda \\ \text{subject to: } & \begin{cases} P_m^o\left(N = -\lfloor\frac{\mathbf{Q}_{r,c}}{2}\rfloor | Y = b\right) \times M_b \leq P_m^o\left(N = \lfloor\frac{\mathbf{Q}_{r,c}}{2}\rfloor | Y = b-1\right) \times M_{b-1} \\ P_m^o\left(N = \lfloor\frac{\mathbf{Q}_{r,c}}{2}\rfloor | Y = b\right) \times M_b \geq \frac{M_{b+1}}{\mathbf{Q}_{r,c}}, \end{cases} \end{aligned} \tag{5.19}$$

and

$$\begin{aligned} \lambda_b^+ &= \underset{10^{-3}\leq\lambda\leq\lambda_{b-1}}{\arg\max}\ \lambda \\ \text{subject to: } & \begin{cases} P_m^o\left(N = -\lfloor\frac{\mathbf{Q}_{r,c}}{2}\rfloor | Y = b\right) \times M_b \leq P_m^o(N = \lfloor\frac{\mathbf{Q}_{r,c}}{2}\rfloor | Y = b-1) \times M_{b-1} \\ P_m^o\left(N = \lfloor\frac{\mathbf{Q}_{r,c}}{2}\rfloor | Y = b\right) \times M_b \geq \frac{M_{b+1}}{\mathbf{Q}_{r,c}}. \end{cases} \end{aligned} \tag{5.20}$$



(a) $\mathbf{Q}_{r,c}$ is an odd number

(b) $\mathbf{Q}_{r,c}$ is an even number

Figure 5.16: For the quantization bin $b > 0$, the probability of DCT coefficient falling in the leftmost integer bin should be no bigger than that in the rightmost integer bin of the neighboring quantization bin $b-1$, meanwhile the probability of coefficients falling in the rightmost integer bin of quantization bin b should be no smaller than that in the leftmost integer bin of quantization bin $b+1$.

Similarly, λ_b^- and λ_b^+ are respectively defined as:

$$\lambda_b^- = \underset{10^{-3} \le \lambda \le \lambda_{b-1}}{\arg\min} \quad \lambda$$

$$\text{subject to: } \begin{cases} P_m^e\left(N = -\frac{\mathbf{Q}_{r,c}}{2} + 1 | Y = b\right) \times M_b \le P_m^e\left(N = -\frac{\mathbf{Q}_{r,c}}{2} | Y = b\right) \times M_b \\ \qquad\qquad + P_m^e\left(N = \frac{\mathbf{Q}_{r,c}}{2} | Y = b-1\right) \times M_{b-1} \\ P_m^e\left(N = \frac{\mathbf{Q}_{r,c}}{2} - 1 | Y = b\right) \times M_b \ge P_m^e\left(N = \frac{\mathbf{Q}_{r,c}}{2} | Y = b\right) \times M_b + \frac{M_{b+1}}{2\mathbf{Q}_{r,c}}, \end{cases} \tag{5.21}$$

and

$$\lambda_b^+ = \underset{10^{-3}\leq\lambda\leq\lambda_{b-1}}{\arg\max} \quad \lambda$$
$$\text{subject to: } \begin{cases} P_m^e\left(N = -\frac{\mathbf{Q}_{r,c}}{2}+1|Y=b\right)\times M_b \leq P_m^e\left(N=-\frac{\mathbf{Q}_{r,c}}{2}|Y=b\right)\times M_b \\ \qquad\qquad +P_m^e\left(N=\frac{\mathbf{Q}_{r,c}}{2}|Y=b-1\right)\times M_{b-1} \\ P_m^e\left(N=\frac{\mathbf{Q}_{r,c}}{2}-1|Y=b\right)\times M_b \geq P_m^e\left(N=\frac{\mathbf{Q}_{r,c}}{2}|Y=b\right)\times M_b + \frac{M_{b+1}}{2\mathbf{Q}_{r,c}}, \end{cases} \tag{5.22}$$

when $\mathbf{Q}_{r,c}$ is an even number.

In other words, using a numerical method, λ_b^- and λ_b^+ are found respectively as the smallest and largest number in the interval $[10^{-3}, \lambda_{b-1}]$ satisfying certain constraints. Note that λ_{b-1} is estimated from the last iteration. If λ_b^- and λ_b^+ cannot be found, the uniform model will be adopted for the current and following quantization bin(s).

In the quantization bin $b < 0$

For the quantization bin $b < 0$, the procedure of building the dithering signal model is very similar to that for quantization bin $b > 0$. For the sake of simplicity, we do not present the details here, but only give the equations for searching λ_b^- and λ_b^+, that are:

$$\lambda_b^- = \underset{10^{-3}\leq\lambda\leq\lambda_{b+1}}{\arg\min} \quad \lambda$$
$$\text{subject to: } \begin{cases} P_m^o\left(N=-\lfloor\frac{\mathbf{Q}_{r,c}}{2}\rfloor|Y=b\right)\times M_b \geq \frac{M_{b-1}}{\mathbf{Q}_{r,c}} \\ P_m^o\left(N=\lfloor\frac{\mathbf{Q}_{r,c}}{2}\rfloor|Y=b\right)\times M_b \leq P_m^o(N=-\lfloor\frac{\mathbf{Q}_{r,c}}{2}\rfloor|Y=b+1)\times M_{b+1}, \end{cases} \tag{5.23}$$

and

$$\lambda_b^+ = \underset{10^{-3}\leq\lambda\leq\lambda_{b+1}}{\arg\max} \quad \lambda$$
$$\text{subject to: } \begin{cases} P_m^o\left(N=-\lfloor\frac{\mathbf{Q}_{r,c}}{2}\rfloor|Y=b\right)\times M_b \geq \frac{M_{b-1}}{\mathbf{Q}_{r,c}} \\ P_m^o\left(N=\lfloor\frac{\mathbf{Q}_{r,c}}{2}\rfloor|Y=b\right)\times M_b \leq P_m^o\left(N=-\lfloor\frac{\mathbf{Q}_{r,c}}{2}\rfloor|Y=b+1\right)\times M_{b+1}, \end{cases} \tag{5.24}$$

when $\mathbf{Q}_{r,c}$ is an odd number, or:

$$\lambda_b^- = \underset{10^{-3}\leq\lambda\leq\lambda_{b+1}}{\arg\min} \quad \lambda$$
$$\text{subject to: } \begin{cases} P_m^e\left(N=-\frac{\mathbf{Q}_{r,c}}{2}+1|Y=b\right)\times M_b \geq P_m^e\left(N=-\frac{\mathbf{Q}_{r,c}}{2}|Y=b\right)\times M_b + \frac{M_{b-1}}{2\mathbf{Q}_{r,c}} \\ P_m^e\left(N=\frac{\mathbf{Q}_{r,c}}{2}-1|Y=b\right)\times M_b \leq P_m^e\left(N=\frac{\mathbf{Q}_{r,c}}{2}|Y=b\right)\times M_b \\ \qquad\qquad +P_m^e\left(N=-\frac{\mathbf{Q}_{r,c}}{2}|Y=b+1\right)\times M_{b+1}, \end{cases} \tag{5.25}$$

and

$$\lambda_b^+ = \underset{10^{-3}\leq\lambda\leq\lambda_{b+1}}{\arg\max} \quad \lambda$$
$$\text{subject to: } \begin{cases} P_m^e\left(N = -\frac{\mathbf{Q}_{r,c}}{2} + 1 | Y = b\right) \times M_b \geq P_m^e\left(N = -\frac{\mathbf{Q}_{r,c}}{2} | Y = b\right) \times M_b + \frac{M_{b-1}}{2\mathbf{Q}_{r,c}} \\ P_m^e\left(N = \frac{\mathbf{Q}_{r,c}}{2} - 1 | Y = b\right) \times M_b \leq P_m^e\left(N = \frac{\mathbf{Q}_{r,c}}{2} | Y = b\right) \times M_b \\ \qquad\qquad + P_m^e\left(N = -\frac{\mathbf{Q}_{r,c}}{2} | Y = b+1\right) \times M_{b+1}, \end{cases} \tag{5.26}$$

when $\mathbf{Q}_{r,c}$ is an even number.

Chapter 6

JPEG Image Quality Enhancement and Anti-Forensics Using a Sophisticated Image Prior Model

Contents

6.1 Introduction and Motivation . . . 106
6.2 JPEG Image Quality Enhancement . . . 107
6.2.1 Prior Art . . . 107
6.2.2 Proposed Method . . . 108
6.3 Non-Parametric DCT Histogram Smoothing . . . 110
6.4 Proposed JPEG Anti-Forensics . . . 116
6.5 Summary . . . 121

In this chapter, we first propose a JPEG image quality enhancement method under the Expected Patch Log Likelihood (EPLL) framework. The rich GMM is used as the image prior model. From the quality enhanced JPEG image and based on calibration, a new, non-parametric method to DCT histogram smoothing is proposed without any histogram statistical model. For JPEG anti-forensic purposes, we further propose to optimize an objective function considering both anti-forensic terms and a natural image statistical model. We also provide discussions and experimental comparisons with the state-of-the-art JPEG anti-forensic methods as well as our two JPEG anti-forensic methods respectively proposed in Chapters 4 and 5.

A paper describing the proposed methods was published in an international conference [Fan+13b]. In the future, we plan to extend this work and probably share the relevant source code freely online.

6.1 Introduction and Motivation

In Chapters 4 and 5, we propose JPEG anti-forensic methods based on TV, which enjoys its popularity in various image restoration applications such as denoising, deblurring and inpainting. In image restoration, the MAP (or one of its variants, or approximate MAP) estimate, which consists of a likelihood term and an image prior, is often used [ZW11]. In the proposed TV-based JPEG deblocking framework described in Chapter 4, the TV can be taken as a simple but effective image prior; however, there is no likelihood term considering the JPEG compression process.

We have seen the effectiveness of introducing the TV from image restoration to JPEG anti-forensics in Chapters 4 and 5. A natural follow-up work of JPEG anti-forensics is to study the JPEG compression process and consider more sophisticated image prior models. By using image restoration techniques, we expect to improve the visual quality of the JPEG image. At the same time, the spatial-domain blocking artifacts and the DCT-domain quantization artifacts should be to some extent mitigated, though they may still be detectable by forensic detectors. The image restoration step not only helps us to obtain a good visual quality of the processed JPEG image, but also reduces the cost of removing the JPEG artifacts which have been partly removed during the restoration process. The following step goes to the integration of anti-forensic strategies/terms to further bring the forensic feature output of the anti-forensic JPEG image into the normal range of the original image. During this step, we may have to sacrifice some image quality of the processed image, for anti-forensic purposes. However, we hope the previous image restoration step will guarantee us a good visual quality of the final anti-forensic JPEG image.

According to the above idea of leveraging on image restoration concepts/methods to design JPEG anti-forensics, in this chapter, we design another JPEG anti-forensic method as follows:

- We firstly study the JPEG compression process and approximate it as a spatial-domain compression noise addition process, following the common practice in JPEG post-processing for quality enhancement purposes [RS05, SC07]. The 0-mean multivariate Gaussian distribution is used to model the compression noise. We obtain the covariance matrix by performing a learning procedure on original and JPEG images. As to the image prior, we use the rich GMM model under the EPLL framework [ZW11]. Therefore, an approximate MAP estimate is formed to seek the quality enhanced JPEG image.

- Unlike Stamm *et al.*'s dithering based DCT histogram smoothing [Sta+10a, SL11] or the perceptual DCT histogram smoothing described in Chapter 5, we propose a non-parametric DCT histogram smoothing method based on calibration [FGH02].

- At last, an image fidelity term, the EPLL term with the GMM model, and some JPEG anti-forensic terms inspired by forensic detectors are formulated together as a minimization problem. This is for the consideration of both the forensic undetectability and the visual quality of the processed image. The anti-forensic JPEG image is generated by solving this optimization problem.

The remainder of this chapter is organized as follows. Section 6.2 briefly reviews related work on JPEG image post-processing for quality enhancement purposes, and thereafter proposes a new JPEG image quality enhancement method under the EPLL framework with the GMM as the image prior model. The proposed non-parametric DCT histogram smoothing method based on calibration is described in Section 6.3. Section 6.4 presents the optimization problem of removing the introduced unnatural spatial-domain noise during DCT histogram smoothing, as well as fooling existing JPEG forensic detectors. We provide some discussions and summarize this chapter in Section 6.5.

6.2 JPEG Image Quality Enhancement

6.2.1 Prior Art

The restoration of JPEG image can be formulated as estimating an image $\hat{\mathbf{x}}$ given the JPEG image $\mathbf{y}$ and its corresponding quantization table $\mathbf{Q}$, using the prior information of natural uncompressed images and with the knowledge of JPEG compression process. In other words, given a JPEG compressed image $\mathbf{y}$ (pixel values in vectorized form), the objective is to obtain a restored image $\hat{\mathbf{x}}$, which is the most likely to be the original uncompressed image. A common practice in image restoration is the adoption of an MAP criterion to seek the solution:

$$\hat{\mathbf{x}} = \arg\max_{\mathbf{x}} p(\mathbf{x}|\mathbf{y}) = \arg\max_{\mathbf{x}} p(\mathbf{y}|\mathbf{x})p(\mathbf{x}), \tag{6.1}$$

where $p(\mathbf{y}|\mathbf{x})$ is the likelihood term describing the distortion caused by JPEG compression, and $p(\mathbf{x})$ is a prior term representing natural image statistics. The maximization problem in Eq. (6.1) is equivalent to the following minimization problem which is derived by taking the negative log function of the cost function in Eq. (6.1):

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} \left\{ -\log\, p(\mathbf{y}|\mathbf{x}) - \log\, p(\mathbf{x}) \right\}. \tag{6.2}$$

JPEG compression can be modeled as a process to add spatial-domain *compression noise* $\mathbf{n}_q$ to the original uncompressed image, then $\mathbf{y} = \mathbf{x} + \mathbf{n}_q$. A common assumption is to treat $\mathbf{n}_q$ as a random quantity; meanwhile $\mathbf{n}_q$ and $\mathbf{x}$ are considered to be independent [RS05, SC07]. Then we have:

$$p(\mathbf{y}|\mathbf{x}) = p(\mathbf{x} + \mathbf{n}_q|\mathbf{x}) = p(\mathbf{n}_q|\mathbf{x}) = p(\mathbf{n}_q). \tag{6.3}$$

For the compression noise model $p(\mathbf{n}_q)$, a 0-mean Gaussian model with covariance matrix $\mathbf{C}^q$, is used in [RS05, SC07]. The (r, c)-th entry of $\mathbf{C}^q$ can be expressed by the corresponding entry in quantization matrix $\mathbf{Q}$ as $\mathbf{C}^q_{r,c} = (\mathbf{Q}_{r,c})^2/12$ [RS05], under the assumption that the quantization noise in the DCT domain is uniformly distributed within the corresponding quantization bin. Sun and Cham [SC07] experimentally demonstrate that the $\mathbf{C}^q$ calculated in this way outperforms the actual covariance matrix calculated using the original and JPEG images in terms of the image quality of the post-processed JPEG image. This somewhat

strange result may be due to the strong assumption made in Eq. (6.3) that $\mathbf{n}_q$ and $\mathbf{x}$ are independent [SC07]. As to the image prior model $p(\mathbf{x})$, Robertson and Stevenson [RS05] use the Huber-Markov Random Field (HMRF). Sun and Cham [SC07] use a higher order Markov random field based on the Fields of Experts (FoE) framework which is able to achieve a higher visual quality of the processed JPEG image than the HMRF model.

In [ZW11], Zoran and Weiss propose a novel image restoration framework which has excellent performance in various applications such as denoising, deblurring, and inpainting. Compared with the traditional MAP-based image restoration framework, the new idea is to maximize the EPLL meanwhile considering the image fidelity with respect to the observed degraded image $\mathbf{y}$. In other words, the objective is to find $\hat{\mathbf{x}}$ in which *every* patch is *likely* under the patch prior. Therefore, the minimization problem is formulated as:

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} \left\{ -\log p(\mathbf{y}|\mathbf{x}) - \sum_i \log p(\mathbf{P}^i\mathbf{x}) \right\}, \tag{6.4}$$

where $\mathbf{P}^i$ is a matrix extracting the i-th *overlapping* patch of size $s \times s$ from the image. The above cost function has a similar form to an MAP estimate which has a likelihood term and an image prior term. The difference here is that the EPLL sums over the log probabilities of all overlapping patches, other than the log probability of a full image. As to the image prior, Zoran and Weiss [ZW11] use the finite GMM model with J mixture components giving the log probability of a given patch $\mathbf{P}^i\mathbf{x}$ as follows:

$$\log p(\mathbf{P}^i\mathbf{x}) = \log \sum_{j=1}^{J} w_j \left\{ \frac{1}{\sqrt{(2\pi)^{s^2}|\mathbf{C}^j|}} \exp\left(-\frac{1}{2}(\mathbf{P}^i\mathbf{x} - \boldsymbol{\mu}^j)^T(\mathbf{C}^j)^{-1}(\mathbf{P}^i\mathbf{x} - \boldsymbol{\mu}^j) \right) \right\}. \tag{6.5}$$

For the j-th mixture component, w_j, $\boldsymbol{\mu}^j$ and $\mathbf{C}^j$ are the mixing weight, the mean and covariance matrix of the corresponding Gaussian distribution, respectively. These parameters can be learned using the Expectation Maximization (EM) algorithm on a set of patches extracted from original natural images.

6.2.2 Proposed Method

In our work, we follow Zoran and Weiss' [ZW11] EPLL framework with GMM as the image prior model. We choose the patch size of 8×8, the same as that of JPEG blocks. The main motivation of choosing this patch-wise prior is to facilitate and speed up the learning and restoration processes, compared with [RS05, SC07]. We still use the 0-mean multivariate Gaussian [RS05] to model the spatial-domain compression noise $\mathbf{n}_q$, which is assumed to be independent with the original image $\mathbf{x}$. The difference is that for a better quality of the recovered image we need to take into account all the *overlapping* patches instead of the *non-overlapping* JPEG blocks. Therefore, we consider $8 \times 8 = 64$ kinds of overlapping patches according to their relative position with respect to JPEG blocks. The minimization problem

is then formulated as:

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{u}} \left\{ \sum_{k=1}^{64} \sum_{\mathbf{P}^i \in \mathcal{S}_k} \frac{1}{2} \left(\mathbf{P}^i(\mathbf{y}-\mathbf{u})\right)^t (\mathbf{C}^k)^{-1} \mathbf{P}^i(\mathbf{y}-\mathbf{u}) - \sum_i \log p(\mathbf{P}^i\mathbf{u}) \right\}, \tag{6.6}$$

where $\mathcal{S}_k$ is the k-th set of matrices which extract patches having the same relative positions with respect to JPEG blocks, and $\mathbf{C}^k$ is the covariance matrix for modeling the compression noise of the k-th group of patches. Besides, $\log p(\mathbf{P}^i\mathbf{u})$ is derived by replacing $\mathbf{P}^i\mathbf{x}$ in Eq. (6.5) by $\mathbf{P}^i\mathbf{u}$. The optimization of Eq. (6.6) can be solved using an approximate MAP estimation procedure giving a Wiener filter solution (for more details, please refer to [ZW11]). At last, a QCS projection is used to assure that the DCT coefficient of the processed image is within the same quantization bin as that of the JPEG image. If a DCT coefficient goes outside its original quantization bin, it will be modified to the boundary value of the quantization bin.

Practically, we learn a GMM prior with 200 mixture components from $338 \times 6000 = 2028000$ randomly sampled 8×8 patches on UCIDLearn dataset (see Section 2.3.1 for more descriptions about the datasets). For JPEG blocks, Robertson and Stevenson [RS05] propose to calculate the covariance matrix $\mathbf{C}^q$ (*i.e.*, one of the $\mathbf{C}^k$ in Eq. (6.6)) from the quantization table $\mathbf{Q}$. Sun and Cham [SC07] also experimentally prove its superiority over the actual compression noise covariance matrix. In the proposed optimization problem in Eq. (6.6), we need to consider 63 more kinds of 8×8 patches besides the JPEG blocks. In these patches, pixels across neighboring JPEG blocks are present in the same patch. Correlations of spatial-domain compression noise across different JPEG blocks may be difficult to calculate. Therefore, for each quality factor q, we propose to learn the 64 covariance matrices $\mathbf{C}^k$ on UCIDLearn dataset. More specifically, given a JPEG compression quality factor q, we JPEG compress all the images from UCIDLearn dataset, the actual compression noise is obtained by subtracting the JPEG image by its corresponding original image. The compression noise is thereafter used for learning the 64 covariance matrices $\mathbf{C}^k$. The learning of the GMM prior as well as the covariance matrices is performed on the 338 images from UCIDLearn dataset using the EM algorithm with unoptimized Matlab code[14].

Table 6.1 lists the PSNR results for 4 classical images and 3 quantization tables Q1, Q2 and Q3, provided by [SC07], for very low bit-rate compression. The original uncompressed image is used as the reference for calculating the PSNR values. We can see that the proposed method is very competitive in terms of PSNR gain. In our JPEG anti-forensic work, we consider the quality factors in $\{50, 51, \cdots, 95\}$, whose corresponding quantization matrices produce much higher bit-rate compression than Q1, Q2 and Q3 used in [SC07]. In practice, we found that the PSNR gain of the proposed JPEG post-processing method is slightly lower than that of Sun and Cham's [SC07] method using these quality factors. However, the proposed method only requires one step of approximate MAP estimation, which is practically around ten times faster than [SC07] using the conjugate gradient descent. As an intermediate result of our anti-forensic JPEG image creation (to be described later), the PSNR gain is already satisfying. We denote the processed JPEG image using the proposed quality enhancement method as $\hat{\mathcal{I}}^J$.

[14] Downloaded from: http://www.mathworks.com/matlabcentral/fileexchange/26184-em-algorithm-for-gaussian-mixture-model.

From the figures listed in Table 6.5, we can see that $\hat{\mathcal{I}}^J$ achieves a PSNR gain of 0.7931 dB and an SSIM gain of 0.0008 than the JPEG image $\mathcal{J}$ on average, on UCIDTest dataset.

Table 6.1: PSNR (using the original image as the reference) results for 4 classical test images and 3 different quantization tables provided in [SC07][15].

		JPEG image	FoE-based [SC07]	Proposed
Lena	Q1	30.71	31.95	**32.06**
	Q2	30.08	31.44	**31.48**
	Q3	27.45	28.83	**28.94**
Peppers	Q1	30.72	32.04	**32.09**
	Q2	30.17	**31.61**	31.59
	Q3	27.66	29.35	**29.40**
Barbara	Q1	25.95	26.65	**26.94**
	Q2	25.60	26.31	**26.56**
	Q3	24.05	24.86	**25.00**
Baboon	Q1	24.32	24.77	**24.84**
	Q2	24.14	24.62	**24.68**
	Q3	22.14	**22.61**	**22.61**

After the image quality enhancement processing of $\mathcal{J}$, $\hat{\mathcal{I}}^J$ has already partly recovered the lost information during JPEG compression, however the DCT-domain information has not been very well recovered. A sample DCT histogram of $\hat{\mathcal{I}}^J$, which still has the comb-like quantization artifacts, is shown in Figure 6.2-(b). A natural following step goes to the DCT histogram smoothing. Other than using the global Laplacian model in [Sta+10a, SL11] or the local Laplacian model in Chapter 5, a new, non-parametric approach based on calibration is proposed without the consideration of any statistical model in Section 6.3.

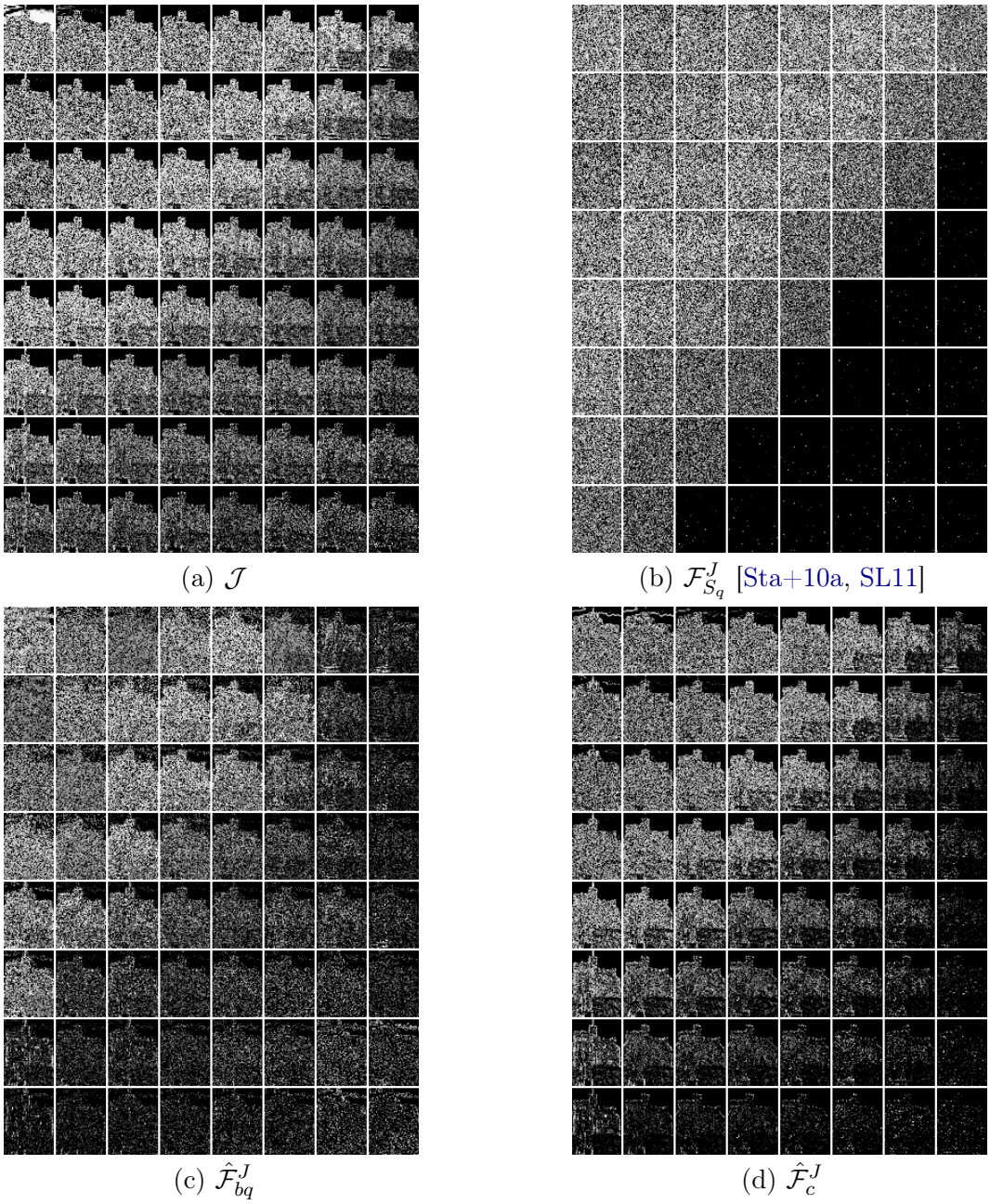
6.3 Non-Parametric DCT Histogram Smoothing

In Section 5.2.2.1, we have discussed the disadvantages of using the Laplacian distribution to model the DCT coefficients in AC subbands, which is the basic assumption of Stamm *et al.*'s [Sta+10a, SL11] dithering based JPEG anti-forensic method. For natural uncompressed images, the DCT histograms for AC components usually have a much higher peak than the Laplacian distribution. This point can also be reflected by the big kurtosis value of AC components, which is often larger than 6 (kurtosis of the Laplacian distribution).

Besides, during the creation of Stamm *et al.*'s [Sta+10a, SL11] anti-forensic JPEG image $\mathcal{F}^J_{S_q}$, the dithering signal is generated depending on the estimated Laplacian parameter using

[15] Note that in this table the PSNR results of Sun and Cham's method are slightly different from the values listed in [SC07]. It is because we rounded the pixel values to integers within $[0, 255]$ before calculating the PSNR value, while in [SC07] the PSNR is computed on floating-point "pixel values".

the MLE [PR99]. The dithering process is thereafter conducted by adding the signal to the quantized DCT coefficient *randomly* without any consideration of the local information of the image in the spatial domain. This constitutes another disadvantage of the method that would lead to a poor visual quality of $\mathcal{F}^J_{S_q}$ [VTT11]. An example can be seen in Figure 6.1-(b), comparing with -(a).



(a) $\mathcal{J}$ (b) $\mathcal{F}^J_{S_q}$ [Sta+10a, SL11]

(c) $\hat{\mathcal{F}}^J_{bq}$ (d) $\hat{\mathcal{F}}^J_{c}$

Figure 6.1: Comparison of DCT-domain quantization noise stacked by the spatial-domain location and by the 64 DCT frequencies. Results are obtained from a UCID image and its JPEG compressed version with quality factor 50. For a better visibility of the noise images, we have taken logarithm of the noise and afterwards carried out a normalization.

In Chapter 5, the proposed perceptual DCT histogram smoothing method considers an adaptive local dithering method which combines the Laplacian distribution and the uniform distribution. The DCT histogram mapping using the Hungarian algorithm explicitly considers the visual quality loss due to the DCT coefficient modification. Therefore, we can see that $\hat{\mathcal{F}}_{bq}$ outperforms $\mathcal{F}_{S_q}$ [Sta+10a, SL11] in terms of both image quality (see Table 5.4) and KL divergence (see Table 5.2), with the original image $\mathcal{I}$ as the reference. Moreover, an example of the DCT-domain quantization noise of $\hat{\mathcal{F}}_{bq}$ can be found in Figure 6.1-(c). It can be seen that the spatial-domain local information of the image is to some extent correctly reflected in the DCT-domain quantization noise distribution.

Both of the DCT histogram smoothing methods for generating $\mathcal{F}_{S_q}$ [Sta+10a, SL11] and $\hat{\mathcal{F}}_{bq}$ rely on a statistical model for the DCT coefficients. In this chapter, we aim to find a new, non-parametric approach to DCT histogram smoothing. We wish to be able to estimate the DCT-domain quantization noise, from the restored JPEG image $\hat{\mathcal{I}}^J$. To this end, we get inspired by the so-called *calibration* method which was initially used in steganalysis [FGH02].

Before presenting the algorithm, we hereby provide a brief introduction of the calibration method. It was introduced by Fridrich *et al.* [FGH02] for steganalysis purposes to estimate the DCT histogram of the cover-image from the stego-image. It is also used by Lai and Böhme [LB11] for JPEG forensic purposes. The main idea is to break the JPEG block structure by cropping the first 4 pixels of the JPEG image both horizontally and vertically. The calibrated image shares some DCT-domain statistical similarities with the original uncompressed image. Therefore, we expect to estimate the DCT-domain quantization noise by comparing the calibrated image and its JPEG compressed version. Based on the above analysis, we propose a new, non-parametric DCT histogram smoothing method based on *relaxed* calibration as described in Algorithm 6.1.

Algorithm 6.1 Non-parametric DCT histogram smoothing procedure.

Require: JPEG image $\mathcal{J}$ with quality factor q

1: Recover $\hat{\mathcal{I}}^J$ from $\mathcal{J}$ by solving the optimization problem in Eq. (6.6).
2: Crop $\hat{\mathcal{I}}^J$ by 1 pixel[16] both horizontally and vertically to obtain $\hat{\mathcal{I}}_c$.
3: JPEG compress $\hat{\mathcal{I}}_c$ with quality factor q to obtain $\hat{\mathcal{J}}_c$.
4: Subtract the DCT coefficients of $\hat{\mathcal{I}}_c$ by those of $\hat{\mathcal{J}}_c$, so as to estimate the DCT quantization noise $\hat{\mathcal{N}}_q$.
5: Add $\hat{\mathcal{N}}_q$ to $\hat{\mathcal{I}}^J$ in the DCT domain, so as to create $\hat{\mathcal{F}}_c^J$ with smoothed DCT histograms.
6: **return:** $\hat{\mathcal{F}}_c^J$

An example of DCT quantization noise of $\hat{\mathcal{F}}_c^J$, which is generated by the proposed calibration based DCT histogram smoothing method, can be found in Figure 6.1-(d). It can be seen that the proposed method can also to some extent reproduce the DCT-domain quantization noise which is more or less related to the local spatial-domain information of the image. On UCIDTest dataset, we compute the KL divergence value between $\mathcal{I}$ and $\mathcal{F}_{S_q}^J$, and that between $\mathcal{I}$ and $\hat{\mathcal{F}}_c^J$ for all DCT subbands. The *difference* between these two KL divergence

[16] Here we only crop the image by 1 pixel instead of 4 pixels as in classical image calibration, it is empirically determined in order to attain higher image quality for $\hat{\mathcal{F}}_c^J$, the output of Algorithm 6.1.

values is reported in Table 6.2. In Stamm *et al.*'s [Sta+10a, SL11] DCT histogram smoothing method, the subband where all the DCT coefficients are quantized to 0 is untouched. For a fair comparison for $\mathcal{F}^J_{S_q}$, these subbands were not counted in the comparison. From Table 6.2, we can see that $\hat{\mathcal{F}}^J_c$ performs consistently better than $\mathcal{F}^J_{S_q}$ in AC components. As shown in Figure 6.1-(b) and in the beginning of this section, we analyze that the quality degradation of $\mathcal{F}^J_{S_q}$ is due to the randomness of adding dithering signal without any consideration of the image spatial-domain information. By comparing the results shown in Table 6.5 and those in Table 5.4, we can see that not only $\hat{\mathcal{F}}^J_c$ achieves a lower KL divergence value, but also the average PSNR and SSIM of $\hat{\mathcal{F}}^J_c$ have gains of 1.91 dB and 0.0120, respectively, compared with $\mathcal{F}^J_{S_q}$ [Sta+10a, SL11].

Table 6.2: The difference of the KL divergence between $\mathcal{I}$ and $\mathcal{F}^J_{S_q}$ [Sta+10a, SL11], and that between $\mathcal{I}$ and $\hat{\mathcal{F}}^J_c$ for all 64 DCT subbands. The average difference value over all subbands is 0.0439 with standard deviation 0.0214. For a fair comparison for $\mathcal{F}^J_{S_q}$, the subbands, whose DCT coefficients are all quantized to 0 in the JPEG image $\mathcal{J}$, are not counted. Results are obtained on UCIDTest dataset.

r \ c	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
1	-0.0060	0.0065	0.0190	0.0369	0.0549	0.0728	0.0796	0.0474
2	0.0069	0.0227	0.0324	0.0381	0.0437	0.0719	0.0564	0.0117
3	0.0289	0.0298	0.0364	0.0416	0.0526	0.0636	0.0604	0.0081
4	0.0300	0.0351	0.0399	0.0388	0.0514	0.0796	0.0579	0.0040
5	0.0436	0.0412	0.0495	0.0571	0.0573	0.0865	0.0687	0.0161
6	0.0508	0.0469	0.0575	0.0503	0.0568	0.0728	0.0564	0.0319
7	0.0806	0.0629	0.0633	0.0630	0.0585	0.0613	0.0590	0.0415
8	0.0598	0.0401	0.0314	0.0243	0.0231	0.0138	0.0140	0.0186

Besides the comparison with Stamm *et al.*'s dithering based DCT histogram smoothing method [Sta+10a, SL11], we also would like to compare the proposed calibration based DCT histogram smoothing method proposed in this section with our other perceptual DCT histogram smoothing method proposed in Section 5.2.2 in the following[17].

From Table 6.3, we can see that the JPEG quality enhancement method proposed in Section 6.2 indeed has advantages in visual quality over the processed image $\hat{\mathcal{F}}^J_b$ using TV-based JPEG deblocking method (see Section 5.2.1). Yet, in the DCT domain, $\hat{\mathcal{F}}^J_b$ is able to recover more information than $\hat{\mathcal{I}}^J$ (see the KL divergence difference in Table 6.3). Comparison of example DCT histograms can also be seen in Figures 6.2-(a) and -(b). The remaining DCT-domain quantization artifacts in $\hat{\mathcal{F}}^J_b$ are less obvious than those in $\hat{\mathcal{I}}^J$.

With another step of DCT histogram smoothing considering the perceptual quality loss due to the DCT coefficient modification (see Section 5.2.2), $\hat{\mathcal{F}}^J_{bq}$ is able to achieve smoothed DCT histogram without much visual quality sacrifice. On the other hand, though $\hat{\mathcal{I}}^J$ has an

[17] Our research work published in [Fan+13a] (also see Chapter 4) and in [Fan+13b] (also see this chapter) was conducted almost in parallel. The work published in [Fan+14] (also see Chapter 5) was carried out later. The comparison presented in this chapter among the three methods is new, and not in any published papers.

Table 6.3: The 2nd and 3rd columns respectively list the mean (μ^{kl}) and standard deviation (σ^{kl}) of the difference of the KL divergence of the DCT histogram between different images. The 4th and 5th columns report the mean (μ^{mse}) and standard deviation (σ^{mse}) of difference of the MSE of the *unnormalized* DCT histogram between different image pairs. All the 64 subbands are considered, no matter whether the DCT coefficients are all quantized to 0 in the JPEG image $\mathcal{J}$. The 6th and 7th columns show the PSNR and SSIM difference. All evaluation metrics are calculated using the original image $\mathcal{I}$ as the reference. The anti-forensic JPEG image is expected to have low KL divergence and MSE values of the DCT histogram and high PSNR and SSIM values. Results are obtained on UCIDTest dataset.

	μ^{kl}	σ^{kl}	μ^{mse}	σ^{mse}	PSNR	SSIM
$\hat{\mathcal{F}}_b^J$ *vs.* $\hat{\mathcal{I}}^J$	-0.4368	0.1481	-17863.56	15711.74	-1.1525	-0.0036
$\hat{\mathcal{F}}_{bq}^J$ *vs.* $\hat{\mathcal{F}}_c^J$	-0.0239	0.0431	-2195.72	2686.60	0.6717	-0.0004
$\mathcal{F}_0^J$ *vs.* $\mathcal{F}_1^J$	-0.0041	0.0328	584.18	980.70	0.2246	0.0011
$\mathcal{F}^J$ *vs.* $\mathcal{F}_1^J$	-0.0202	0.0295	-143.63	745.90	0.7287	0.0034

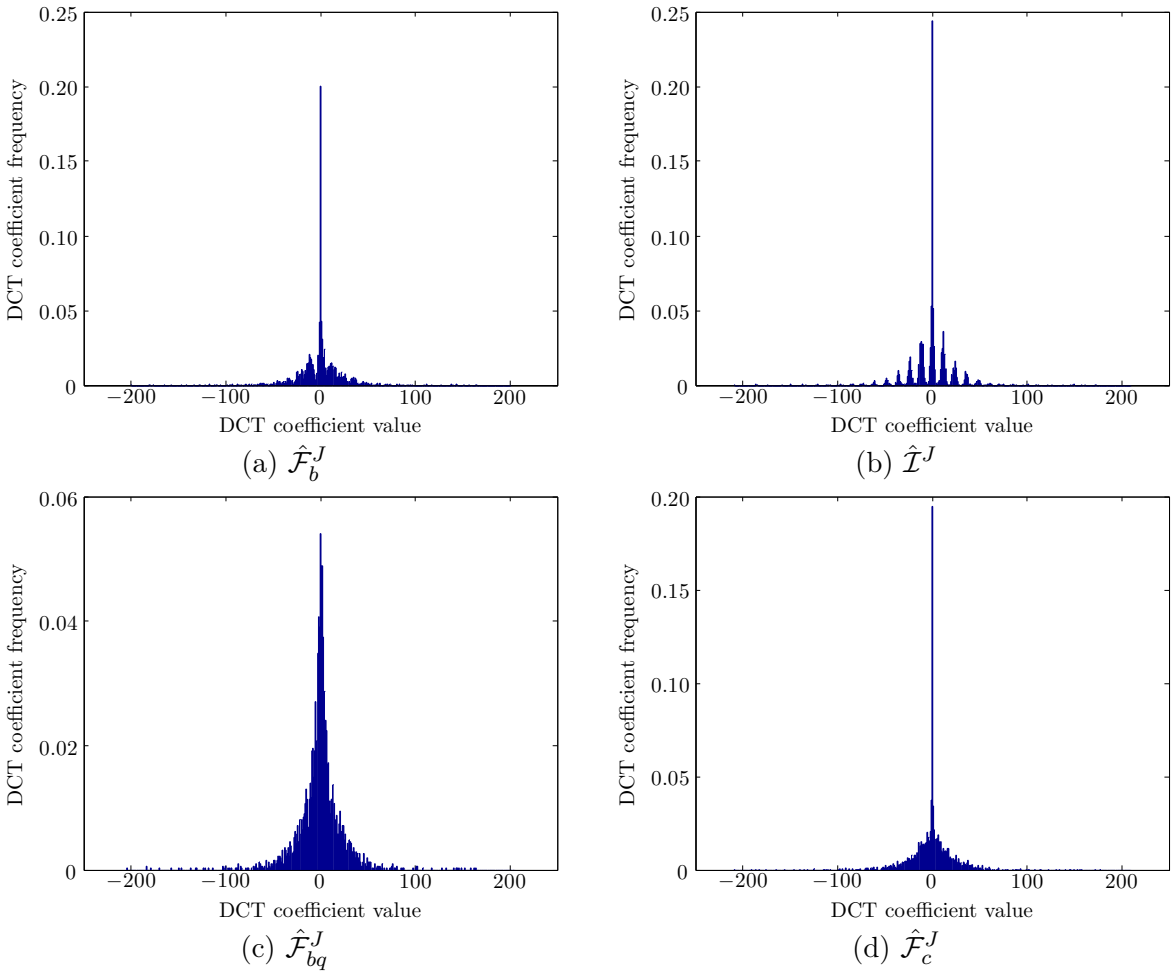


Figure 6.2: Example $(2, 2)$ subband DCT histograms of $\hat{\mathcal{F}}_b^J$, $\hat{\mathcal{I}}^J$, $\hat{\mathcal{F}}_{bq}^J$ and $\hat{\mathcal{F}}_c^J$. The corresponding JPEG image $\mathcal{J}$ is compressed with quality factor 50 from a UCID image.

even higher visual quality than the JPEG image $\mathcal{J}$, extra noise is later introduced in $\hat{\mathcal{F}}_c^J$, by adding the estimated DCT-domain quantization noise estimated based on calibration directly back to $\hat{\mathcal{I}}^J$. This results in that $\hat{\mathcal{F}}_c^J$ achieves a lower image quality than $\hat{\mathcal{F}}_{bq}^J$. On the DCT histogram recovery side, $\hat{\mathcal{F}}_{bq}^J$ is able to achieve a lower KL divergence value than $\hat{\mathcal{F}}_c^J$ on average over all 64 DCT subbands. However, Table 6.4 shows that in fact $\hat{\mathcal{F}}_c^J$ performs better than $\hat{\mathcal{F}}_{bq}^J$ in the low-frequency DCT subbands. Though $\hat{\mathcal{F}}_c^J$ does not outperform $\hat{\mathcal{F}}_{bq}^J$ in general, we can conclude that the very simple slight cropping helps $\hat{\mathcal{I}}_c$ get a quite good estimation of the original DCT histogram of $\mathcal{I}$, especially in the low-frequency DCT subbands.

Table 6.4: The difference of the KL divergence between $\mathcal{I}$ and $\hat{\mathcal{F}}_{bq}^J$, and that between $\mathcal{I}$ and $\hat{\mathcal{F}}_c^J$ for all 64 DCT subbands. The average difference value over all subbands is -0.0239 with standard deviation 0.0431. All the subbands are considered, no matter whether the DCT coefficients are all quantized to 0 in the JPEG image $\mathcal{J}$. Results are obtained on UCIDTest dataset.

r \ c	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
1	-0.0081	-0.0001	0.0096	0.0121	0.0075	0.0126	0.0171	-0.0265
2	0.0037	0.0081	0.0127	0.0025	-0.0013	0.0201	0.0056	-0.0393
3	0.0171	0.0120	0.0098	-0.0036	-0.0062	-0.0050	-0.0099	-0.0663
4	0.0133	0.0046	-0.0013	-0.0110	-0.0194	-0.0086	-0.0097	-0.0708
5	0.0106	0.0022	-0.0037	-0.0159	-0.0291	-0.0237	-0.0591	-0.1173
6	0.0090	0.0114	-0.0104	-0.0179	-0.0323	-0.0253	-0.0692	-0.1229
7	0.0353	0.0223	-0.0124	-0.0079	-0.0660	-0.0690	-0.0936	-0.1067
8	-0.0050	-0.0332	-0.0730	-0.0720	-0.1385	-0.1372	-0.1235	-0.0357

In all, the perceptual DCT histogram smoothing after TV-based JPEG deblocking proposed in Section 5.2.2 outperforms the proposed calibration based DCT histogram smoothing method proposed in this section, in both overall DCT histogram recovery and visual quality of the processed image. However, also as discussed in Section 5.3.3, the perceptual DCT histogram smoothing method proposed in Section 5.2.2 is the bottleneck of the computation cost of the proposed JPEG anti-forensic method in Chapter 5. As to the calibration based DCT histogram smoothing method, it is much less computationally demanding for cropping, JPEG compression and computing/adding the DCT-domain quantization noise.

Furthermore, some possible improvements over the method proposed in this section are as follows. In order to decrease the KL divergence values in the high-frequency DCT subbands for $\hat{\mathcal{F}}_c^J$, we could conduct a "clever cropping" for the calibration. For example, the quantization noise of different subbands can be estimated by cropping different "optimal" numbers of pixels in the calibration. Moreover, the image quality of $\hat{\mathcal{F}}_c^J$ could also be improved by conducting a similar procedure to the DCT mapping described in Section 5.2.2.3 in the perceptual DCT histogram smoothing method. A hybrid method combining the calibration based and the perceptual DCT histogram smoothing methods is likely to further push the performance of JPEG anti-forensics towards a better DCT histogram recovery as well as a higher image quality of the anti-forensic JPEG image, compared with the proposed methods in this thesis.

6.4 Proposed JPEG Anti-Forensics

As analyzed in Section 6.3, we are aware that the DCT-domain quantization noise estimation based on calibration cannot be very accurate. The injection of $\hat{\mathcal{N}}_q$ must have introduced some extra unnatural spatial-domain noise to $\hat{\mathcal{I}}^J$. In this section, we will focus on the denoising on $\hat{\mathcal{F}}_c^J$ as well as JPEG anti-forensics against the detectors. For convenience, the spatial-domain noise present in $\hat{\mathcal{F}}_c^J$ is assumed to be additive white Gaussian noise, with standard deviation σ_n. The forensic detectors in consideration are K_F [FD03], K_V [Val+11] and K_L [LB11][18].

We still follow Zoran and Weiss' [ZW11] EPLL based image restoration framework, yet adding some terms for JPEG anti-forensic purposes. The minimization problem we propose to solve is formulated as follows:

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{u}} \left\{ \frac{\lambda}{2}\|\mathbf{u}-\mathbf{y}\|^2 + \alpha \times \iota(\mathbf{u}) + \beta \sum_{k=1}^{28}\sum_{c=0}^{7} \left|\nu_k(\mathbf{u}_c) - \hat{\sigma}_k^2\right| + \sum_i \frac{\gamma}{2}\|\mathbf{P}^i\mathbf{u} - \mathbf{z}^i\|^2 - \log p(\mathbf{z}^i) \right\}, \quad (6.7)$$

where λ, α, β and γ are regularization parameters, $\mathbf{u}_c$ is the calibrated image obtained by cropping $\mathbf{u}$ by c pixels in both horizontal and vertical directions [FGH02, LB11], and $\{\mathbf{z}^i\}$ is a set of auxiliary variables facilitating the optimization [ZW11].

Now we explain the different terms in Eq. (6.7) in order. The first term is for the image fidelity control, which expects the obtained image is still close to the given JPEG image. The second term is designed for K_V [Val+11], and $\iota(\cdot)$ computes the TV of the image (the ℓ_1 norm of the spatial first-order derivatives). We wish the image would have low TV, in other words not very noisy. The third term is for K_L [LB11], and $\nu_k(\cdot)$ returns the variance of the k-th high-frequency subband (defined in [LB11]) of the image, while $\hat{\sigma}_k^2$ is the estimated variance of the original image from the k-th high-frequency DCT subband of $\hat{\mathcal{F}}_c^J$. As the JPEG block size is 8×8, basically we have 7 ways to crop the image along the diagonal direction by c ($c = 1, \cdots, 7$) pixel(s) in both horizontal and vertical directions. Following the main idea of K_L, 6 other detectors can be built using the similar way by comparing the variances of high-frequency subbands between the image and its calibrated version. If the image and all its 7 calibrated versions have close variances to the estimated one in all the high-frequency subbands, K_L and the other 6 potential detectors described above can be fooled. The last term is the image prior term which expects every patch is likely under the prior, because of

[18] The JPEG anti-forensic method presented in this chapter was published in a conference paper [Fan+13b] in 2013. At that time, in literature K_F, K_V and K_L are the scalar-based forensic detectors specifically considered in the state-of-the-art JPEG anti-forensics [Sta+10a, Sta+10b, SL11] and forensics [Val+11, LB11] to counter JPEG anti-forensics. We think it is possible to integrate more anti-forensic terms into the cost function in Eq. (6.7) to further improve the forensic undetectability of the anti-forensic JPEG image against more detectors. We will leave it as a future study. Different from [Fan+13b], in this section, we test the anti-forensic JPEG image against all the 8 scalar-based JPEG forensic detectors listed in Table 3.1, though some of them were not considered in [Fan+13b]. The test of the proposed anti-forensic JPEG image in this chapter against the two SVM-based detectors used in Chapter 5 and some possible improvement will also be left as a future work.

which the outputs of K_F [FD03] can be largely decreased.

The optimization problem can be solved using the "Half Quadratic Splitting" [ZW11] (see Section 2.4.3 for more details) and it consists of two sub-problems:

- $\mathbf{z}$ sub-problem, solving $\mathbf{z}$ given $\mathbf{u}$ — This is solved using the approximate MAP estimation, given the parameter γ which is related to the noise standard deviation σ_n [ZW11]. The main objective of this sub-problem is to remove the introduced extra unnatural noise in $\hat{\mathcal{F}}_c^J$, through the regularization by the natural image statistical model.

- $\mathbf{u}$ sub-problem, solving $\mathbf{u}$ given $\mathbf{z}$ — This can be solved using the subgradient method (see Section 2.4.1 for more details). JPEG anti-forensics is mainly carried out in this sub-problem, while keeping the image still close to $\mathbf{y}$.

For solving Eq. (6.7), we do 3 iterations and the parameter setting is: $\lambda = \frac{s^2}{10\sigma_n^2}$, $\alpha = \frac{s^2}{10\sigma_n^2}$, $\beta = \frac{8s^2}{10\sigma_n^2}$, $\gamma = \frac{1}{\sigma_n^2}[1, 8, 32]$, and σ_n is empirically set as $-0.1q+13$ according to the quality factor q of $\mathcal{J}$. For each iteration of constant γ value, $\mathbf{z}$ and $\mathbf{x}$ sub-problems are solved alternatively once for either problem. In practice, we found that the output of K_L [LB11] is not easy to be further decreased into the normal range. Moreover, images tend to be over-smoothed as K_F^Q [FD03] detects many '3' entries (see Section 4.2.1 for more explanation and analysis on this). We tackle this problem by adding a slight amount of white Gaussian noise in the middle of solving the $\mathbf{x}$ sub-problem during the last iteration. We set $\lambda = 0$, $\alpha = 0.01$ and $\beta = 1$ to emphasize on decreasing the K_L output. The added noise is mostly suppressed during the later subgradient iterations but can successfully reduce the occurrences of '3' in the quantization table estimation of K_F^Q [FD03].

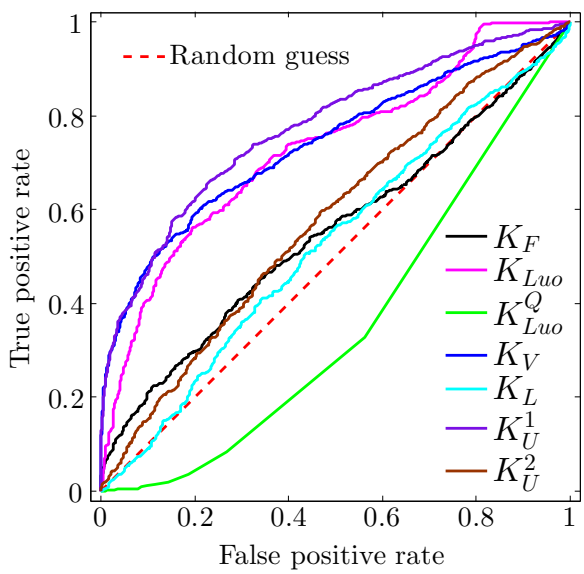


Figure 6.3: ROC curves achieved by $\mathcal{F}_1^J$ against scalar-based JPEG forensic detectors. Results are obtained on UCIDTest dataset.

Table 6.5: From the 2nd to the 8th columns, the AUC values for different kinds of images against scalar-based JPEG forensic detectors are listed; the image quality (with $\mathcal{I}$ as the reference) comparison is reported in the last 2 columns. Results are obtained on UCIDTest dataset.

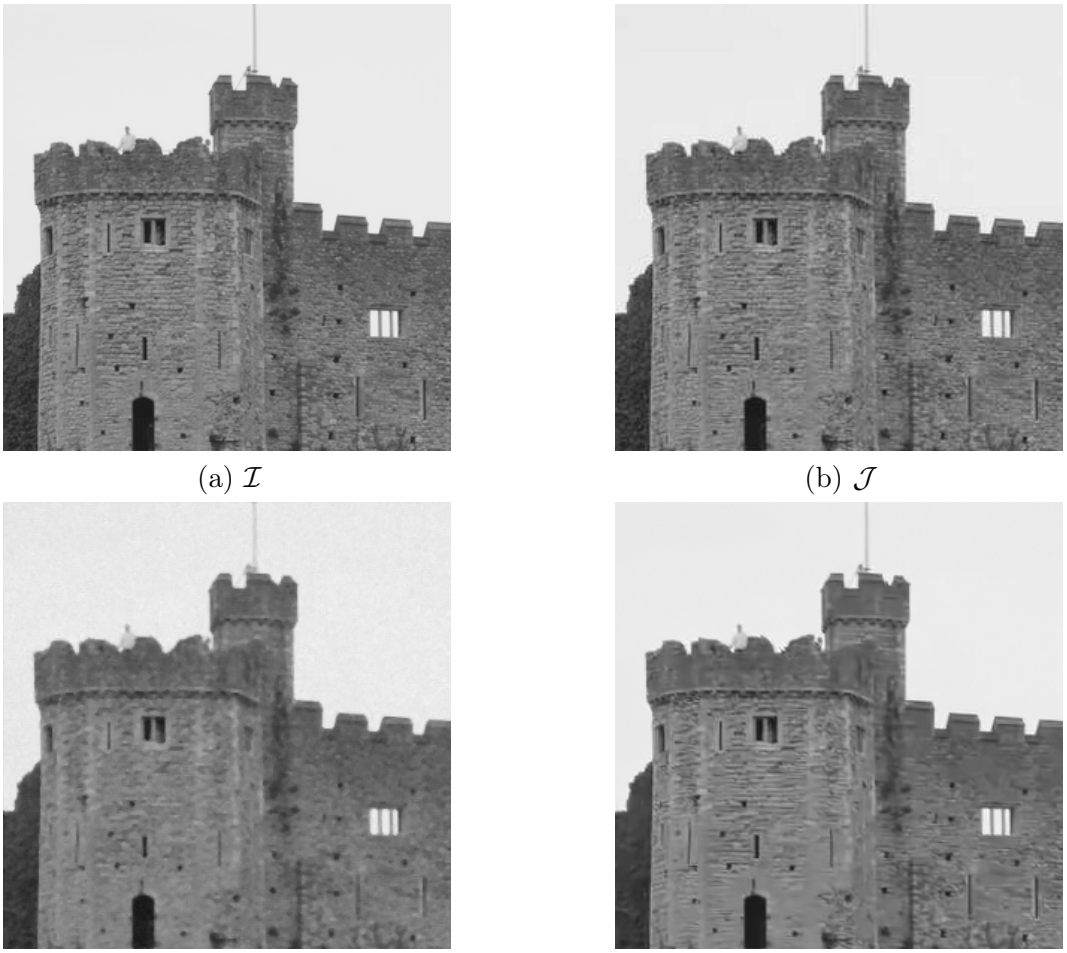
	K_F	K_{Luo}	K_{Luo}^Q	K_V	K_L	K_U^1	K_U^2	PSNR	SSIM
$\mathcal{J}$	0.9991	1.0000	0.9996	0.9976	0.9811	0.9860	0.8840	37.0999	0.9919
$\mathcal{F}_{S_qS_b}^J$	0.3783	0.0806	0.6288	0.8337	0.5338	0.6309	0.4854	30.4591	0.9509
$\hat{\mathcal{I}}^J$	0.9997	0.9982	0.9528	0.7851	0.9698	0.9878	0.8779	37.8930	0.9927
$\hat{\mathcal{F}}_c^J$	0.9994	0.8147	0.5949	0.7383	0.9868	0.9868	0.8944	35.3209	0.9876
$\mathcal{F}_1^J$	0.5522	0.7291	0.3594	0.7394[19]	0.5272	0.7750	0.5787	35.2568	0.9832
$\mathcal{F}_0^J$	0.6756	0.6046	0.5194	0.6210	0.4490	0.6772	0.5880	35.4814	0.9843
$\mathcal{F}^J$	0.5398	0.6425	0.4598	0.6159	0.4344	0.5894	0.5317	35.9855	0.9866

Let $\mathcal{F}_1^J$ denote the anti-forensic image created using the proposed method described in this chapter. Figure 6.3 shows the ROC curves of $\mathcal{F}_1^J$ against 7 scalar-based JPEG forensic detectors listed in Table 3.1. Table 6.5 reports the AUC values and average PSNR and SSIM values of different (anti-forensic/post-processed) JPEG images. We also calculated the AUC values of $\mathcal{F}_1^J$ against the above mentioned 6 potential detectors which are built in a similar way as K_L [LB11], the average AUC value is 0.4456 with the standard deviation 0.0872. Compared with the state-of-the-art anti-forensic JPEG images [Sta+10a, Sta+10b, SL11, VTT11, SS11] (see Table 5.4 for AUC, PSNR and SSIM values, and see Figures 4.2 for ROC curves), our anti-forensic JPEG image $\mathcal{F}_1^J$ achieves a better tradeoff between undetectability against existing JPEG forensic detectors and the visual quality of processed images. Take the comparison with $\mathcal{F}_{S_qS_b}$ [Sta+10a, Sta+10b, SL11] as an example, not only a slightly better overall forensic undetectability is achieved, but also $\mathcal{F}_1^J$ has a PSNR gain of 4.80 dB and an SSIM gain of 0.0323.

Figure 6.4 shows example results of (anti-forensic) JPEG images compared with the original image. As demonstrated in Figure 6.4-(d), $\mathcal{F}_1^J$ is able to keep more details such as textures and edges than $\mathcal{F}_{S_qS_b}$ [Sta+10a, Sta+10b, SL11] exampled in -(c) (please refer to the electronic version for a better visibility). Even though the quantization table estimation based detector K_F^Q [FD03] was proven not very reliable in Section 4.2.1, we still tested it on $\mathcal{F}_1^J$ on UCIDTest dataset. Among all the anti-forensic JPEG images, 94.50% of them are classified as never compressed by K_F^Q. We also examine the shape of the DCT histogram, and some example results are shown in Figure 6.5, in which no noticeable artifacts can be observed.

The last two rows of Table 6.3 show the DCT histogram recovery and image quality comparison of $\mathcal{F}_1^J$, $\mathcal{F}_0^J$, and $\mathcal{F}^J$. It can be seen that both $\mathcal{F}_0^J$ and $\mathcal{F}^J$ outperform $\mathcal{F}_1^J$.

[19] In our published paper [Fan+13b], the K_V detector is another version presented in Valenzise *et al.*'s work [Val+11], other than the version (in Valenzise *et al.*'s improved work in [VTT13]) used in this thesis. In that setting, there is a lag parameter (see [Val+11] for details) set as 5 for the re-compression quality factors for K_V. This specific setting is for the consideration that the quality factors in use in [Fan+13b] are integer multiples of 5. These setting changes result in the relatively high AUC value shown here for $\mathcal{F}_1^J$ against K_V [VTT13].



(a) $\mathcal{I}$ (b) $\mathcal{J}$

(c) $\mathcal{F}_{S_q S_b}$ (d) $\mathcal{F}_1^J$

Figure 6.4: Example results (close-up images) of $\mathcal{F}_1^J$ compared with $\mathcal{I}$, $\mathcal{J}$, and $\mathcal{F}_{S_q S_b}$ [Sta+10a, Sta+10b, SL11], where $\mathcal{J}$ is compressed with quality factor 50.

However, we know that the DCT histogram is not well smoothed in $\mathcal{F}_0^J$ (an example can be seen in Figure 4.10-(b)). Yet, its average KL divergence has an even lower value than that of $\mathcal{F}_1^J$ whose DCT histogram is explicitly smoothed by the proposed calibration based DCT histogram smoothing method described in Section 6.3. This may be explained by the fact that the smoothness of the DCT histogram is not always conveyed by the KL divergence. Therefore, we also calculate the MSE (see Eq. (2.5) for the version on images, the calculation on histograms is similar) between the unnormalized DCT histogram of the processed JPEG image and that of its corresponding original image. The mean and standard deviation of the difference of these MSE values are reported in the 4th and the 5th columns of Table 6.3. Under this evaluation metric, $\mathcal{F}_1^J$ outperforms $\mathcal{F}_0^J$.

Generally speaking, compared with $\mathcal{F}_0^J$ generated by our JPEG anti-forensic method in Chapter 4 and $\mathcal{F}^J$ generated by our JPEG anti-forensic method in Chapter 5, the anti-forensic JPEG image $\mathcal{F}_1^J$ created in this chapter does not outperform them in terms of forensic undetectability and visual quality of the processed image (see Tables 6.3 and 6.5 for the comparison,

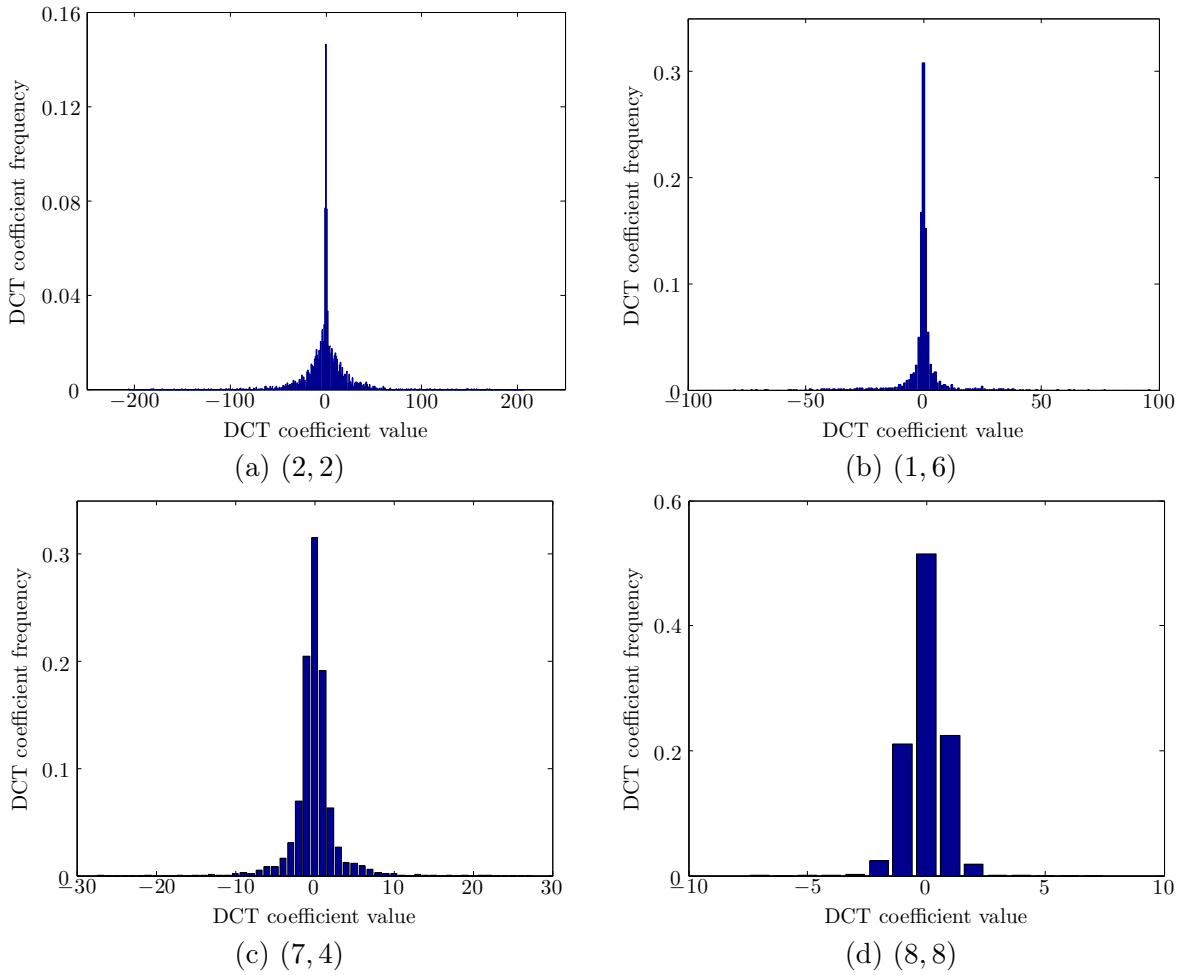


Figure 6.5: Example DCT histograms of different subbands of an anti-forensic JPEG image $\mathcal{F}_1^J$ shown in Figure 6.4-(d). No noticeable comb-like DCT quantization artifacts appear in the histograms.

ROC curves can also be compared in Figure 4.6-(c), Figure 5.8-(b) and Figure 6.3). Intuitively, this is to some extent out of our expectation. We expected that the more sophisticated image prior model would be able to push the performance of the resulting anti-forensic JPEG image towards a higher image quality than the TV-based method while still keeping a good forensic undetectability. However, these results are understandable and may be explained by the following reasons:

- In fact, the ability of the EPLL framework with the GMM as the image prior for JPEG image quality enhancement is proven by experiments on 4 classical test images (see Table 6.1) and a large-scale test on UCIDTest dataset (see Table 6.5). However, as shown in Figure 6.2-(b) compared with -(a) produced by the TV-based JPEG deblocking, the DCT coefficient modification in $\hat{\mathcal{I}}^J$ is relatively minor. In order to remove the remaining comb-like DCT-domain quantization artifacts, the cost will be more expensive. In other words, more image quality may need to be sacrificed. On the other hand, $\hat{\mathcal{I}}^J$ is our only

kind of (intermediate) anti-forensic JPEG image in this thesis that has a higher visual quality than the JPEG image. Further improvement would be possible to create from $\hat{\mathcal{I}}^J$ the anti-forensic JPEG image with even higher quality than the JPEG image, if the following anti-forensic steps are more carefully designed.

- As shown in Figure 6.1-(d), the calibration is able to help $\hat{\mathcal{I}}^J$ get a quite good estimation of the DCT-domain quantization noise with some spatial-domain coherence. The process is very fast, however the estimation is still quite crude. This leads to main image degradation (see the PSNR value change from the 4nd row to the 5th row in Table 6.5) during the creation of the proposed anti-forensic JPEG image $\mathcal{F}_1^J$.
- Regarding the forensic undetectability, $\mathcal{F}_1^J$ has less good anti-forensic performance in blocking artifacts or spatial-domain unnatural noise examination than $\mathcal{F}_0^J$ and $\mathcal{F}^J$. This may be explained by the fact that there is an explicit TV-based deblocking term (see Eq. (4.6) and Figure 4.3) in the creation of $\mathcal{F}_0^J$ and $\mathcal{F}^J$, which is not used in Eq. (6.7).

6.5 Summary

By using the MAP-based image restoration, calibration and a prior of natural image statistics, we create anti-forensic JPEG image $\mathcal{F}_1^J$ which outperforms the state-of-the-art anti-forensic JPEG images [Sta+10a, Sta+10b, SL11, VTT11, SS11] with a higher image quality as well as a slightly better forensic undetectability against existing forensic detectors. Moreover, the DCT histogram is explicitly smoothed by the proposed calibration based DCT-domain quantization noise estimation to better approximate the original one.

However, the proposed method based on a sophisticated image prior model does not outperform our previous JPEG anti-forensic methods proposed in Chapters 4 and 5. The possible reasons are provided in the end of Section 6.4. As a preliminary JPEG anti-forensics work leveraging on more sophisticated image prior model than the TV, the performance of the proposed method in this chapter may be further improved by a more accurate DCT-domain quantization noise estimation and the integration of more anti-forensic terms in Eq. (6.7). Some ideas of possible improvement are provided at the end of Sections 6.3 and 6.4. One ambitious but challenging objective is to create anti-forensic JPEG images with even higher visual quality than the JPEG image, where a good image prior model would be indispensable.

From Chapter 4, Chapter 5 and this chapter, we have seen the superiority of anti-forensic methods designed based on concepts/methods from image restoration over the methods based on noise injection and simple image processing. This draws our interest to work on other image anti-forensic problems following the same research line. During our study of JPEG anti-forensics, we are aware that Stamm *et al.* [Sta+10b, SL11] propose to use median filtering to remove the spatial-domain blocking artifacts from a JPEG image. Moreover, median filtering is also used for image resampling anti-forensic purposes in [KR08]. The median filtering traces present in one image, not only expose that the image is median filtered, but also indicate other image processing operations may have been applied. Therefore, it is important to study

median filtering on the anti-forensic side. This work will be conducted in Chapter 7, by still following our research line of leveraging on image restoration to design image anti-forensics.

Chapter 7

Median Filtered Image Quality Enhancement and Anti-Forensics via Variational Deconvolution

Contents

7.1 Introduction and Motivation . . . 125

7.2 Analysis of Median Filtering and Its Impact on Image Statistics . . . 126

7.2.1 Median Filtering Process . . . 126

7.2.2 Observations of Pixel Value Difference Distribution . . . 127

7.3 Proposed Image Variational Deconvolution Framework . . . 129

7.3.1 Problem Formulation . . . 129

7.3.2 Kernel Selection and Parameter Settings . . . 132

7.3.3 Median Filtered Image Quality Enhancement . . . 133

7.3.4 Anti-Forensics against Median Filtering Detection . . . 135

7.3.4.1 Pixel Value Perturbation . . . 137

7.3.4.2 Parameter Settings . . . 138

7.3.4.3 Performance Evaluation and Comparison . . . 142

7.4 Applications: Disguising Footprints of Both Median Filtering and Targeted Image Operation of Median Filtering Processing . . . 146

7.4.1 Hiding Traces of Image Resampling . . . 146

7.4.2 Removing JPEG Blocking Artifacts . . . 149

7.5 Summary . . . 150

This chapter proposes an image variational deconvolution framework for both quality enhancement and anti-forensics of the median filtered image (*i.e.*, MF image for short). The proposed optimization-based framework consists of a convolution term, a fidelity term with respect to the MF image, and a prior term. The first term is for the approximation of the median filtering process, using a convolution kernel. The second fidelity term keeps the processed image to some extent still close to the MF image, retaining some denoising or other image processing artifacts hiding effects. Using the generalized Gaussian as the distribution model, the last image prior term regularizes the pixel value derivative of the obtained image so that its distribution resembles that of the original image. The proposed method can serve as an MF image quality enhancement technique, whose efficacy is validated by experiments conducted on MF images which have been previously "salt & pepper" noised. Using another parameter setting and with an additional pixel value perturbation procedure, the proposed

method outperforms the state-of-the-art median filtering anti-forensic methods, with a better forensic undetectability against existing detectors as well as a higher visual quality of the processed image. Furthermore, the feasibility of concealing image resampling traces and JPEG blocking artifacts is demonstrated by experiments, using the proposed median filtering anti-forensic method.

A paper describing this work has been accepted for publication in an international journal [Fan+15]. The Matlab code of the method is freely shared online and can be downloaded from: http://www.gipsa-lab.grenoble-inp.fr/~wei.fan/documents/AFMF-TIFS15.tar.gz.

7.1 Introduction and Motivation

In Chapters 4, 5 and 6, we have seen the feasibility of employing some concepts/methods from image restoration meanwhile integrating some extra anti-forensic terms/strategies to devise JPEG anti-forensics with better performance than the state-of-the-art methods. In this chapter, we still follow this research line and study another image anti-forensic problem, that is, median filtering anti-forensics. Our choice to study median filtering is because of its link to JPEG anti-forensics, which is one of our two main research topics in this thesis. In [Sta+10b, SL11], Stamm *et al.*'s proposed to use median filtering for removing the JPEG blocking artifacts in the spatial domain after the DCT histogram of the JPEG image is smoothed by a dithering operation [Sta+10a, SL11]. Despite of the effectiveness of median filtering in the JPEG blocking artifacts removing task, the resulting anti-forensic JPEG image has not been examined by median filtering forensic detectors yet [BK13]. This partially motivates us to conduct the study of median filtering anti-forensics.

The median filter, a well-known local image operator, is frequently adopted for image denoising or smoothing purposes. Median filtering has the reputation of good edge preserving ability, and does not introduce new pixel values to the processed image. However, median filtering also has the shortcoming to degrade the image quality, causing undesired image blurring. Besides, image anti-forensic researchers pointed out the destructive nature of median filtering to other image processing footprints, *e.g.*, disguising JPEG blocking artifacts [Sta+10b, SL11] as described earlier, and hiding traces of image resampling [KR08]. Therefore, the presence of median filtering traces, not only suggests the image has been previously median filtered, but also implies the possibility that other image processing operations may have been applied to the image.

Similar to our JPEG anti-forensic work presented in Chapter 6, we would like to first consider image restoration of MF images, via solving an ill-posed inverse problem. More specifically, the specific inverse problem of median filtering can be treated as a blind deconvolution problem. The word "*blind*" means that we do not have complete knowledge about the convolution kernel used for the filtering. In general, for image deconvolution, the MAP estimation (or one of its variants) is often employed. For the non-blind deconvolution problem, the convolution kernel is known and the use of a good image prior is essential for the problem solving [KF09, KTF11]. As to the blind deconvolution, it is a popular way to perform the kernel estimation first and thereafter solve the problem using non-blind methods [Lev+09, KTF11]. Yet, the median filtering deconvolution is an even more difficult task, since the kernel is not only unknown but also spatially heterogeneous. As a first trial to solve this difficult problem, in this chapter we simplify it to be a tractable blind deconvolution problem with an approximated spatially homogeneous kernel.

In this chapter, we investigate into the median filtering process and the pixel value difference statistical change caused by median filtering. Thereafter, we choose a proper convolution kernel by experimentally studying several spatially homogeneous convolution kernels. As to the image prior, we use the generalized Gaussian distribution to model the pixel value difference whose statistics have a notable change after the image is median filtered. By using a

proper convolution kernel and a good image prior, we form an image variational devolution framework to solve the inverse problem of median filtering. The achievement of the proposed framework is two-fold: (i) improving the visual quality of MF images (with a certain parameter setting) as an image quality enhancement method, and (ii) fooling existing median filtering forensic detectors (with another parameter setting and an additional pixel value perturbation procedure) for anti-forensic purposes.

The remainder of the thesis is organized as follows. Section 7.2 analyzes the median filtering process and image derivative statistical change due to median filtering. The proposed method is described in Section 7.3, where experimental results of MF image quality enhancement and anti-forensics are also presented, with comparisons with the state-of-the-art anti-forensic methods. In Section 7.4, the proposed median filtering anti-forensic method is proven to be practical in the application of image resampling and JPEG blocking anti-forensics, again with experimental comparisons with the state-of-the-art methods. Finally, a summary is given in Section 7.5.

7.2 Analysis of Median Filtering and Its Impact on Image Statistics

7.2.1 Median Filtering Process

According to Eq. (3.11) (basics of median filtering can be found in Section 3.2.1), for each local window, the median filter outputs the median value inside the filter window. During the median filtering process, no new pixel values are generated. For the considered 3×3 window, and for each local window, the median filter can be expressed using a 3×3 convolution kernel matrix:

$$\Psi^k = \begin{bmatrix} \psi_1^k & \psi_4^k & \psi_7^k \\ \psi_2^k & \psi_5^k & \psi_8^k \\ \psi_3^k & \psi_6^k & \psi_9^k \end{bmatrix}, \quad \begin{array}{l} \text{with } \psi_k^k = 1 \text{ and } \psi_i^k = 0 \text{ for } \forall i \neq k, \\ i, k \in \{1, 2, \cdots, 9\}, \end{array} \tag{7.1}$$

which can come in 9 shapes. Nevertheless, for a specific pixel neighborhood, the form of Ψ^k in use is completely dependent on the order statistics of the original pixels compassed by the filter window. Yet, this piece of information is permanently lost during the median filtering process.

Median filtering is a spatially heterogeneous process. For each pixel $\mathbf{x}_i$ of the original image $\mathbf{x}$, there are s^2 possible masks (*i.e.*, the convolution kernel matrices in Eq. (7.1)) in choose, during the median filtering process. For an N-pixel image, the space of possible mask combinations is enormous – there are $(s^2)^N$ possibilities (with a small window size 3×3, and a very small image with size 10×10, there are 2.7×10^{95} possibilities). It appears not practical to consider all the $(s^2)^N$ possible mask combinations for modeling the median filtering process. In order to simplify this model, we propose, as a first trial, to use a single mask and *approximate* the median filtering process as a spatially homogeneous image convolution procedure. A key issue here concerns a properly defined convolution kernel, and we postpone the relevant study

and discussion to Section 7.3.2, where 6 different kinds of spatially homogeneous kernels are experimentally compared.

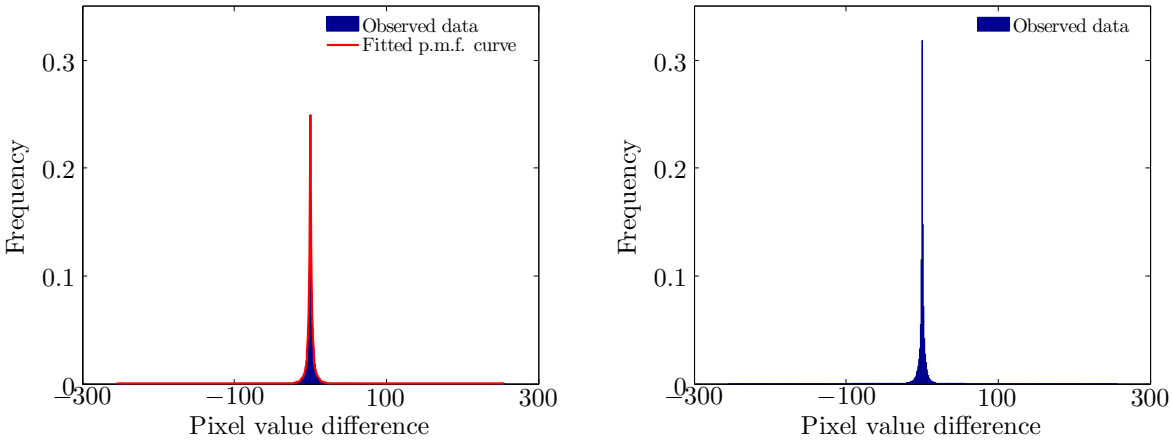
Based on the above analysis, Section 7.3.1 will further formulate the median filtering process approximation as the first term in Eq. (7.4). Indeed, the modeling may appear to be oversimplified, which however makes the problem easy to solve and in practice yields good results (see Sections 7.3.3 and 7.3.4). It is an interesting yet difficult open research problem to estimate the spatially heterogeneous convolution kernel for median filtering, and we leave it for a future study.

7.2.2 Observations of Pixel Value Difference Distribution

Kirchner and Fridrich [KF10] built the forensic measures K_K and $\hat{K}_K$ (see Eqs. (3.12)-(3.13) in Section 3.2.2) based on the first-order pixel value difference histogram. The SPAM feature [PBF10] is based on the analysis of the first-order pixel value difference images, and was initially designed for measuring the effects of ± 1 steganography. It also performed excellently in distinguishing MF images from the original [KF10]. The feature proposed by Cao *et al.* [Cao+10] was built by measuring the first-order pixel value difference image in the textured areas. Yuan [Yua11] did not directly use the pixel value difference for constructing the feature. However, the MFF feature measures local pixel dependency artifacts, where the pixel value difference statistics can contribute as an important one. Also, Chen *et al.* [CN11] studied the pixel neighbor correlation, using a linear prediction model. Furthermore, in [Che+12, CNH13], Chen *et al.* explicitly pointed out that their median filtering forensic method works in the image pixel value difference domain. Kang *et al.* [PK12, Kan+12, Kan+13] did not work on the image pixel value difference domain directly, but on the median filter residual. Nevertheless, they studied the residual correlation in a certain neighborhood, using the Markov chains [PK12] or the autoregressive model [Kan+12, Kan+13]. Concerning Zhang *et al.*'s [Zha+14] median filtering discriminating feature, it was built on the first-order image derivatives.

From the above analysis, we can see that existing median filtering forensic methods [KF10, PBF10, Cao+10, Yua11, CN11, Che+12, CNH13, PK12, Kan+12, Kan+13, Zha+14] either directly analyze the pixel value difference of the image, or study the image statistical change after median filtering in a highly related way. This motivates us to conduct study in the pixel value difference domain as well, via constructing histograms. More precisely, given an image $\mathbf{u}$, we convolve it using a derivative filter, *e.g.*, the first-order horizontal one $\mathbf{f}^1 = [1, -1]$ to obtain the pixel value difference, and then construct its histogram with integers $\{-255, -254, \cdots, 255\}$ as bin centers. For the sake of brevity, we use the matrix multiplication $\mathbf{F}^1\mathbf{u}$ to denote the pixel value difference extraction from the image $\mathbf{u}$ using the filter $\mathbf{f}^1$. The first-order horizontal pixel value difference histogram of an example MFTE (see Section 2.3.2 for the datasets we use for median filtered image quality enhancement and forensic testing) image is shown in Figure 7.1-(a).

In order to properly model the pixel value difference, we adopt the 0-mean two-parameter generalized Gaussian distribution, which enjoys its popularity in natural image statistics



(a) Original image (b) MF image

Figure 7.1: First-order horizontal pixel value difference histograms of an example MFTE image and its MF version. The sample variance and sample kurtosis of the pixel value differences are respectively: (a), $\hat{\sigma}^2 = 36.2568$, $\hat{\kappa} = 27.4651$; (b), $\hat{\sigma}^2 = 29.8866$, $\hat{\kappa} = 36.6555$. The red curve in (a) is its fitted p.m.f. curve, using the generalized Gaussian distribution with estimated parameters $\hat{\alpha} = 0.4797$, and $\hat{\beta} = 0.4860$. The squared fitting error is 0.0047.

[BS99, WSL13]. Its p.d.f. of random variable d is given by:

$$g(d) = \frac{\beta}{2\alpha\Gamma(1/\beta)} e^{-(|d|/\alpha)^{\beta}}, \tag{7.2}$$

where α (> 0) is the scale parameter, β (> 0) is the shape parameter, and $\Gamma(\cdot)$ is the Gamma function. The two parameters α and β are directly related to the variance and the kurtosis of the distribution:

$$\sigma^2 = \frac{\alpha^2\Gamma(3/\beta)}{\Gamma(1/\beta)}, \quad \kappa = \frac{\Gamma(5/\beta)\Gamma(1/\beta)}{\Gamma(3/\beta)^2}. \tag{7.3}$$

Therefore, the estimated parameters $\hat{\alpha}$ and $\hat{\beta}$ can be obtained from the sample variance and kurtosis, by solving the above two equations, using a numerical method [BS99].

The red curve in Figure 7.1-(a) shows the fitted p.m.f. for the pixel value difference histogram, which well describes its shape. The average squared fitting error on MFTE dataset is 0.0107, which also indicates that the generalized Gaussian distribution is a good model in terms of goodness-of-fit. As a counterpart of Figure 7.1-(a), -(b) presents the first-order horizontal pixel value difference histogram for the median filtered version of the example original MFTE image. We also calculate the average variance and kurtosis of the p.m.f. of $\mathbf{F}^1\mathbf{u}$ on MFTE dataset, for the original images and the MF images, respectively. For $\mathbf{F}^1\mathbf{x}$, the average variance is 129.2823, and the average kurtosis is 22.8263. Whereas for MF images, the average variance of $\mathbf{F}^1\mathbf{y}$ is 80.5936 and its average kurtosis is 40.0911, which are notably lower and higher than those of the original images, respectively. Indeed, the median filter is known to have image smoothing effects, leading to (nearly) constant pixel values in a certain neighborhood. This will lead to a remarkably high peak at bin 0 of the derivative histogram after median filtering, *e.g.*, Figure 7.1-(b), which is embodied by the high kurtosis and low

variance of the p.m.f. of $\mathbf{F}^1\mathbf{y}$.

The above analysis shows that the generalized Gaussian distribution is a suitable model for the derivative histogram of original images in terms of goodness-of-fit, indicating that it can serve as a good image prior. Section 7.3.1 will further formulate the image prior term in Eq. (7.4), which regularizes the image derivative distribution to bring down the high peak of MF image derivative histograms. This regularization is important for median filtering anti-forensics, since existing forensic methods all either directly or in a highly related way analyze the pixel value difference statistical change after median filtering. Besides, we will also discuss the parameter estimation of the generalized Gaussian distribution used as the image prior in Section 7.3.1.

7.3 Proposed Image Variational Deconvolution Framework

7.3.1 Problem Formulation

As discussed in Section 7.1, we propose to treat the MF image quality enhancement and anti-forensics as an ill-posed image restoration problem based on energy minimization. Based on the analysis in Section 7.2, and to some extent inspired by [KF09], we hereby propose to minimize the following cost function:

$$\tilde{\mathbf{x}} = \arg\min_{\mathbf{u}} \left(\frac{\lambda}{2} \left(\|\mathbf{K}\mathbf{u} - \mathbf{y}\|_2^2 + \omega \|\mathbf{u} - \mathbf{y}\|_2^2 \right) + \sum_{j=1}^{J} \left\| \frac{\mathbf{F}^j \mathbf{u}}{\alpha_j} \right\|_{\beta_j}^{\beta_j} \right), \tag{7.4}$$

where the first term approximates the median filtering process, and the second term is the image fidelity term with respect to the MF image $\mathbf{y}$, while the last one is the image prior term. It is an image variational deconvolution problem, where λ is a parameter balancing the different energy terms, $\mathbf{K}\mathbf{u}$ is the matrix multiplication form of the image convolution using kernel $\mathbf{k}$, ω (< 1) is a small positive parameter balancing the first two terms more or less related to the image quality (the setting of $\mathbf{k}$ and ω is to be further discussed in Section 7.3.2), J is the number of derivative filters in use, and $\mathbf{F}^j\mathbf{u}$ is the matrix multiplication form for calculating the j-th type of image derivatives.

The analysis in Section 7.2.1 leads to the formation of the first term in Eq. (7.4), that is $\|\mathbf{K}\mathbf{u} - \mathbf{y}\|_2^2$, approximating the median filtering process using image convolution. It can also be considered as an image quality term, because by using this term, we hope in some sense that the processed image $\tilde{\mathbf{x}}$ is close to the original image $\mathbf{x}$, as the filtered version of $\tilde{\mathbf{x}}$ should be close to the MF image $\mathbf{y}$.

In order to keep some median filtering effects (*e.g.*, denoising, and hiding traces for other image processing operations), the second term $\|\mathbf{u} - \mathbf{y}\|_2^2$ is designed for the image fidelity with respect to the MF image (the strength is controlled by the small positive parameter ω), hoping that the processed image is still to some extent close to the MF image. The setting of

the parameter ω will be further discussed in Section 7.3.2, according to different processing purposes of MF images.

The last term in Eq. (7.4) is the image prior term, modeling the image derivative using the 0-mean two-parameter generalized Gaussian distribution, based on the analysis in Section 7.2.2. It is derived by maximizing the log-likelihood of Eq. (7.2), replacing d by $\mathbf{F}^j\mathbf{u}$, α by α_j, and β by β_j, respectively. The effect of this image prior is to regularize the distribution of the derivatives of the obtained image, so that it resembles that of the original image. In this thesis, in total we consider $J = 4$ derivative filters. Besides the filter $\mathbf{F}^1$ defined earlier, here $\mathbf{F}^2$, $\mathbf{F}^3$, and $\mathbf{F}^4$ respectively correspond to the derivative filters $\mathbf{f}^2 = [1, -1]^T$, $\mathbf{f}^3 = [1, 0, -1]$, and $\mathbf{f}^4 = [1, 0, -1]^T$. Theoretically, additional image derivative filters can be added into this framework, however leading to a high computation cost but minor impact on the final results. Moreover, it is of more importance to recover these four image derivative histograms than others constructed by filters where pixels in subtraction are more widely apart.

A key issue of the image prior concerns the two parameters α_j and β_j of the derivative distribution of the original image. However, we cannot get access to the original image in our task. As shown in Section 7.2.2, the two parameters are directly related to the variance and kurtosis of the observed data. We therefore propose to use the linear regression for estimating the original derivative variance and kurtosis by:

$$\begin{cases} \hat{\sigma}^2(\mathbf{F}^j\mathbf{x}) = \mathbf{c}_1^{\sigma^2} + \sum_{m=1}^{M} \mathbf{c}_{m+1}^{\sigma^2} \times \hat{\sigma}^2\left(\mathbf{F}^j\mathcal{M}\mathcal{F}^{(m)}(\mathbf{x})\right), \\ \hat{\kappa}(\mathbf{F}^j\mathbf{x}) = \mathbf{c}_1^{\kappa} + \sum_{m=1}^{M} \mathbf{c}_{m+1}^{\kappa} \times \hat{\kappa}\left(\mathbf{F}^j\mathcal{M}\mathcal{F}^{(m)}(\mathbf{x})\right), \end{cases} \tag{7.5}$$

where $\hat{\sigma}^2(\cdot)$ and $\hat{\kappa}(\cdot)$ respectively return the sample variance and kurtosis of the input, and $\mathbf{c}_m^{\sigma^2}$ and $\mathbf{c}_m^{\kappa}$ $(m = 1, 2, \cdots, M+1)$ are the linear regression coefficients. With $\mathbf{c}^{\sigma^2}$ and $\mathbf{c}^{\kappa}$, the derivative variance and kurtosis of the original image can be estimated from the MF image and its median filtered versions. In practice, the estimation accuracy improves when M increases. In this thesis, we set $M = 5$, which gives us satisfying results. From the 107 images of MFPE dataset, $\mathbf{c}^{\sigma^2}$ and $\mathbf{c}^{\kappa}$ are obtained. The accuracy of the proposed linear prediction based estimation is quite satisfactory, for example, the mean absolute percentage error is around 10% for the variance estimated on the MFTE dataset.

Indeed, our image variational deconvolution based problem formulation (*i.e.*, Eq. (7.4)) shares some similarities with Krishnan and Fergus' [KF09] non-blind image deconvolution problem, or other image deconvolution problems in the image restoration literature. However, the two have some important differences. Firstly, the proposed problem is not a strict MAP estimate as Krishnan and Fergus' [KF09] which is composed of a prior term and a likelihood term. Our cost function in Eq. (7.4) is more complex, in particular having an additional fidelity term with respect to the MF image. Secondly, a more sophisticated image prior is employed in our problem other than the hyper-Laplacian model for the two first-order image derivatives used in [KF09]. The 0-mean two-parameter generalized Gaussian model has the capability to better describe the image derivative distribution than the one-parameter hyper-Laplacian model [KF09]. Thirdly, the two generalized Gaussian parameters are adaptively adjusted (estimated) according to each given image, which is also different from the one fixed

hyper-Laplacian parameter used in [KF09]. Finally, an in-depth motivation behind our image prior modeling image derivatives is based on the specific median filtering inverse problem, where the neighboring pixel relation is prominently different before and after median filtering. Accordingly, two additional filters are adopted in the proposed image variational deconvolution framework.

The optimization of the cost function in Eq. (7.4) is not trivial. Practically, it can be solved by using the widely used "Half Quadratic Splitting" (see Section 2.4.3 for more descriptions) [KF09, ZW11] and the split Bregman method (see Section 2.4.4 for more descriptions) [GO09, Cha09, KTF11]. After introducing a set of auxiliary variables $\{\mathbf{w}^j\}_{j=1}^J$ and using the "Half Quadratic Splitting", the optimization problem in Eq. (7.4) can be re-written as:

$$\min_{\mathbf{u},\{\mathbf{w}^j\}} \left(\frac{\lambda}{2} \left(\|\mathbf{K}\mathbf{u} - \mathbf{y}\|_2^2 + \omega \|\mathbf{u} - \mathbf{y}\|_2^2 \right) + \sum_{j=1}^{J} \left(\frac{\gamma}{2} \left\|\mathbf{F}^j\mathbf{u} - \alpha_j \mathbf{w}^j\right\|_2^2 + \left\|\mathbf{w}^j\right\|_{\beta_j}^{\beta_j} \right) \right). \tag{7.6}$$

where γ is a regularization parameter.

We apply the Bregman iteration to Eq. (7.6). Thereafter, the split Bregman method solves the problem at the $(k+1)$-th $(k = 0, 1, 2, \cdots)$ iteration by the following formulas:

$$\begin{cases} \left(\mathbf{u}^{(k+1)}, \{\left(\mathbf{w}^j\right)^{(k+1)}\}\right) = \arg\min\limits_{\mathbf{u},\{\mathbf{w}^j\}} \left(\frac{\lambda}{2} \left(\|\mathbf{K}\mathbf{u} - \mathbf{y}\|_2^2 + \omega \|\mathbf{u} - \mathbf{y}\|_2^2 \right) \right. \\ \qquad\qquad\qquad\qquad \left. + \sum\limits_{j=1}^{J} \left(\frac{\gamma}{2} \left\|\mathbf{F}^j\mathbf{u} + \left(\mathbf{b}^j\right)^{(k)} - \alpha_j \mathbf{w}^j\right\|_2^2 + \left\|\mathbf{w}^j\right\|_{\beta_j}^{\beta_j} \right) \right), \\ (\mathbf{b}^j)^{(k+1)} = (\mathbf{b}^j)^{(k)} + \left(\mathbf{F}^j\mathbf{u}^{(k+1)} - \alpha_j (\mathbf{w}^j)^{(k+1)}\right). \end{cases} \tag{7.7}$$

where $\{\mathbf{b}^j\}_{j=1}^J$ are the Bregman variables, and $(\mathbf{b}^j)^{(0)} = \mathbf{0}$.

The minimization problem in Eq. (7.7) can be solved by alternating between the following two sub-problems:

- $\mathbf{w}$ sub-problem, solving $\mathbf{w}$ given $\mathbf{u}$. This is solved using a numerical method for different values of $\mathbf{F}^j\mathbf{u} + \left(\mathbf{b}^j\right)^{(k)}$, in a similar way as [KF09].

- $\mathbf{u}$ sub-problem, solving $\mathbf{u}$ given $\mathbf{w}$. This quadratic problem has a closed-form solution, which can be obtained by taking the derivative of the problem with respect to $\mathbf{u}$ and then setting it to 0.

In practice, we run 20 Bregman iterations, during each of which, $\mathbf{w}$ sub-problem and $\mathbf{u}$ sub-problem are respectively solved once. For a 512×512 MFTE image median filtered with window size 3×3, it requires around 5 seconds to process the image, using Matlab R2012b on a PC with 16G RAM and 2.80GHz CPU.

7.3.2 Kernel Selection and Parameter Settings

For a given MF image to be processed, in order to solve the proposed variational deconvolution problem (see Eq. (7.7)), there are several parameters that we can adjust: the convolution kernel $\mathbf{k}$, ω, λ, and γ.

As discussed in Section 7.2.1, we propose to use a spatially homogeneous convolution kernel $\mathbf{k}$ to approximate the median filtering process. For natural images, inside the $s \times s$ window, the block center pixel appears more frequently to hold the median value. This is observed and extracted by Yuan [Yua11] as a feature for detecting median filtering in images. Here, we calculate the empirical distribution of the block median, namely Yuan's [Yua11] $\mathbf{f}^{DBM}$ feature vector (see Section 3.2.2.3), and use the normalized version as a convolution kernel (denoted as the DBM kernel). In this thesis, the DBM kernel for $s = 3$ is estimated from the 107 original images in MFPE dataset as:

$$\Psi^{DBM} = \begin{bmatrix} 0.0930 & 0.1076 & 0.0927 \\ 0.1109 & 0.1921 & 0.1109 \\ 0.0926 & 0.1074 & 0.0929 \end{bmatrix}. \tag{7.8}$$

Besides, we also study the widely used average kernel (AVE) and Gaussian kernel with standard deviation 0.5 (GAU):

$$\Psi^{AVE} = \begin{bmatrix} 0.1111 & 0.1111 & 0.1111 \\ 0.1111 & 0.1111 & 0.1111 \\ 0.1111 & 0.1111 & 0.1111 \end{bmatrix}, \tag{7.9}$$

$$\Psi^{GAU} = \begin{bmatrix} 0.0113 & 0.0838 & 0.0113 \\ 0.0838 & 0.6193 & 0.0838 \\ 0.0113 & 0.0838 & 0.0113 \end{bmatrix}. \tag{7.10}$$

We can see that the AVE and GAU kernels are blind to the median filtering process, while the DBM kernel tries to integrate some information of this process. The order statistics in the filter window varies according to the image content itself. We therefore adopt the blind convolution kernel estimation in [KTF11] for an adaptive kernel adjustment according to the given image. For the sake of brevity, we use DBME, AVEE, and GAUE for referring to the estimated kernels using the DBM, AVE, and GAU kernels as the initialization guess of the blind kernel estimation algorithm [KTF11], respectively. Note that for the same given image, the kernel estimation algorithm of [KTF11] may return different kernel matrices for DBME, AVEE, and GAUE. This is because the kernel estimation [KTF11] solves a complex optimization problem whose output is sensitive to the initialization guess.

Moreover, the parameter ω is to balance the first two terms in Eq. (7.4). Obviously, the higher the value of ω is, the more the minimization of the proposed cost function will favor images close to the MF image. Furthermore, λ and γ are also parameters we can tune.

For determining the convolution kernel and ω value, we conduct the test on MFTE100 dataset using different (λ, γ) combinations and choose the most favorable one for either MF im-

age quality enhancement or anti-forensic purposes. Here, we consider $\lambda \in \{1000, 2000, 3000\}$, and $\gamma = \{300, 900, 1500\}$. A more fine-grained parameter grid search of (λ, γ) will be carried out in Sections 7.3.3 and 7.3.4, after finding a proper setting for the convolution kernel and the ω value.

In order to facilitate the parameter choosing for MF image quality enhancement, Figure 7.2 shows the highest average oPSNR[20] value achieved when fixing the type of kernel and the value of ω, and varying the (λ, γ) setting. When $\omega = 0$, the DBM kernel is able to achieve the highest oPSNR value, indicating it is the best one for median filtering process approximation among all the 6 types of convolution kernels in consideration. This result is expected because each element in the DBM kernel is proportional to the probability of a pixel at a given position to be the block median, while the AVE and GAU kernels are blind to the median filtering process. Yet, as discussed in Section 7.2.1, it is indeed oversimplified using a single convolution filter to approximate the spatially heterogeneous median filtering process. We find it is very useful to introduce the $\|\mathbf{u} - \mathbf{y}\|_2^2$ term (with $\omega > 0$) into the proposed framework of Eq. (7.4), so as to reduce the median filtering process approximation errors. From Figure 7.2, we can see that for the three best performing kernels DBM, AVE, and AVEE, an ω value in the interval of $[0.3, 0.5]$ can effectively improve the oPSNR value of the processed MF image on MFTE dataset. Considering $\omega \in \{0, 0.05, 0.1, 0.2, \cdots, 1\}$ and different kernels, the highest average oPSNR value is achieved with $\omega = 0.4$ and the AVE kernel. This setting will be adopted for MF image quality enhancement.

As to median filtering anti-forensics, the value of ω needs to be small, so that the processed image is not too close to the MF image retaining too many median filtering artifacts. We study several different values around 0.1, and observe that different ω settings have minor impact on the final results. When $\omega = 0.1$, the AVE and AVEE kernels outperform the other 4 kernels in terms of anti-forensic performance. Take into account the image quality, the AVEE kernel is able to achieve slightly higher oPSNR, oSSIM, and mSSIM values, and 0.4585 dB of mPSNR gain on average, compared with the AVE kernel. For a better tradeoff between the forensic undetectability and image quality of the anti-forensic MF images, we choose the AVEE kernel with $\omega = 0.1$ as the final setting.

7.3.3 Median Filtered Image Quality Enhancement

As suggested in Section 7.3.2, we use the AVE kernel and $\omega = 0.4$ for MF image quality enhancement. The average oPSNR and oSSIM values can be seen in Figure 7.3, for different (λ, γ) combinations, where $\lambda \in \{1000, 1500, \cdots, 3000\}$ and $\gamma \in \{200, 300, \cdots, 1000\}$. Compared with the average oPSNR value 36.8114 dB and the average oSSIM value 0.9823 for

[20] For evaluating the image quality, the well-known PSNR and structural similarity (SSIM) [Wan+04] metrics are used. In this thesis, we calculate the image quality metric values of the given image, using both the original image and the MF image as the references, under different circumstances. For the sake of clarity and to avoid confusion, we hereafter use "oPSNR" and "oSSIM" to specifically refer to the two mentioned image metrics using the original image as the reference, whereas "mPSNR" and "mSSIM" are for those when the MF image serves as the reference. Note that, these notations only apply in this chapter. Besides, "PSNR" and "SSIM" still generically mean the two image quality metrics.

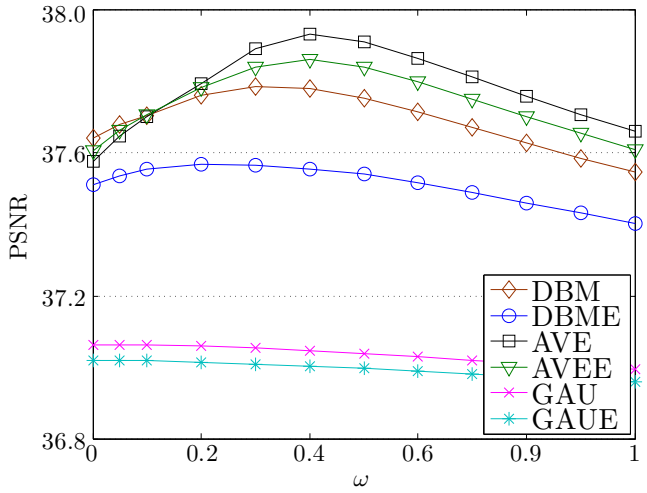
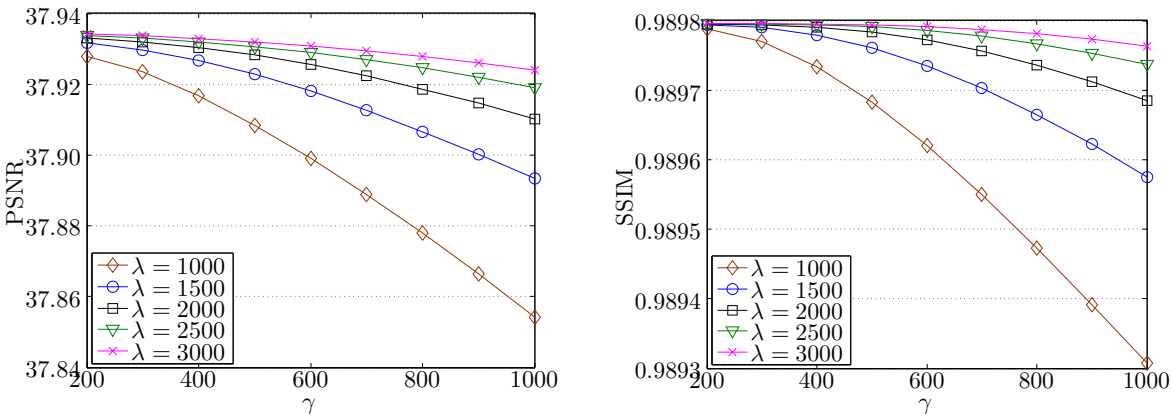


Figure 7.2: The average oPSNR value (obtained on MFTE100 dataset) for different convolution kernels and ω values. Each point in the figure corresponds to the (λ, γ) setting which achieves the highest average oPSNR value, when fixing a convolution kernel and the ω value.

MF images, an evident image quality improvement of the processed images can be observed. In consideration of both PSNR and SSIM metrics, we choose $\lambda = 3000$ and $\gamma = 200$ as the final parameter setting, which can achieve 37.9343 dB of average oPSNR value and 0.9898 of average oSSIM value on MFTE100 dataset.



(a) oPSNR (b) oSSIM

Figure 7.3: The average oPSNR and oSSIM values for processed MF images by the proposed quality enhancement method using the AVE kernel with $\omega = 0.4$, for different parameter combinations (λ, γ). The average oPSNR and oSSIM values for MF images are: 36.8114 dB, and 0.9823, respectively. Results are obtained on MFTE100 dataset.

Table 7.1 reports the image quality improvement of processing the median filtered "salt & pepper" noised image, using the proposed method. With experimental results obtained from 1000 MFTE images, it can be seen that when the noise density is below 7%, both the oPSNR and the oSSIM values are well improved. Though the average oPSNR value is slightly lower

than that of the MF image when the noise density is 7%, we can still see some oSSIM value increase. In Figure 7.4, example results from an MFTE image are compared, for the original image, the "salt & pepper" noised image, the median filtered noised image, and the processed quality enhanced image using the proposed method. It can be seen that the median filter is indeed powerful for noise removal. Yet, image blurring is also introduced, especially in the image textured regions. The proposed method can help recover part of the information lost during the median filtering process, with a higher image quality achieved than the MF image.

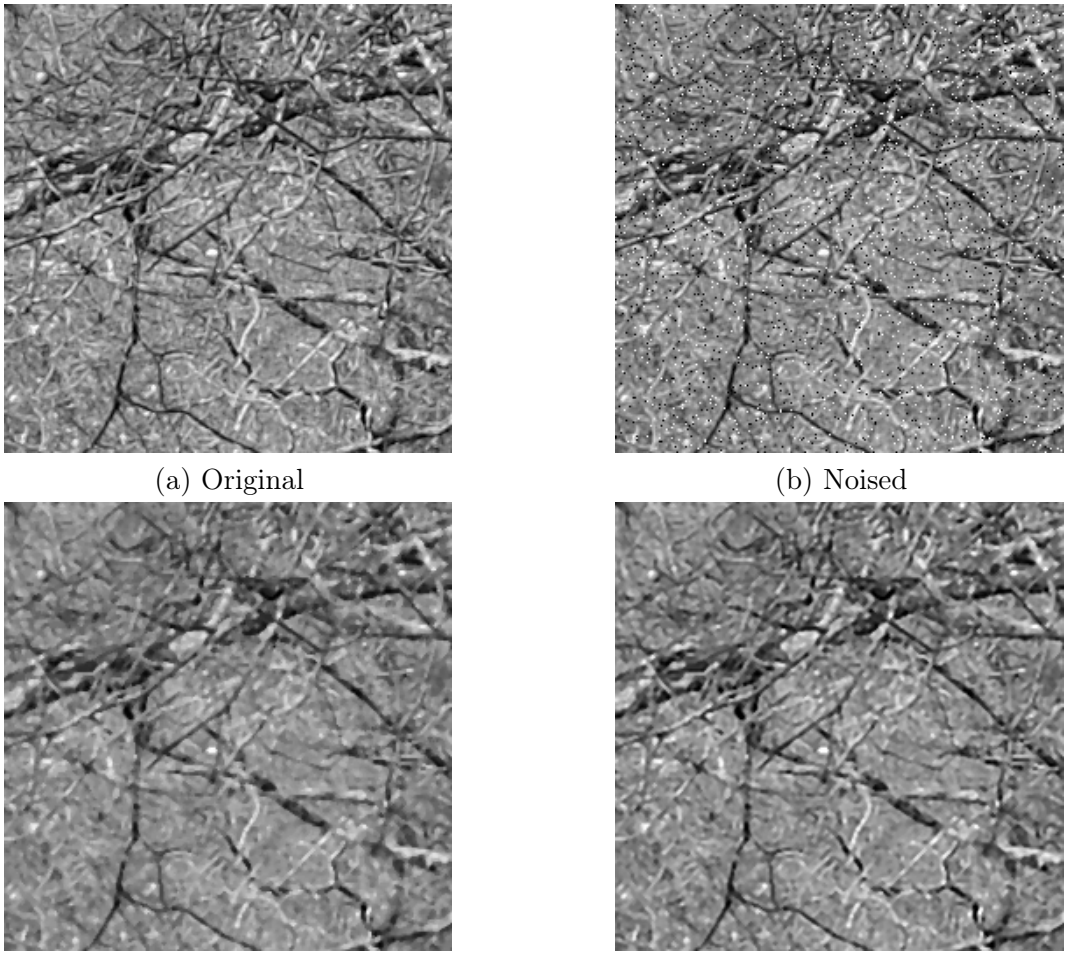
Table 7.1: The average oPSNR and oSSIM values for "salt & pepper" noised, median filtered, and quality enhanced images, respectively. The noise density varies from 1% to 7%. Results are obtained on MFTE dataset.

		Noised	Median filtered	Quality enhanced
1%	oPSNR	25.1365	37.1336	38.0723
	oSSIM	0.8308	0.9827	0.9892
2%	oPSNR	22.1257	36.9719	37.8236
	oSSIM	0.7185	0.9822	0.9885
3%	oPSNR	20.3642	36.7957	37.5155
	oSSIM	0.6388	0.9818	0.9876
4%	oPSNR	19.1161	36.6114	37.2058
	oSSIM	0.5793	0.9813	0.9867
5%	oPSNR	18.1466	36.4031	36.7914
	oSSIM	0.5327	0.9807	0.9850
6%	oPSNR	17.3540	36.1758	36.2933
	oSSIM	0.4948	0.9801	0.9828
7%	oPSNR	16.6851	35.8924	35.7542
	oSSIM	0.4632	0.9793	0.9803

7.3.4 Anti-Forensics against Median Filtering Detection

For each forensic testing scenario, the same number of positive samples ((anti-forensic) MF images) and negative samples (original images) are mixed for classification. Then, we can draw the ROC curve, each point of which corresponds to a classification strategy by the detector. Thereafter, an AUC value can be calculated to quantitatively evaluate the detector's forensic performance (or the forensic undetectability performance of the anti-forensic image).

We consider the median filtering forensic detectors listed in Table 3.3, and use the state-of-the-art anti-forensic MF images listed in Table 3.4 for comparison. For the sake of conciseness, we also use notations for referring to the processed MF images using the methods proposed in this chapter. With the application of median filtering with $s = 3$ to the original image $\mathcal{I}$, the MF image $\mathcal{M}$ is generated, from which the following images are created:



(a) Original (b) Noised

(c) Median filtered (d) Quality enhanced

Figure 7.4: Example results (close-up images) of an MFTE image. The original image (a) is contaminated by some "salt & pepper" noise of density 3% to generate (b), which is then median filtered to obtain (c), which is afterwards processed for quality enhancement using the proposed method to create (d). Their oPSNR and oSSIM values are: (b) 20.8403 dB, 0.9009; (c) 25.1402 dB, 0.9343; (d) 26.5970 dB, 0.9636, respectively.

- $\mathcal{M}^p$, with the application of the MF image quality enhancement method proposed in Section 7.3.3 to the MF image $\mathcal{M}$;
- $\bar{\mathcal{M}}$, with the application of the proposed pixel value perturbation (to be described in Section 7.3.4.1) to the MF image $\mathcal{M}$;
- $\mathcal{F}'^M$, with the application of the proposed anti-forensic method in Section 7.3.1 (with parameters properly tuned) to the MF image $\mathcal{M}$;
- $\mathcal{F}^M$, with the application of the proposed anti-forensic method in Section 7.3.1 (with parameters properly tuned) to the perturbed MF image $\bar{\mathcal{M}}$.

7.3.4.1 Pixel Value Perturbation

From Table 7.2, it can be seen that the anti-forensic performance of the processed MF image $\mathcal{M}^p$ using the proposed quality enhancement method in Section 7.3.3 has been improved, compared to that of $\mathcal{M}$. It implies the possibility of using the proposed framework described in Section 7.3.1 for anti-forensic MF image creation. However, in practice, it needs more image quality sacrifice for a further forensic undetectability improvement by pure image variational deconvolution. This motivates us to firstly perform some pixel value perturbation on the MF image before further processing.

Table 7.2: From the 2nd to the 5th rows, the average PSNR and SSIM values are reported. The following 4 rows show the AUC values of different kinds of images against the 4 scalar-based detectors [KF10, Cao+10, Yua11]. The average KL divergence values of the pixel value difference histograms between the original images and the (quality enhanced) (anti-forensic) MF images are listed in the last 4 rows. Results are obtained on MFTE dataset.

		$\mathcal{M}$	$\mathcal{M}^p$	$\mathcal{F}_W^M$	$\mathcal{F}_D^M$	$\mathcal{F}'^M$	$\mathcal{F}^M$
Image quality	oPSNR	37.2847	38.2953	33.6033	33.4272	37.5526	37.5184
	oSSIM	0.9831	0.9896	0.9552	0.9714	0.9902	0.9901
	mPSNR	–	43.2040	37.5618	36.4076	38.9635	38.8653
	mSSIM	–	0.9956	0.9713	0.9871	0.9899	0.9898
Anti-forensic performance	K_K	0.9722	0.7839	0.4592	0.5347	0.6633	0.5595
	$\hat{K}_K$	0.9824	0.8375	0.6586	0.4635	0.6762	0.5061
	K_C	0.9938	0.8213	0.6668	0.7479	0.6690	0.6490
	K_Y	0.9984	0.7922	0.3336	0.6518	0.6216	0.5886
KL divergence	$\mathbf{f}^1$	0.1632	0.0990	0.1148	0.0547	0.0571	0.0484
	$\mathbf{f}^2$	0.1611	0.0949	0.1338	0.0563	0.0534	0.0449
	$\mathbf{f}^3$	0.0775	0.0525	0.0619	0.0383	0.0314	0.0272
	$\mathbf{f}^4$	0.0753	0.0495	0.0689	0.0389	0.0278	0.0238

The median filter is known to smooth the image, in a way creating (nearly) constant pixel values in a certain neighborhood. This can be easily detected, by analyzing the first-order pixel value difference, *e.g.*, by detectors K_K [KF10], $\hat{K}_K$ [KF10], and K_C [Cao+10]. Through a careful analysis of the above mentioned detectors, we can see that their outputs are highly related to the 0-valued first-order pixel value difference, especially in textured regions of the image. Based on the above analysis and consideration, we propose to firstly perform some minor pixel value perturbation to the MF image as the following.

In the horizontal direction, let us consider three adjacent pixels denoted as a triple $(\mathbf{y}_{i+1}, \mathbf{y}_{i+2}, \mathbf{y}_{i+3})$. If $\mathbf{y}_{i+1} = \mathbf{y}_{i+2} = \mathbf{y}_{i+3}$, there will be two horizontal first-order pixel value differences being 0, that are $\mathbf{y}_{i+2} - \mathbf{y}_{i+1}$ and $\mathbf{y}_{i+3} - \mathbf{y}_{i+2}$. This will directly contribute to the h_0 in Eq. (3.12), leading to a high detector output of K_K which may classify the given image as median filtered. A simple but efficient adjustment is to modify the central pixel $\mathbf{y}_{i+2}$ to $\mathbf{y}_{i+2} \pm 1$. This effectively modifies the first-order statistics with a minor impact on the image

quality. Similarly, we can also apply some modifications to adjacent pixel pair $(\mathbf{y}_{i+1}, \mathbf{y}_{i+2})$ when $\mathbf{y}_{i+1} = \mathbf{y}_{i+2}$. Based on the above analysis, we propose the following MF image pixel value perturbation before it is further processed for anti-forensic purposes, in the horizontal direction followed by that in the vertical direction:

1. In the currently achieved image, find all the adjacent pixel triples $(\mathbf{y}_{i+1}, \mathbf{y}_{i+2}, \mathbf{y}_{i+3})$, where $\mathbf{y}_{i+1} = \mathbf{y}_{i+2} = \mathbf{y}_{i+3}$. Substitute $\mathbf{y}_{i+2}$ randomly with $\mathbf{y}_{i+2} + 1$ or $\mathbf{y}_{i+2} - 1$, meanwhile make sure $\mathbf{y}_{i+1}$ and $\mathbf{y}_{i+3}$ will not be touched in the subsequent modification.

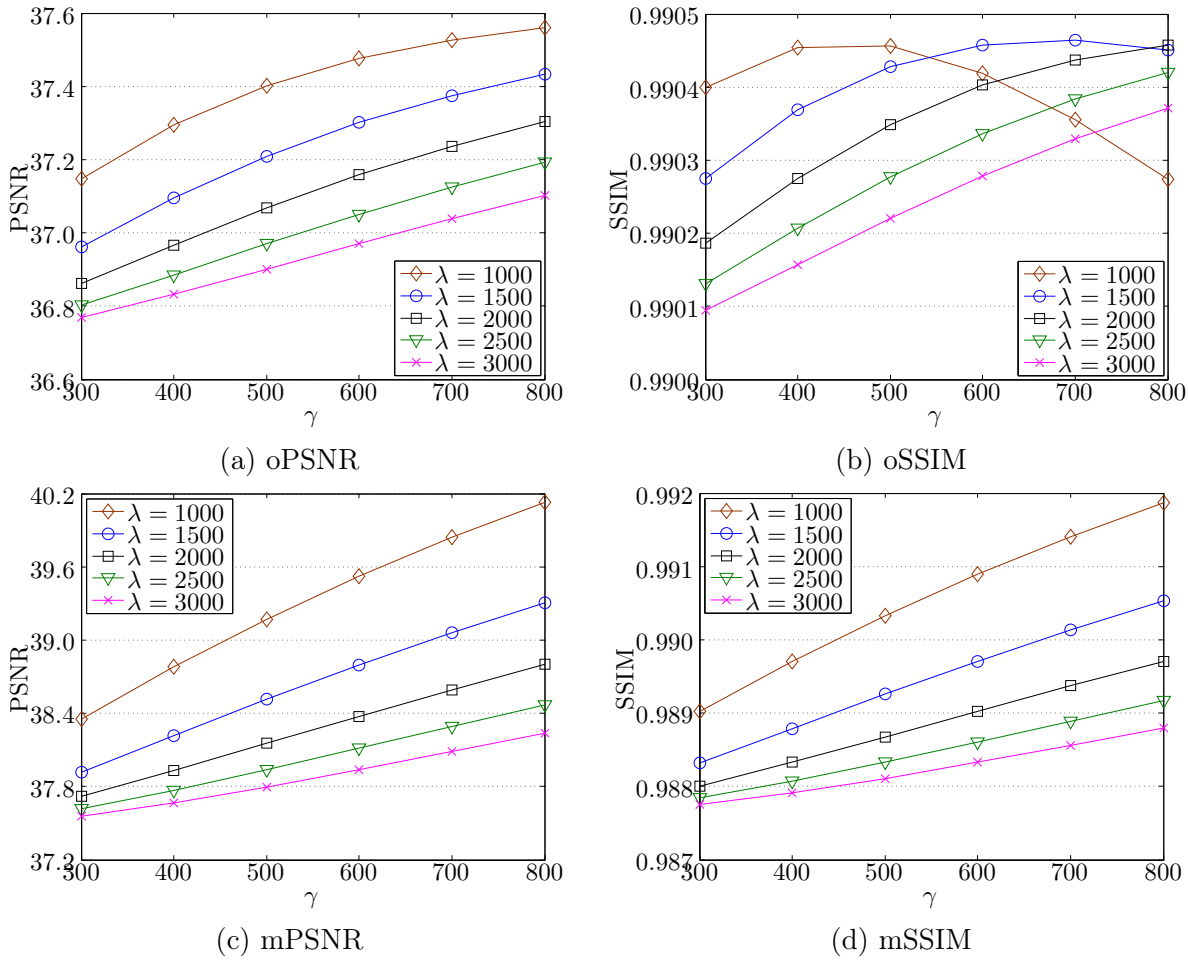
2. From the updated image, find all the adjacent pixel pairs $(\mathbf{y}_{i+1}, \mathbf{y}_{i+2})$, where $\mathbf{y}_{i+1} = \mathbf{y}_{i+2}$. Choose between $\mathbf{y}_{i+1}$ and $\mathbf{y}_{i+2}$ with a higher local variance, update the value of the selected pixel by randomly incrementing or decrementing it by 1, meanwhile make sure the other pixel is not altered. In order to avoid excessive modification to the first-order pixel value difference histogram, only 30% of the equal-valued pixel pairs, with the highest local variances, are processed.

7.3.4.2 Parameter Settings

The proposed median filtering anti-forensic method is to process the MF image obtained after the pixel value perturbation $\bar{\mathcal{M}}$, via solving the image variational deconvolution problem in Eq. (7.4). According to Section 7.3.2, we set $\omega = 0.1$ and use the AVEE kernel for median filtering anti-forensics. A parameter grid search is conducted to find a proper (λ, γ) combination, where $\lambda \in \{1000, 1500, \cdots, 3000\}$ and $\gamma \in \{300, 400, \cdots, 800\}$. The goal is to look for a setting which leads to a good tradeoff between forensic undetectability and image quality for the created anti-forensic MF images.

With experiments carried out on MFTE100 dataset, Figures 7.5 and 7.6 report the variation trend of the forensic undetectability and the image quality, for different settings of (λ, γ). In the existing median filtering anti-forensic work [FB12, WSL13, DN+13], the image quality metrics are all calculated using the MF image as the reference. However, in various applications, such as denoising, hiding traces of JPEG blocking artifacts, the goal is to bring the processed image as close as possible to the original image. In this thesis, we therefore consider both the original image and the MF image as the references for image quality evaluation. Note that, in order to get a quick overview of the anti-forensic performance of the anti-forensic MF image, for now only the scalar-based detectors (K_K [KF10], $\hat{K}_K$ [KF10], K_C [Cao+10], and K_Y [Yua11]) are considered. As to the SVM-based detectors, the forensic undetectability of the proposed method will be reported later.

From Figure 7.5, it can be seen that the image quality improves when λ decreases or γ increases. Yet, when the value of γ is too big, the oSSIM metric suffers from some declines, and this decrease goes faster for smaller λ values. Concerning the AUC metric, it is contradictory with the image quality for detectors K_C and K_Y as shown in Figure 7.6. The AUC for Kirchner and Fridrich's detectors K_K and $\hat{K}_K$ does have a low value when the image quality is high. However, note that the closer to 0.5 the AUC value is, the better forensic undetectability the



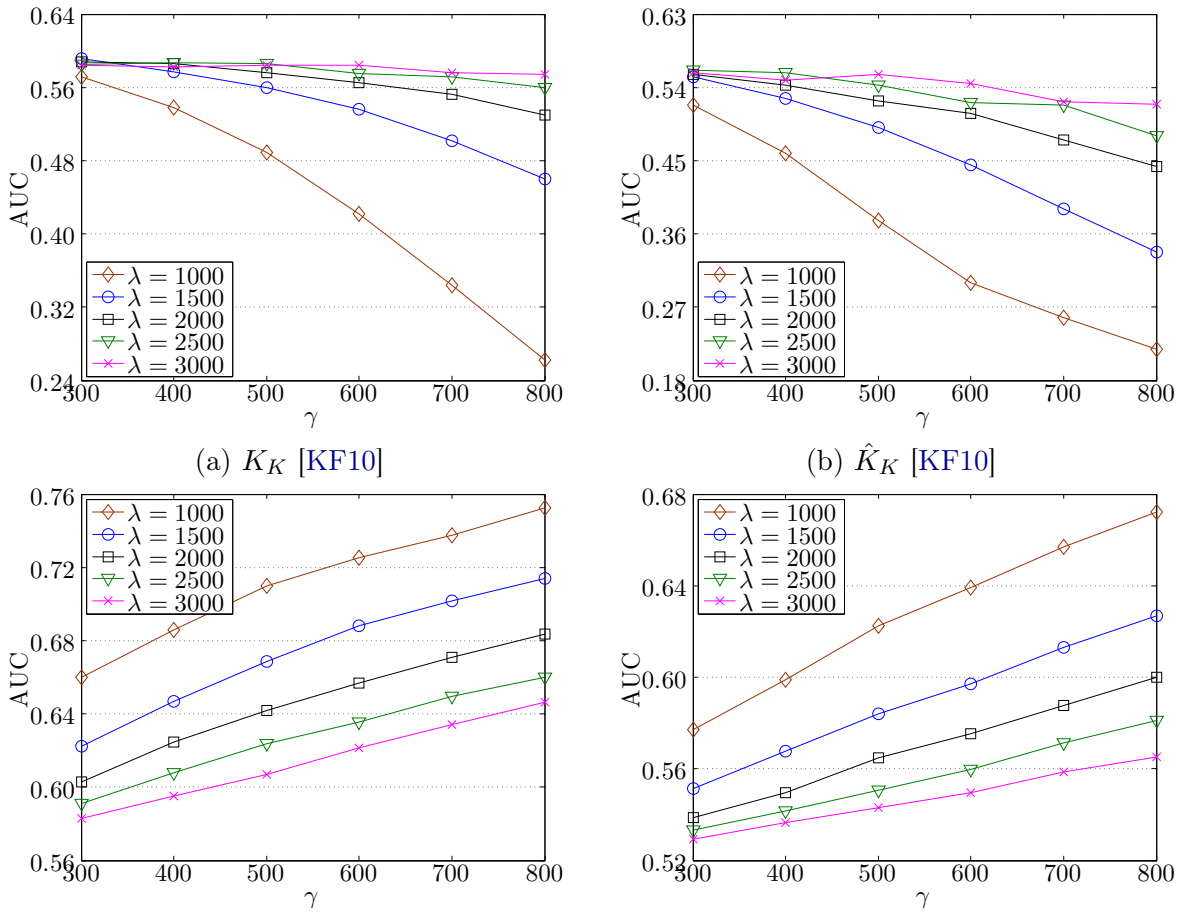
(a) oPSNR (b) oSSIM

(c) mPSNR (d) mSSIM

Figure 7.5: The average image quality evaluation metric values, for anti-forensic MF images $\mathcal{F}^M$ using the AVEE kernel with $\omega = 0.1$, for different parameter combinations (λ, γ). Results are obtained on MFTE100 dataset.

forgery achieves. A very low AUC value does not mean a better anti-forensic performance, since one can flip the classification strategy to gain a high AUC value for the detector. Among all the scalar-based median filtering detectors in consideration, it can be seen that detector K_C is the one relatively hard to be fooled. Using it as a baseline, we choose the parameter setting with which the AUC value for K_C is around 0.65 and the image quality is high. In the end, $\lambda = 1500$ and $\gamma = 500$ are used as the final parameter setting for the median filtering anti-forensic task.

Figure 7.7-(c) is an example anti-forensic MF image $\mathcal{F}^M$ created using the proposed median filtering anti-forensic method. With comparison to the MF image shown in Figure 7.7-(b), we can see that the image quality has been improved and the image blurring has been mitigated. Besides, Figure 7.7-(d) illustrates the ground-truth difference image between the original image and the MF image. Figure 7.7-(e) shows the difference image of our anti-forensic MF image with respect to the MF image. The two difference images -(d) and -(e) are rather visually similar. It can also be seen that the proposed method, to some extent, is able to reproduce



(a) K_K [KF10]

(b) $\hat{K}_K$ [KF10]

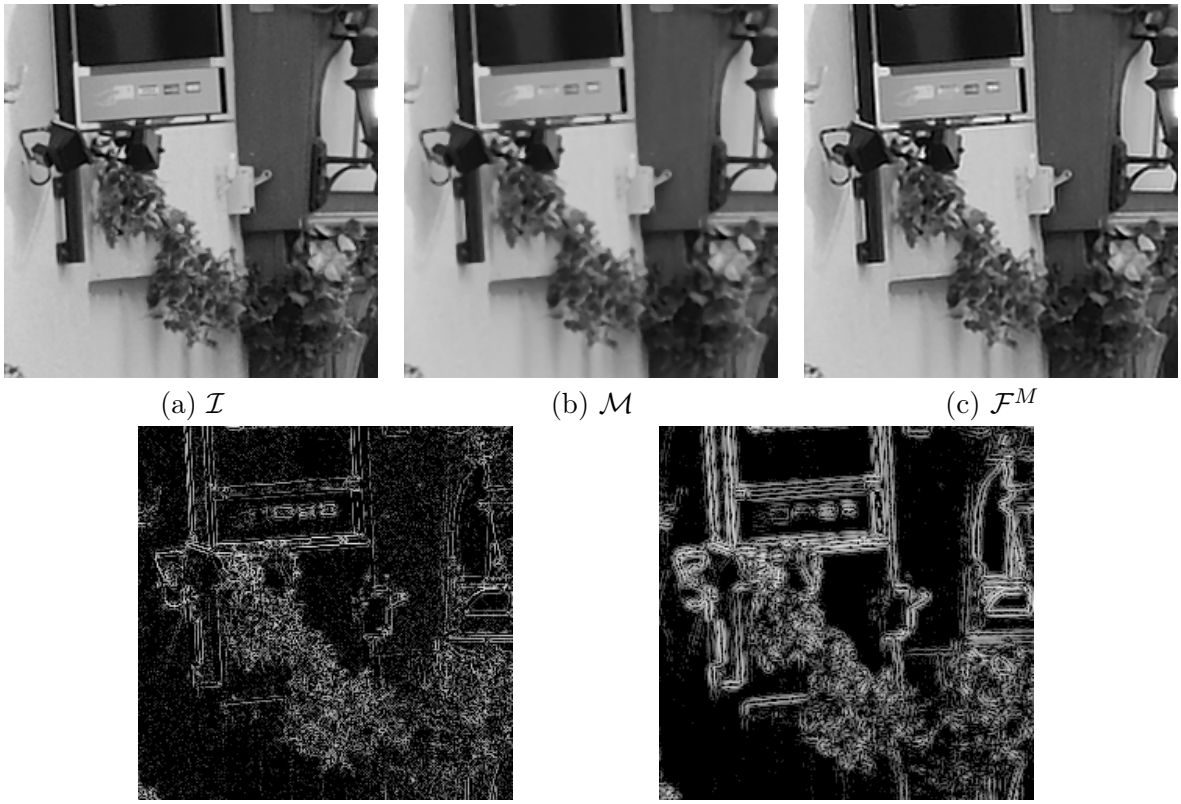
(c) K_C [Cao+10]

(d) K_Y [Yua11]

Figure 7.6: The anti-forensic performance against scalar-based detectors K_K [KF10], $\hat{K}_K$ [KF10], K_C [Cao+10], and K_Y [Yua11], for anti-forensic MF images $\mathcal{F}^M$ using the AVEE kernel with $\omega = 0.1$, for different parameter combinations (λ, γ). Results are obtained on MFTE100 dataset.

the noise-like pattern of the median filtering difference, especially in the textured areas.

Even without the pixel value perturbation, another version of our anti-forensic MF image $\mathcal{F}'^M$ is also capable of creating similar noise-like pattern in the difference image with respect to the MF image shown in Figure 7.7-(e). We refrain from showing the corresponding result due to the fact that it is hard to check the difference between $|\mathcal{F}'^M - \mathcal{M}|$ and $|\mathcal{F}^M - \mathcal{M}|$ by human naked eyes. This can also be inferred from the very small image quality metric value differences between $\mathcal{F}'^M$ and $\mathcal{F}^M$ (see Table 7.2). However, the median filtering anti-forensic performance of $\mathcal{F}^M$ is visibly improved compared with that of $\mathcal{F}'^M$. We are aware that the proposed pixel value perturbation strategy for generating $\bar{\mathcal{M}}$ is well designed for regularizing the first-order pixel value differences 0 and ± 1. Nevertheless, it can be seen that the pixel value perturbation has very minor impact on the image quality, but is able to achieve better forensic undetectability, especially for the specifically targeted detectors K_K and $\hat{K}_K$ [KF10]. Besides, the proposed pixel value perturbation is also capable of dragging the processed image



(a) $\mathcal{I}$ (b) $\mathcal{M}$ (c) $\mathcal{F}^M$

(d) $|\mathcal{I} - \mathcal{M}|$ (e) $|\mathcal{F}^M - \mathcal{M}|$

Figure 7.7: Example results (close-up images) of an MFTE image. For a better visibility of the difference images shown in (d) and (e), we have taken logarithm of the pixel value differences and afterwards carried out a normalization.

out from the detection regions of other detectors to some extent (see results for K_C [Cao+10] and K_Y [Yua11] in Table 7.2). Moreover, the merit of the proposed pixel value perturbation can be found in the further pixel value difference histogram restoration capability (see the KL divergence results in Table 7.2).

Dang-Nguyen *et al.* [DN+13] have shown the superiority of their anti-forensic method over Fontani and Barni's [FB12]. We therefore refrain from the comparison with Fontani and Barni's pioneer method [FB12]. For a fairer comparison with the other two state-of-the-art median filtering anti-forensic methods [WSL13, DN+13], we tune certain parameter(s) of their algorithms on the new forensic dataset MFTE, and choose the setting having a good tradeoff between forensic undetectability and image quality. Figure 7.8 shows the image quality and anti-forensic performance variations with respect to a threshold parameter T (see [DN+13] for details) of Dang-Nguyen *et al.*'s [DN+13] method. We choose $T = 2$ as the final setting, which is in accord with Dang-Nguyen *et al.*'s choice, though a different image dataset is in use. A similar strategy is applied for the parameter setting of Wu *et al.*'s method. For the sake of brevity, the results are not shown here. Specifically, the threshold for the noise is 3, the correction factor is 0.1, and the block size is 7×7 (see [WSL13] for details).

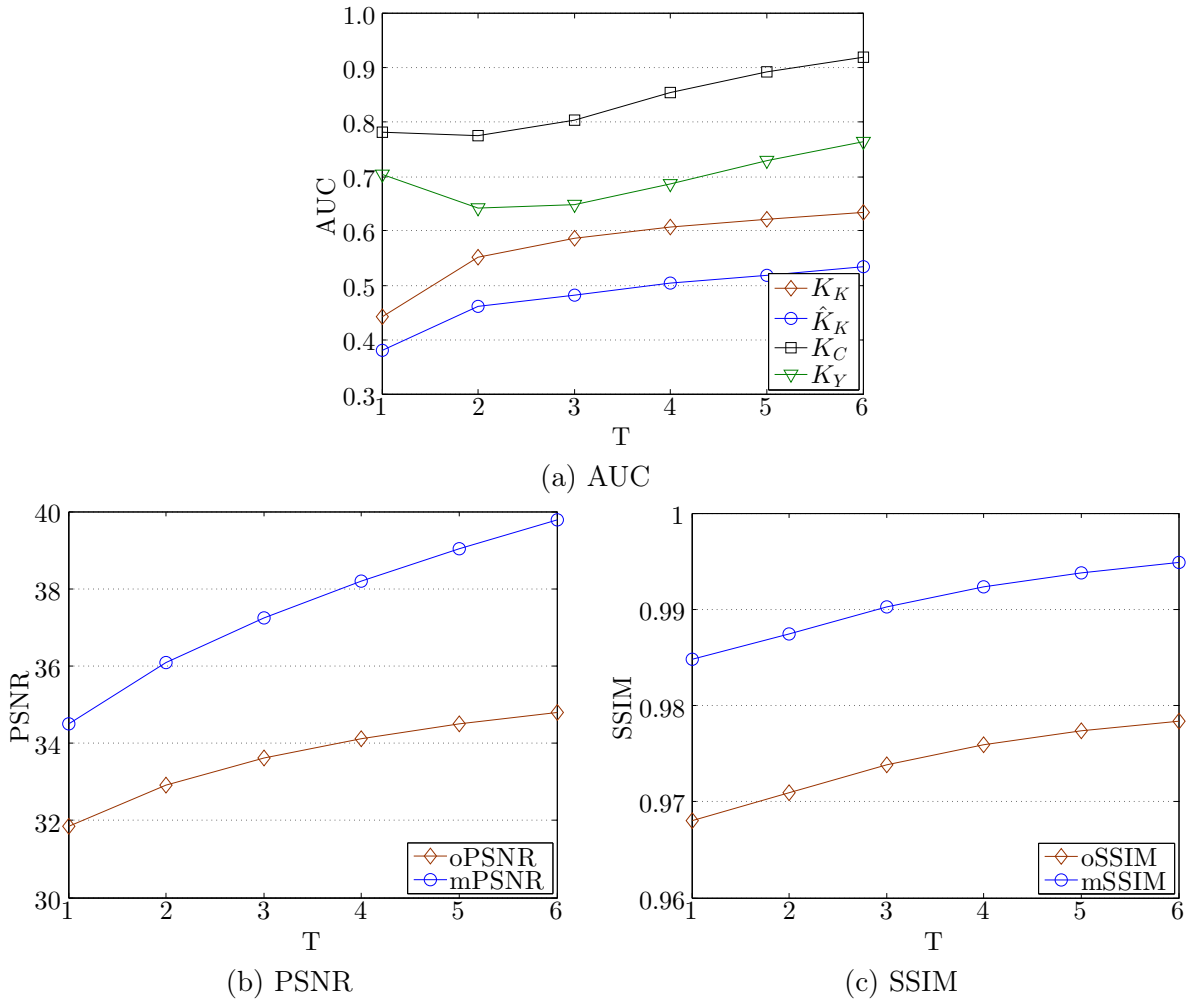


Figure 7.8: Anti-forensic performance against the scalar-based detectors [KF10, Cao+10, Yua11] and image quality of $\mathcal{F}_D^M$ [DN+13], with respect to different settings of the parameter T. Results are obtained on MFTE100 dataset.

7.3.4.3 Performance Evaluation and Comparison

Table 7.2 reports the performance comparison of different kinds of (quality enhanced) (anti-forensic) MF images. ROC curves of anti-forensic MF images against the four scalar-based detectors [KF10, Cao+10, Yua11] are shown in Figure 7.9: among all the existing median filtering anti-forensic methods, ours is able to bring the ROC curves the closest to the random guess. Besides a better overall anti-forensic performance against existing scalar-based median filtering detectors, our anti-forensic MF image $\mathcal{F}^M$ also outperforms Wu *et al.*'s $\mathcal{F}_W^M$ [WSL13] and Dang-Nguyen *et al.*'s $\mathcal{F}_D^M$ [DN+13] in terms of image quality. Moreover, it is worth noticing that our anti-forensic MF image $\mathcal{F}^M$ also achieves even higher average oPSNR and oSSIM values than the MF image $\mathcal{M}$ on MFTE dataset. It means that the proposed median filtering anti-forensic method is not only able to disguise the median filtering traces, but is also capable of improving the visual quality of the MF image at the same time.

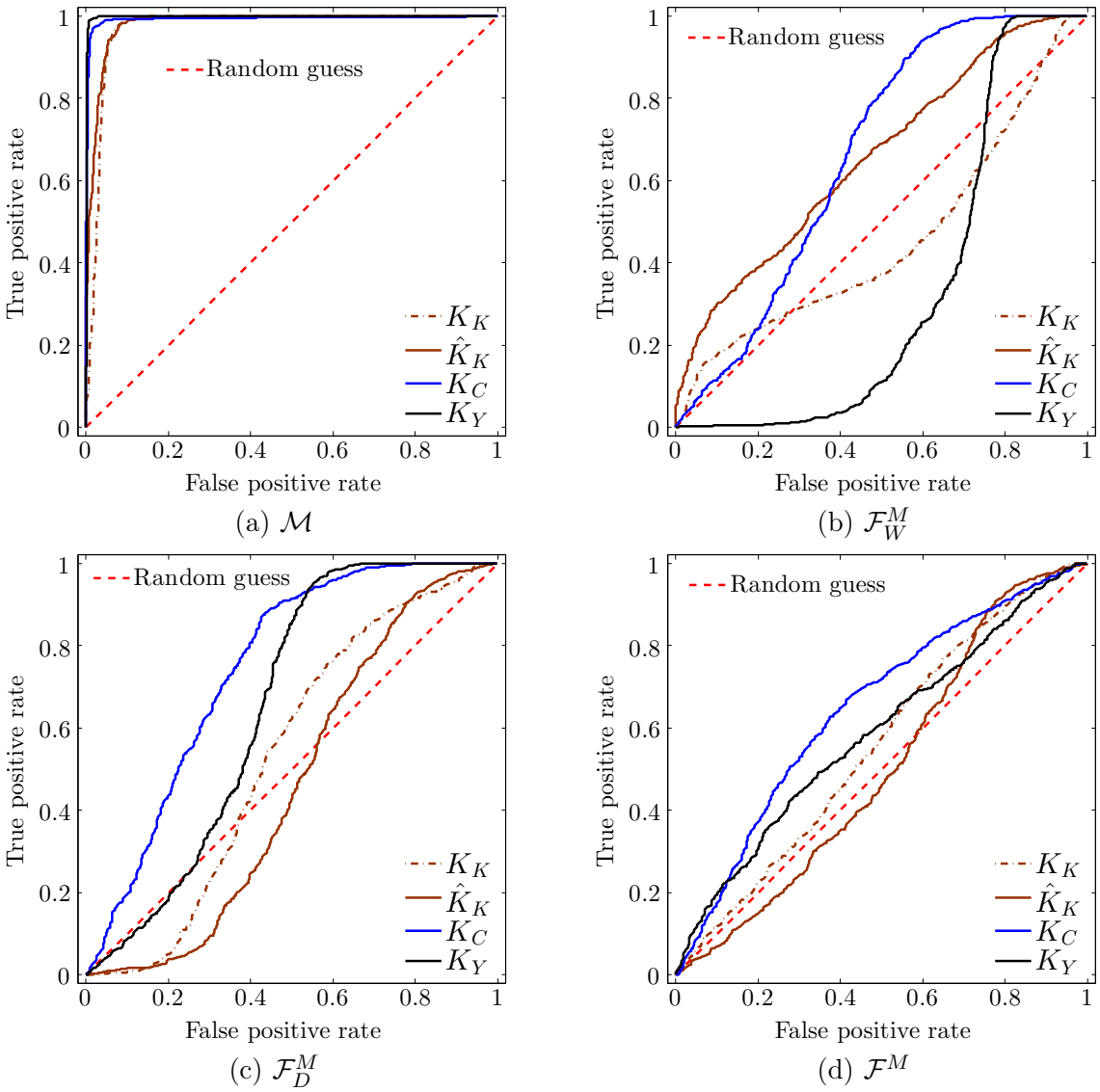


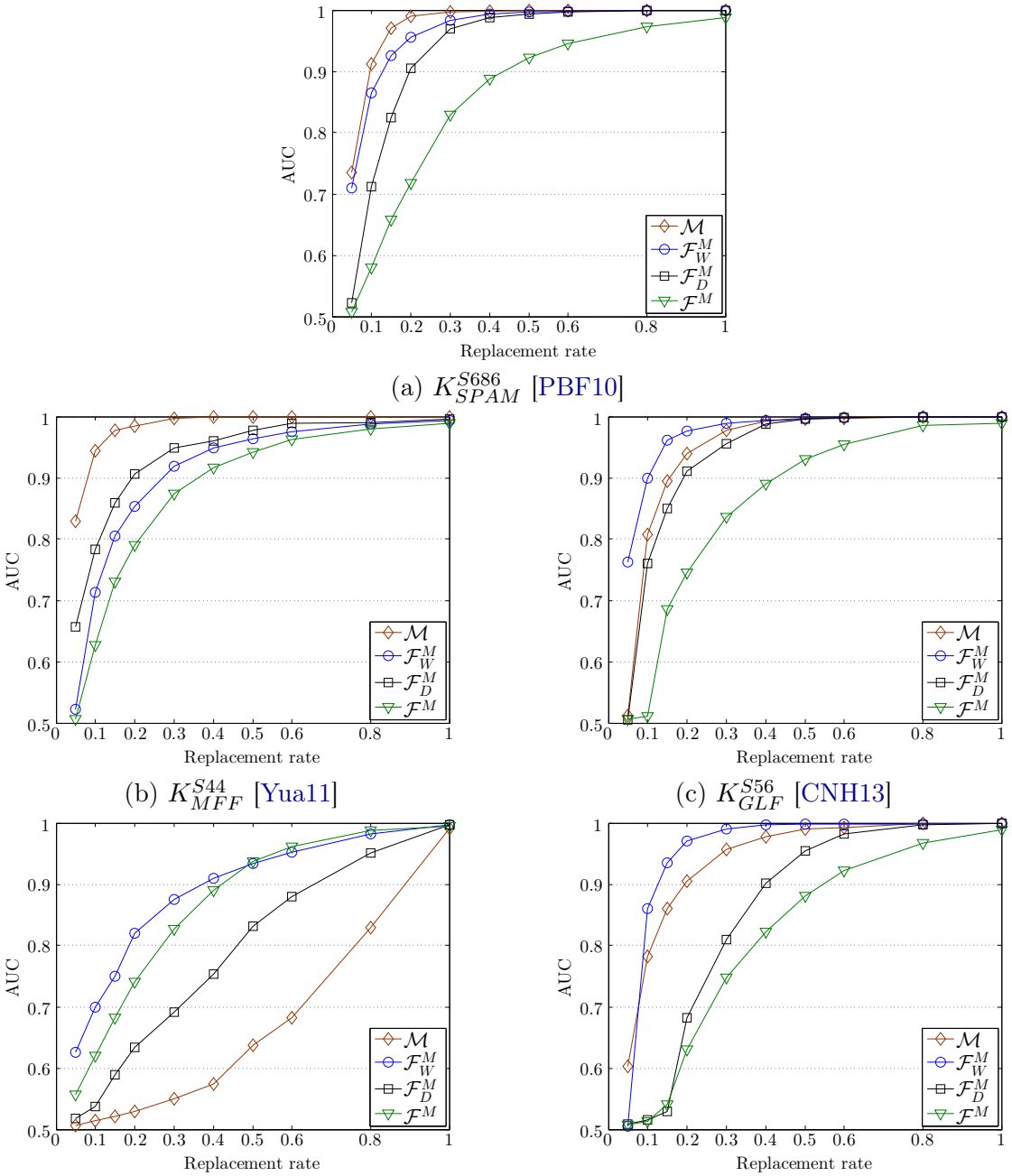
Figure 7.9: ROC curves of the MF image $\mathcal{M}$, Wu *et al.*'s anti-forensic MF image $\mathcal{F}_W^M$ [WSL13], Dang-Nguyen *et al.*'s anti-forensic MF image $\mathcal{F}_D^M$ [DN+13], and our anti-forensic MF image $\mathcal{F}^M$, against the four scalar-based median filtering detectors K_K [KF10], $\hat{K}_K$ [KF10], K_C [Cao+10], and K_Y [Yua11]. Results are obtained on MFTE dataset.

Existing median filtering forensic work either explicitly or implicitly analyzes the pixel value difference. Therefore, besides the forensic undetectability and image quality, it is also important to recover the pixel value difference statistics for median filtering anti-forensics. This can be quantitatively evaluated, via the KL divergence (see Section 2.2.3 for more descriptions) between the derivative histogram of the original image and that of the image under examination. A smaller value of KL divergence means a better resemblance between the two compared histograms. We consider four derivative filters $\mathbf{f}^1$, $\mathbf{f}^2$, $\mathbf{f}^3$, and $\mathbf{f}^4$. For each image under investigation, each type of derivative histogram is constructed and compared with that of the original image using the KL divergence. The average KL divergence values between

the original image and different kinds of images for different kinds of derivative histograms are reported in the last four rows of Table 7.2. It can be seen that the proposed method has the ability to bring the pixel value difference histogram closer to that of the original image than the two state-of-the-art methods [WSL13, DN+13]. This proves the pixel value difference statistics restoration capability of the image prior we use in the proposed optimization problem. It is also an interesting point that the proposed pixel value perturbation strategy is capable of bringing the derivative histogram of the anti-forensic MF image $\mathcal{F}^M$ even closer to that of the original image than the other version of our anti-forensic MF image $\mathcal{F}'^M$.

Concerning the SVM-based detectors, we follow the testing strategy described in Section 2.2.1.2. For a given original image, a square central part of the image is replaced by the (anti-forensic) MF image with a replacement rate around the values in $\{0.05, 0.1, 0.15, 0.2, 0.3, 0.4, 0.5, 0.6, 0.8, 1\}$ to create the image forgery. The processed images together with the original images are put together for forensic training/testing. The training is carried out using LIBSVM [CL11] with a Gaussian kernel on dataset MFTR. The parameters of the SVM classifier is obtained using a five-fold cross validation with the multiplicative grid suggested in [PBF10]. The testing is carried out on dataset MFTE. Thereafter, for each image replacement rate, each kind of (anti-forensic) MF image, and each kind of SVM-based median filtering forensic detector, a ROC curve can be plotted and an AUC value can be calculated. The AUC curve as a function of the replacement rate is plotted in Figure 7.10 by different kinds of images and different kinds of SVM-based detectors.

The effectiveness of existing SVM-based median filtering forensic detectors [PBF10, Yua11, CNH13, Kan+13, Zha+14] in discriminating (anti-forensic) MF images from the original images can be evaluated when the replacement rate is 1. From Figure 7.10, it can be seen that they all perform excellently, and are able to achieve AUC values around 1, namely perfect classification. Except for detector K_{AR}^{S10} [Kan+13], our anti-forensic MF image $\mathcal{F}^M$ achieves the best forensic undetectability against the considered SVM-based detectors among all kinds of (anti-forensic) MF images. Moreover, the performance of our forgery $\mathcal{F}^M$ is quite stable, yielding AUC values almost at the same level for the same replacement rate for different SVM-based detectors. When the replacement rate is about 0.1, our anti-forensic MF image $\mathcal{F}^M$ is able to achieve AUC values lower than 0.65 for all considered SVM-based detectors. In this case, we can safely replace a 162×162 block in a 512×512 MFTE image. We remain reserved on whether our anti-forensic MF image is able to pass off as never median filtered, since it can be detected by machine learning based forensic methods, when the replacement rate is relatively high. However, it can still find many applications in various anti-forensic scenarios, *e.g.*, image splicing and tampering with relatively low replacement rate.



(a) K_{SPAM}^{S686} [PBF10]

(b) K_{MFF}^{S44} [Yua11]

(c) K_{GLF}^{S56} [CNH13]

(d) K_{AR}^{S10} [Kan+13]

(e) K_{LTP}^{S220} [Zha+14]

Figure 7.10: The AUC value as a function of image replacement rate for different kinds of (anti-forensic) MF images, when tested using the SVM-based detectors. Results are obtained on MFTE dataset, with detectors trained on MFTR dataset.

7.4 Applications: Disguising Footprints of Both Median Filtering and Targeted Image Operation of Median Filtering Processing

Median filtering is chosen by many forgers to disguise footprints left by certain image processing operations, *e.g.*, image resampling [KR08], and JPEG compression [SL11]. Median filtering anti-forensics further tries to remove the footprints caused by the filtering itself. It is important that median filtering anti-forensics does not impair the trace removal effects of the filtering initially applied, while keeping a high quality of the processed image.

7.4.1 Hiding Traces of Image Resampling

It is a very likely scenario for a forger, *e.g.*, during a composite image forgery creation, to use the scaling operation. Thus, image resampling is often involved in creating fake images. On the forensic investigation side, Popescu and Farid [PF05] proposed a powerful resampling forensic detector (denoted as K_P), by examining the periodic correlations between pixel neighbors. Standing on the image resampling anti-forensic side, Kirchner and Röhme [KR08] developed a technique which was able to fool K_P [PF05], via attacks based on median filtering and geometric distortion.

As in [PF05], we use bicubic interpolation for image resampling, and the factors in use are randomly selected from $\{0.7, 0.8, 0.9, 1.1, 1.2\}$. The 3×3 median filtering alone (without geometric distortion applied) works well under these scaling factors, for resampling anti-forensic purposes [KR08].

For the sake of conciseness, we create different anti-forensic images from the resampled image $\mathcal{R}$ as follows:

- $\mathcal{R}^m$, median filtered with window size $s = 3$;
- $\mathcal{R}^w$, with the application of Wu *et al.*'s [WSL13] median filtering anti-forensic method to $\mathcal{R}^m$;
- $\mathcal{R}^d$, with the application of Dang-Nguyen *et al.*'s [DN+13] median filtering anti-forensic method to $\mathcal{R}^m$;
- $\mathcal{R}^f$, with the application of the proposed median filtering anti-forensic method to $\mathcal{R}^m$.

Table 7.3 reports the image quality and forensic undetectability against Popescu and Farid's resampling forensic detector [PF05] and the scalar-based median filtering detectors [KF10, Cao+10, Yua11], for different kinds of anti-forensic MF images created from the resampled image $\mathcal{R}$. Moreover, the 5 SVM-based median filtering detectors are also tested, following the experimental setup described in Section 2.2.1.2. Figure 7.11 plots the achieved

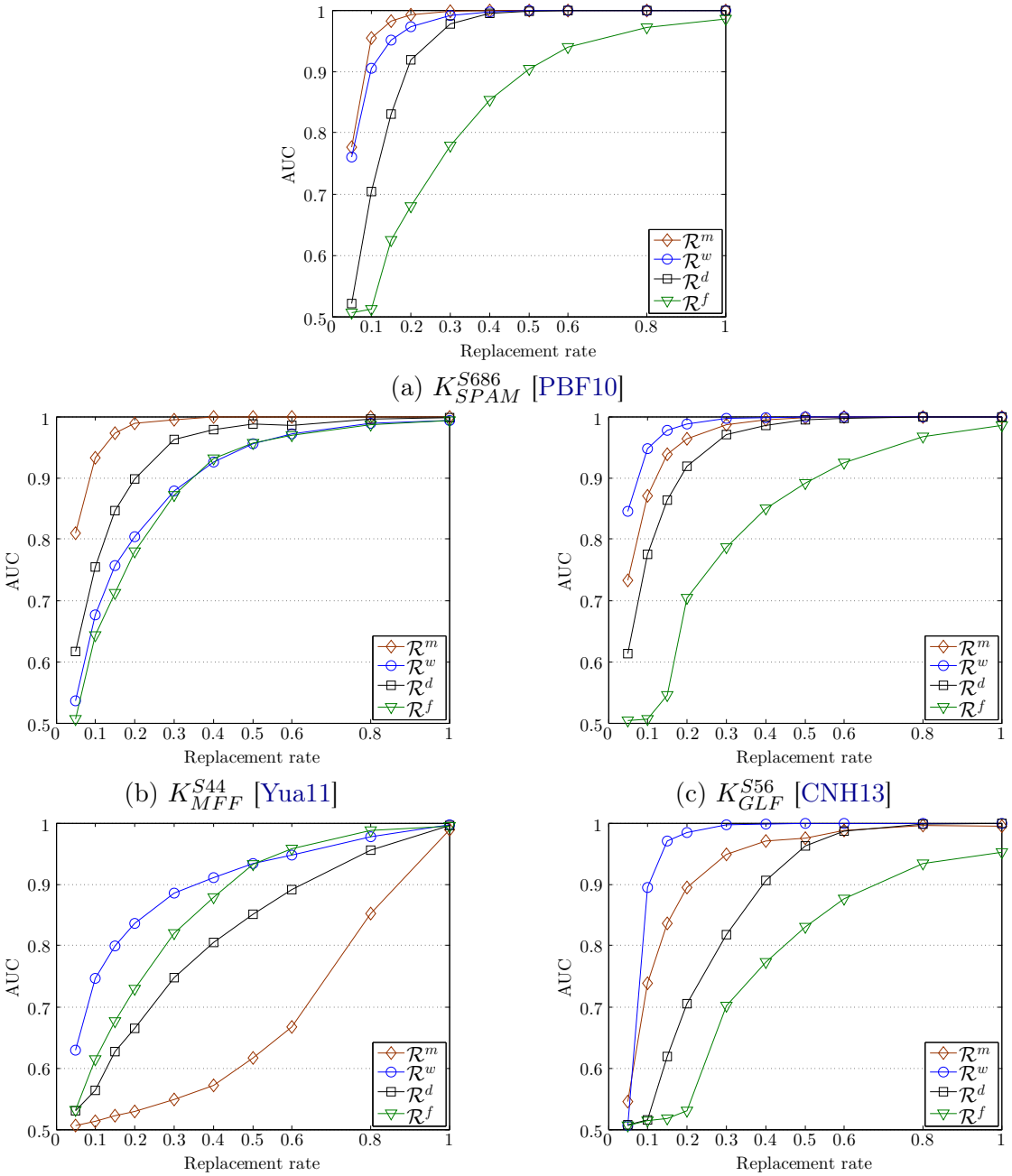
AUC values by different kinds of images, against different SVM-based median filtering forensic detectors at different replacement rates.

Table 7.3: From the 2nd to the 5th rows, the average PSNR and SSIM values are reported for different kinds of images in the resampling traces hiding application. The last 5 rows show the AUC values of different kinds of images against the resampling detector [PF05] and the 4 scalar-based median filtering detectors [KF10, Cao+10, Yua11]. Results are obtained on MFTE dataset.

		$\mathcal{R}$	$\mathcal{R}^m$	$\mathcal{R}^w$	$\mathcal{R}^d$	$\mathcal{R}^f$
Image quality	rPSNR	–	37.9091	33.7600	33.3101	37.2413
	rSSIM	–	0.9749	0.9391	0.9711	0.9894
	mrPSNR	37.9091	–	37.5393	36.3006	38.8262
	mrSSIM	0.9749	–	0.9618	0.9870	0.9896
Anti-forensic performance	K_P	0.8643	0.6818	0.9834	0.5001	0.6495
	K_K	0.3775	0.9672	0.4289	0.5129	0.5364
	$\hat{K}_K$	0.2792	0.9782	0.6113	0.4422	0.4599
	K_C	0.4834	0.9929	0.6511	0.7356	0.6386
	K_Y	0.5096	0.9984	0.3212	0.6424	0.5876

Since the resampling changes the pixel locations, it is not straightforward to calculate the image quality metric values with the original image as the reference. Hence, the resampled image (for "rPSNR" and "rSSIM") and the median filtered resampled image (for "mrPSNR" and "mrSSIM") are used as the references. Table 7.3 shows the superiority of the proposed method in keeping a good image quality. Our forgery $\mathcal{R}^f$ achieves 4.34 dB of rPSNR enhancement and 0.0443 of rSSIM gain compared with $\mathcal{R}^w$, and 4.53 dB of rPSNR improvement and 0.0219 of rSSIM increase compared with $\mathcal{R}^d$. Using the image quality evaluation metrics mrPSNR and mrSSIM, our forgery $\mathcal{R}^f$ also outperforms $\mathcal{R}^w$ [WSL13] and $\mathcal{R}^d$ [DN+13]. Moreover, both the average rPSNR and rSSIM values of $\mathcal{R}^f$ are even higher than the median filtered image $\mathcal{R}^m$.

Concerning Popescu and Farid's resampling detector K_P [PF05], it can be seen that both Dang-Nguyen *et al.*'s [DN+13] and the proposed median filtering anti-forensic methods can even reinforce the anti-forensic ability of median filtering in hiding resampling traces. This is reflected by the even lower AUC values of detector K_P, compared with that of $\mathcal{R}^m$ (see Table 7.3). However, our forgery can achieve a better overall forensic undetectability against the median filtering detectors than $\mathcal{R}^d$ [DN+13] (see results in Table 7.3 and Figure 7.11). Wu *et al.*'s forgery $\mathcal{R}^w$ can be well detected by K_P. Recall that their method is based on the noise attack to the pixel value difference. In their approach, the given image is divided into non-overlapping blocks, each of which has an anchor point as the reference of the noise attack to alter the pixel value differences. This block-wise pixel modification may invoke the appearance of new periodicity of pixel neighbor correlation, which consequently can be detected by K_P. A randomized anchor point selection may be a possible improvement to this approach, which however is beyond the scope of this thesis.



(a) K^{S686}_{SPAM} [PBF10]

(b) K^{S44}_{MFF} [Yua11]

(c) K^{S56}_{GLF} [CNH13]

(d) K^{S10}_{AR} [Kan+13]

(e) K^{S220}_{LTP} [Zha+14]

Figure 7.11: The AUC value as a function of image replacement rate for different kinds of images, when tested using the SVM-based median filtering forensic detectors, in the resampling traces hiding application. Results are obtained on MFTE dataset, with detectors trained on MFTR dataset.

7.4.2 Removing JPEG Blocking Artifacts

We have studied JPEG anti-forensics in Chapters 4-6. As known, two kinds of artifacts present in JPEG images, *i.e.*, the DCT-domain quantization artifacts and the spatial-domain blocking artifacts, which are often utilized for JPEG forensic purposes. The consistent discontinuities across 8×8 block borders, *i.e.*, the JPEG blocking artifacts, can be detected by detectors K_F [FD03], K_U^1 and K_U^2 (see Table 3.1). In one state-of-the-art JPEG anti-forensic method [Sta+10a, Sta+10b, SL11], Stamm and Liu separated the task of disguising the JPEG footprints into two steps: adding a dithering signal to smooth the DCT histogram followed by a median filtering based JPEG blocking artifact removal procedure. During the second step, they proposed a JPEG deblocking process consisting of a median filtering and a small-energy 0-mean white Gaussian noise addition (see Section 3.1.5 for more descriptions).

In order to purely study the footprint concealing capability of median filtering, here we apply the median filtering directly to JPEG images without the noise attack. The JPEG compression quality factors in use are randomly selected from $\{85, 86, \cdots, 95\}$. We find that for JPEG images with quality factors in this set, the median filtering alone can perform very well in concealing JPEG blocking artifacts.

We use the following notations to refer to different anti-forensic images created from the JPEG image $\mathcal{J}$:

- $\mathcal{J}^m$, median filtered with $s = 3$;
- $\mathcal{J}^w$, with the application of Wu *et al.*'s [WSL13] median filtering anti-forensic method to $\mathcal{J}^m$;
- $\mathcal{J}^d$, with the application of Dang-Nguyen *et al.*'s [DN+13] median filtering anti-forensic method to $\mathcal{J}^m$;
- $\mathcal{J}^f$, with the application of the proposed median filtering anti-forensic method to $\mathcal{J}^m$.

The average image quality and anti-forensic performance against JPEG blocking detectors and scalar-based median filtering detectors are reported in Table 7.4. At different replacement rates, the AUC values achieved by different anti-forensic images tested against 5 SVM-based median filtering detectors are shown in Figure 7.12. Note that "mjPSNR" and "mjSSIM" respectively stand for the PSNR and SSIM metrics, when the median filtered JPEG image is used as the reference. It can be seen that our forgery $\mathcal{F}^M$ is able to achieve a better overall forensic undetectability against the median filtering detectors than $\mathcal{J}^w$ [WSL13], and $\mathcal{J}^d$ [DN+13]. As to JPEG blocking artifacts hiding, all the three kinds of anti-forensic MF images are able to even further decrease the AUC values of detectors K_F [FD03], K_U^1, and K_U^2 [Fan+13a]. However, the price for $\mathcal{J}^w$ and $\mathcal{J}^d$ is an evident image degradation: an oPSNR loss of 3.58 dB and 3.78 dB compared to $\mathcal{J}^m$ on average, respectively. As to our forgery $\mathcal{J}^f$, not only a good overall forensic undetectability is guaranteed, but also an oPSNR gain of 0.15 dB and a slight oSSIM increase are achieved, compared with $\mathcal{J}^m$. Moreover, the mjPSNR and

mjSSIM values achieved by our forgery $\mathcal{J}^f$ are also both higher than those of $\mathcal{J}^w$ [WSL13] and $\mathcal{J}^d$ [DN+13].

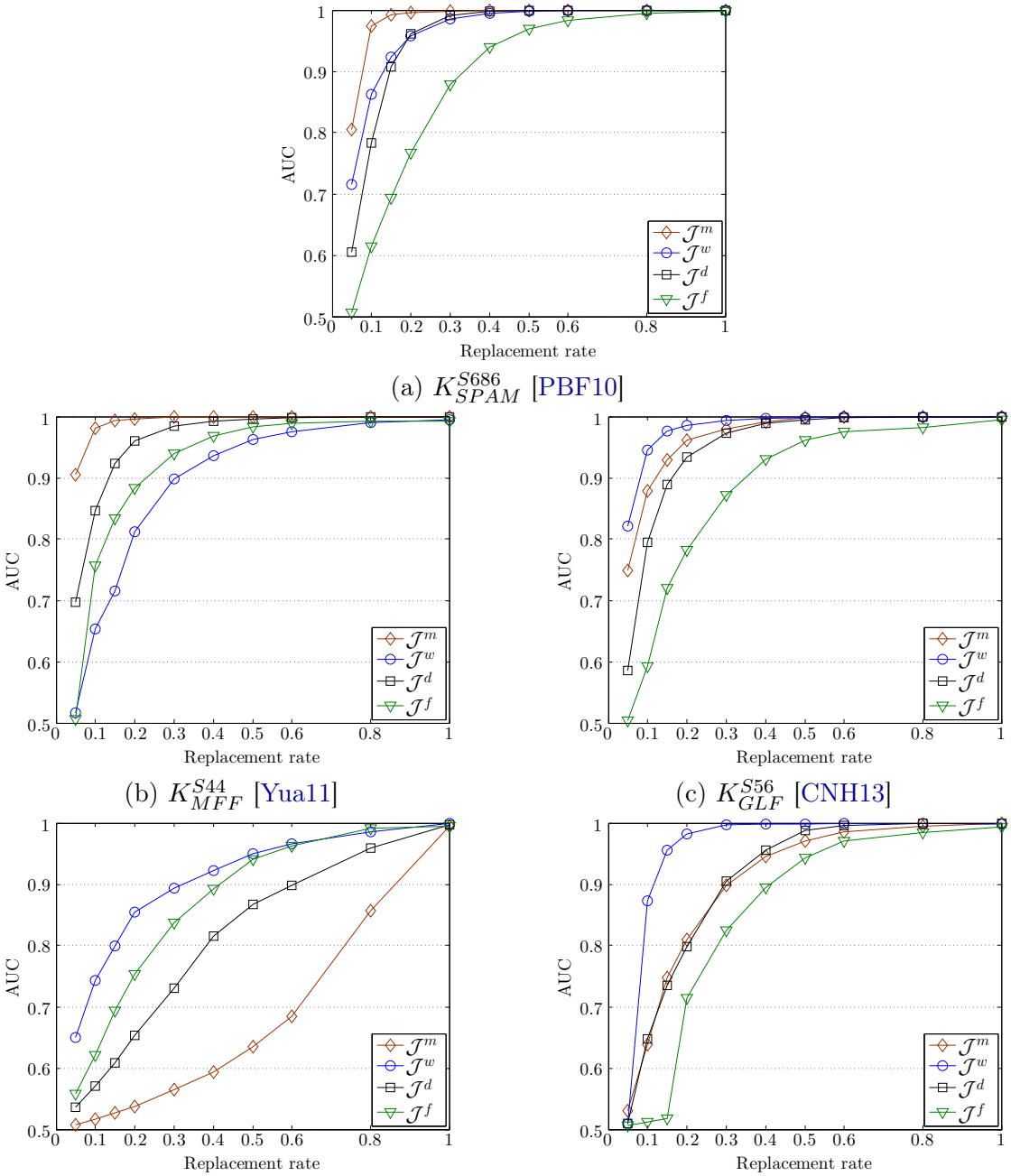
Table 7.4: From the 2nd to the 5th rows, the average PSNR and SSIM values are reported for different kinds of images in the JPEG blocking artifacts removing application. The last 7 rows show the AUC values of different kinds of images against JPEG blocking detectors [FD03] and the 4 scalar-based median filtering detectors [KF10, Cao+10, Yua11]. Results are obtained on MFTE dataset.

		$\mathcal{J}$	$\mathcal{J}^m$	$\mathcal{J}^w$	$\mathcal{J}^d$	$\mathcal{J}^f$
Image quality	oPSNR	42.9742	37.0888	33.5111	33.3101	37.2413
	oSSIM	0.9975	0.9830	0.9549	0.9711	0.9894
	mjPSNR	37.6087	–	37.4983	36.3006	38.8262
	mjSSIM	0.9838	–	0.9710	0.9870	0.9896
Anti-forensic performance	K_F	0.9930	0.6128	0.6321	0.4945	0.6003
	K_U^1	0.9871	0.7151	0.5709	0.6255	0.5168
	K_U^2	0.8937	0.5288	0.5363	0.5274	0.4754
	K_K	0.3895	0.9659	0.4109	0.5035	0.5137
	$\hat{K}_K$	0.2699	0.9786	0.6061	0.4256	0.4489
	K_C	0.3990	0.9924	0.6369	0.7230	0.6244
	K_Y	0.4985	0.9986	0.3225	0.6489	0.5976

7.5 Summary

In this chapter, we continue the research line of leveraging on image restoration to design median filtering anti-forensic method, similar to what we did in JPEG anti-forensics in Chapters 4-6. More specifically, we have proposed an image variational deconvolution framework, which can serve as an MF image quality enhancement method as well as an MF image anti-forensic approach. Indeed, a single convolution filter is used to approximate the spatially heterogeneous median filter. It may appear to be somewhat irrational but makes the problem solvable, and in practice yields good results. The image prior adopts the generalized Gaussian distribution to model image derivatives. It is a simple yet especially suitable prior for MF image quality enhancement and anti-forensics. The median filter largely alters the statistics in the pixel value difference domain, whereas the proposed prior is able to well regularize the pixel value difference histogram, which is proven in experiments by a low KL divergence value with respect to the histogram of the original image.

For anti-forensic purposes, a pixel value perturbation strategy is proposed to process the MF image in advance, with a very minor impact on the visual quality of the final forgery. Experimental results show that our anti-forensic MF image outperforms the state-of-the-art anti-forensic MF images in terms of forensic undetectability, with an even higher image quality than the MF image on MFTE dataset. Moreover, backed by experiments, the proposed median



(a) K_{SPAM}^{S686} [PBF10]

(b) K_{MFF}^{S44} [Yua11]

(c) K_{GLF}^{S56} [CNH13]

(d) K_{AR}^{S10} [Kan+13]

(e) K_{LTP}^{S220} [Zha+14]

Figure 7.12: The AUC value as a function of image replacement rate for different kinds of images, when tested using the SVM-based median filtering forensic detectors, in the JPEG blocking artifacts removing application. Results are obtained on MFTE dataset, with detectors trained on MFTR dataset.

filtering anti-forensic method is effective in applications of disguising image resampling traces and JPEG blocking artifacts.

Chapter 8

Conclusions

8.1 Summary of Contributions

In this thesis, we have presented our research work on image anti-forensics to JPEG compression and median filtering. During our study of image anti-forensics concerning image coding/processing, we notice that it to some extent shares some similarities with image restoration. Both of them aim to recover the information lost during the image degradation to the utmost. However, besides image quality, image anti-forensics has an additional indispensable goal, *i.e.*, good forensic undetectability of the anti-forensic image against forensic detectors. To this end, we introduce some advanced concepts/methods from image restoration to design new anti-forensic methods, meanwhile integrating some anti-forensic strategies/terms. Experimental results demonstrate that the proposed anti-forensic methods outperform the state-of-the-art methods in creating anti-forensic images with better forensic undetectability against existing forensic detectors as well as a higher visual quality of the processed image. Though image restoration itself is not our main research topic in this thesis, we also propose two image quality enhancement methods for JPEG compression and median filtering, respectively. They serve as the preliminary step of the proposed anti-forensic methods, but are also proven to have good performance in terms of PSNR and SSIM gains compared with the JPEG/MF image.

The contributions of this thesis are summarized as follows.

Proposing a new research line of designing image anti-forensics via image restoration: After conducting the literature review, we find that it is a common way to use simple image processing for hiding traces left by a targeted operation to design image anti-forensics. For example, median filtering is used for JPEG anti-forensics to remove blocking artifacts. Noise injection is used to hide traces of median filtering. These methods can be effective in attacking targeted forensic detectors, however can also be detected by more advanced detectors. Moreover, the resulting anti-forensic image suffers from a low visual quality, which may spontaneously cause the doubt upon its authenticity. In this thesis, we propose a new research line of designing image anti-forensics leveraging on advanced concepts/methods from image restoration. Given a degraded image processed by an irreversible image processing operation, *e.g.*, JPEG compression and median filtering, the objective of image anti-forensics is to hide the relevant image processing traces meanwhile maintaining a good image quality of the anti-forensic image. Based on the similarities between image anti-forensics and image restoration, the introduced optimization framework/terms from image restoration help to partly remove

the traces, while keeping a good (sometimes even higher) image quality compared with the given degraded image. Good anti-forensic performance of the image is achieved by adding anti-forensic terms to the optimization framework or using additional anti-forensic strategies. We follow this research line throughout this thesis for JPEG anti-forensics as well as median filtering anti-forensics. Its superiority over state-of-the-art anti-forensic methods is well proven by experimental results, with a better forensic undetectability and a higher image quality of the processed image. To the best of our knowledge, we are the first to conduct image anti-forensics following this research direction. We also hope this new research line would be useful in more image anti-forensic problems when image coding/processing is involved, *e.g.*, image resampling, contrast enhancement, *etc.*

JPEG anti-forensics using TV-based deblocking: We have proposed a JPEG anti-forensic method using the TV-based deblocking method. This is implemented by optimizing a constrained TV-based minimization problem with a TV term and a TV-based blocking measurement term. Meanwhile, the visual quality of the processed image is controlled by a modified QCS projection. Moreover, the powerful calibration based detector is defeated by minimizing a cost function close to the calibration based JPEG forensic feature. This method has good performance in spatial-domain JPEG blocking artifacts removal. Besides, the DCT-domain quantization artifacts are also largely mitigated, to the extent that the anti-forensic JPEG image is able to pass off as never compressed under the examination of quantization artifacts detectors.

Improved JPEG anti-forensics with perceptual DCT histogram smoothing: Indeed, the previously proposed TV-based JPEG deblocking method is able to achieve good forensic undetectability against all existing JPEG forensic detectors including the ones examining quantization artifacts. However, the DCT histogram is still not always well smoothed especially in the mid-frequency subbands. This weakness is not exposed by existing JPEG forensic detectors, yet may be detected by potential, more advanced forensic algorithms. In order to tackle this problem, we have proposed a four-step improved JPEG anti-forensic method with perceptual DCT histogram smoothing. With the help of the partly recovered DCT coefficients after the TV-based deblocking, an adaptive local dithering method is proposed by combining the Laplacian distribution and the uniform distribution. DCT coefficients are modified by solving a simplified assignment problem which minimizes the SSIM loss in the spatial domain. A mild second-round TV-based deblocking and de-calibration are performed to further improve the forensic undetectability with a very minor image quality loss.

JPEG image quality enhancement and anti-forensics using a sophisticated image prior model: We have proposed another JPEG anti-forensic method based on a sophisticated image prior model, which are described by steps in the following.

- *JPEG image quality enhancement with a sophisticated image prior model*: The effectiveness of using TV in JPEG anti-forensics motivates us to seek more sophisticated image prior models, with the hope to further push the performance of JPEG anti-forensics. To this end, we use the EPLL framework with the GMM as the image prior model for overlapping image patches. The spatial-domain compression noise is modeled using

the multivariate 0-mean Gaussian model, for which 64 kinds of covariance matrices are learned for each quality factor. Therefore, a JPEG image quality enhancement method has been proposed by minimizing the cost function considering the above two terms. The quality enhanced JPEG image can be obtained by solving an optimization problem using one step of approximate MAP estimation. This method achieves good image quality gain for JPEG images, especially for very low bit-rate compression.

- *Calibration based non-parametric DCT histogram smoothing*: Compared with our previously proposed perceptual DCT histogram smoothing method, we also have proposed another calibration based DCT histogram smoothing procedure without using any statistical model. Based on the translation invariance of image statistics and the effectiveness of calibration, the DCT-domain quantization noise can be estimated by slightly cropping the quality enhanced JPEG image and recompressing the calibrated image. The DCT histogram is smoothed by adding the estimated DCT-domain quantization noise back to the quality enhanced JPEG image. This new, non-parametric DCT histogram smoothing method does not achieve a better overall performance but has better performance in the low-frequency DCT subbands than the perceptual DCT histogram smoothing method. Moreover, the former has a much lower computation cost than the latter.

- Forensic undetectability is considered by minimizing a cost function with an image fidelity term, an EPLL-based image prior term, and several anti-forensic terms inspired from existing forensic algorithms.

To recapitulate, the proposed quality enhancement method is proven by experiments on 4 classical test images and on UCIDTest dataset to improve the visual quality of the JPEG image. Indeed, the proposed anti-forensic method needs improvement, however still outperforms Stamm *et al.*'s method [Sta+10a, Sta+10b, SL11] in terms of forensic undetectability and visual quality of the processed image.

Median filtered image quality enhancement and anti-forensics via variational deconvolution: We have proposed an MAP-based image variational deconvolution framework for both median filtered image quality enhancement and anti-forensics. The median filtering process is approximated using a convolution kernel. Besides, we also hope the processed MF image is still to some extent close to the MF image, retaining some median filtering effects such as denoising or artifacts hiding for other image operations. As to the image prior, the generalized Gaussian distribution is used to model the pixel value difference whose statistics changes largely after the image is median filtered. Based on the above consideration, the proposed minimization cost function is formed by a convolution term, a fidelity term with respect to the MF image, and a prior term. The quality enhanced MF image can be obtained by solving the proposed minimization problem, with good visual quality gain compared to the MF image. With another parameter setting and an additional pixel value perturbation procedure, the anti-forensic MF image is also generated. It is able to achieve a better forensic undetectability against existing median filtering forensic detectors and a higher image quality, compared with the state-of-the-art anti-forensic MF images.

In summary, the main characteristic/novelty of this thesis is to introduce advanced

concepts/methods from image restoration to design image anti-forensics, when image coding/processing is involved. More specifically, we have designed JPEG anti-forensics and median filtering anti-forensics, leveraging on the following elements from image restoration: the TV, the EPLL framework with GMM as the prior model for overlapping image patches, the MAP-based image deconvolution framework. For a given specific image anti-forensic problem, some anti-forensic terms/strategies are integrated for defeating existing detectors. For JPEG compression, a TV-based blocking measurement term is used for JPEG deblocking purposes. Moreover, two DCT histogram smoothing methods are also proposed to remove DCT-domain quantization artifacts: a perceptual DCT histogram smoothing procedure conducted by building an adaptive local dithering model and solving a simplified assignment problem, and a non-parametric DCT histogram smoothing method based on calibration. Some anti-forensic terms inspired from existing forensic algorithms are also integrated into the MAP-based optimization framework, for fooling relevant forensic detectors. As to median filtering, the image prior model is specifically chosen to regularize the image pixel value differences, which are either explicitly or implicitly examined in existing forensic algorithms. Moreover, a pixel value perturbation procedure is also proposed to further improve the forensic undetectability with very minor visual quality loss of the processed image.

8.2 Perspectives

In the short term, the perspectives of this thesis include the performance improvement of anti-forensics, and applying the research line of leveraging on image restoration to design algorithms to solve other suitable image anti-forensic problems. Please see the following for more detailed descriptions.

Improvement of JPEG anti-forensics to generate the anti-forensic image with an even higher quality than the JPEG image: In Chapters 4-5, the TV is employed from image restoration to JPEG anti-forensics. The TV can be considered as a simple but effective image prior. Chapter 6 presents the natural follow-up work on JPEG anti-forensics, where a more sophisticated image prior model than the TV is used. However, the new JPEG anti-forensic method does not outperform the ones based on the TV. Analysis and possible underlying reasons are provided in the end of Sections 6.3 and 6.4. However, we can still see the effectiveness of the proposed JPEG quality enhancement method based on the EPLL framework with the GMM as the image prior model. A challenging but very interesting JPEG anti-forensic problem is whether we are able to create anti-forensic JPEG image with an even higher visual quality than the JPEG image. A possible research direction could be the combination of the JPEG image quality enhancement, the TV-based deblocking, and the perceptual smoothing of DCT histogram. In fact, a similar goal has already been achieved in Chapter 7 for median filtering anti-forensics: the anti-forensic MF image is created with an even higher visual quality than the MF image on MFTE dataset.

Development of other image anti-forensic methods leveraging on image restoration: In this thesis, we choose to work on JPEG compression and median filtering anti-

forensics. In image processing, there are various other image operations whose anti-forensics may also be formulated as ill-posed inverse problems. For example, image resampling anti-forensics may share some similarities with superresolution [PPK03]. Image contrast enhancement anti-forensics may be related to the estimation of pixel brightness transform [ZL14]. Ideally, a good performance of the anti-forensic method can be achieved if these concepts/methods from image restoration are well combined with some anti-forensic terms/strategies.

In the long term, one of the ultimate goals of image forensics and anti-forensics is to develop universal methods without targeting at a specific (set of) anti-forensic methods or forensic algorithms. This is to avoid the endless cat-and-mouse chasing between forensics and anti-forensics. Though the research work of image (anti-)forensics has been going on for over a decade, there are very few universal methods. Probably the only universal image anti-forensic method is the one proposed by Barni *et al.* [BFT12]. However, they also make the assumption that the forensic detectors examine the first-order statistics only. We are aware that the proposed anti-forensic methods all belong to the "targeted" category according to Böhme and Kirchner's [BK13] classification of image anti-forensics (see Section 2.1.5). However, we believe the proposed anti-forensic methods have the potential to be further generalized for developing universal image anti-forensics.

Universal image anti-forensics: In Chapters 6-7, the MAP-based optimization problems are proposed to perform JPEG anti-forensics and median filtering anti-forensics, respectively. The universality of the variational image restoration framework can be seen through these two problems. When it concerns image coding/processing, the objective of anti-forensics is to create anti-forensic images that are as "natural" as possible. To this end, the "natural" image generation problem can be formulated as an image restoration problem and use statistical methods, such as MAP estimate, to obtain an image that appears never processed. Under this framework, a good enough image prior model is indispensable. Moreover, the likelihood term, describing the image degradation process caused by a certain image operation, should vary according to different anti-forensic problems. The "universality" here is that the anti-forensic framework is generic, which can be used to hide the traces left by various image processing operations.

Besides, some other open research problems closely related to this thesis are listed as follows:

- Is it possible to design a single step attack for JPEG anti-forensics, considering multiple forensic detectors working in different domains?
- How can we estimate the spatially heterogeneous convolution kernel for median filtering?
- Is it possible to create anti-forensic image, so that it as a whole can pass off as never processed by machine learning based forensic detectors trained on the original images and anti-forensic images?

These questions also point out a few interesting future research directions in the vast domain of image anti-forensics. Moreover, concerning the close link between digital images and digital

videos, the proposed image anti-forensic methods to JPEG compression and median filtering may also contribute to video anti-forensics.

In this thesis, we conduct the study on JPEG anti-forensics and median filtering anti-forensics following the research line of leveraging on image restoration. These two specific image anti-forensic problems just constitute a small fraction of image anti-forensics. We are also aware that this newly formed research line may be only applied to image anti-forensic problems, where image coding/processing is involved. For other image anti-forensic problems, *e.g.*, attacking physically based or geometric-based [Far09a] forensic algorithms, this research line may be invalid. For example, in our lighting-based image anti-forensic work [Fan+12], a very different research line is followed.

In conclusion, for image anti-forensic problems involving image coding/processing, we believe that natural image statistics is essential in creating the anti-forensic image with good forensic undetectability as well as a high visual quality. To this end, some advanced concepts/methods from image restoration are introduced to combine with anti-forensic terms/strategies in this thesis for JPEG anti-forensics and median filtering anti-forensics. From our image anti-forensic research work conducted following this research line, we can catch a glimpse of the enormous potential of natural image statistics in image anti-forensics, image forensics and image restoration. Böhme and Kirchner [BK13] also point out the importance of natural image statistical model in the battle between forensics and anti-forensics. Therefore, despite of much room for improvement, we hope that our work can serve as a good start for future research on a broad range of image forensic and anti-forensic problems.

Appendix A

Résumé en Français

Sommaire

A.1 Introduction . . . 160

A.1.1 Pouvez-vous croire vos yeux ? . . . 160

A.1.2 Anti-criminalistique en images numériques . . . 163

A.1.3 Objectifs et contributions . . . 163

A.1.4 Organisation du résume . . . 166

A.2 Préliminaires . . . 167

A.2.1 Classification de la criminalistique et l'anti-criminalistique d'image . . . 167

A.2.2 Métriques d'évaluation . . . 169

A.2.3 Ensembles d'images naturelles . . . 172

A.2.4 Algorithmes pertinents d'optimisation . . . 173

A.3 État de l'art en (anti-)criminalistique de compression JPEG et de filtrage médian . . . 174

A.3.1 (Anti-)Criminalistique de compression JPEG . . . 174

A.3.2 (Anti-)Criminalistique du filtrage médian . . . 176

A.4 Anti-criminalistique de compression JPEG basée sur la TV . . . 177

A.4.1 Introduction et motivation . . . 177

A.4.2 Déblocage JPEG en minimisant un problème contraint basé sur la TV . . . 178

A.4.3 Décalibrage . . . 179

A.4.4 Quelques résultats expérimentaux . . . 179

A.5 Anti-criminalistique de compression JPEG avec un lissage perceptuel de l'histogramme DCT . . . 181

A.5.1 Introduction et motivation . . . 181

A.5.2 Lissage perceptuel de l'histogramme DCT . . . 183

A.5.3 Quelques résultats expérimentaux . . . 185

A.6 Amélioration de qualité et anti-criminalistique de l'image JPEG basée sur un modèle d'image avancé . . . 187

A.6.1 Introduction et motivation . . . 187

A.6.2 Amélioration de qualité de l'image JPEG . . . 187

A.6.3 Anti-criminalistique de compression JPEG . . . 188

A.7 Amélioration de la qualité et anti-criminalistique de l'image filtrée par le filtre médian à l'aide d'une déconvolution variationnelle d'image 190

A.7.1 Introduction et motivation . . . 190

A.7.2 Déconvolution variationnelle d'image . . . 191

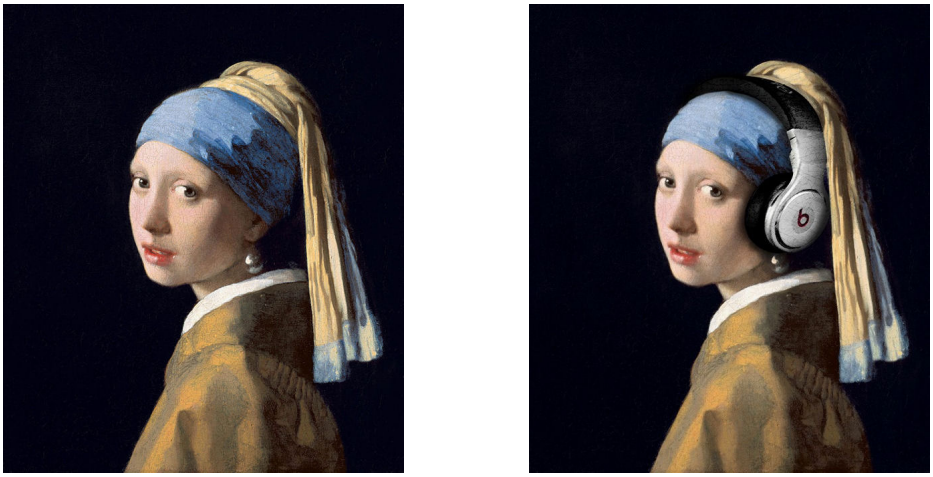
A.7.3 Amélioration de qualité de l'image MF . . . 193

A.7.4 Anti-criminalistique de filtrage médian 195
A.8 Conclusions et perspectives 197
A.8.1 Résumé des contributions 197
A.8.2 Perspectives 201

A.1 Introduction

A.1.1 Pouvez-vous croire vos yeux ?

Voir c'est croire.
Une image vaut mille mots.



(a) Originale (b) Modifiée[21]

Figure A.1: (a) L'original : *La Jeune Fille à la Perle*, Vermeer (*ca.* 1665), et (b) la version falsifiée dans laquelle les écouteurs, un produit de la technologie moderne, ont été intégrés numériquement par l'utilisateur *bigchopper* du site *Worth1000*.

On a coutume de dire que *voir c'est croire*, ou qu'*une image vaut mille mots*. C'est peut-être pourquoi l'image est l'un des médias les plus couramment utilisés sur Internet. Selon le rapport annuel Mary Meeker sur les tendances d'Internet [Mar], plus de 1,8 milliards de photos sont téléchargées et partagées sur Internet chaque jour ! Les images numériques sont littéralement omniprésentes, et on leur fait souvent aveuglément confiance. Cependant, cette confiance est constamment remise en cause par le développement de caméras à haute définition et de logiciels de retouche puissants. Dans la figure A.1-(b), *La Jeune Fille à la Perle*, peint

[21] Cette image était téléchargée depuis : `http://www.worth1000.com/entries/740270/girl-with-the-beats`.

par Johannes Vermeer au XVII^{ème}s., le personnage semble apprécier la musique provenant d'écouteurs pourtant commercialisés au XXI^{ème}s.

Aujourd'hui encore plus qu'hier, nous avons besoin de voir plus profondément qu'avec nos yeux pour démêler le vrai du faux dans une image.

Créer de fausses images visuellement plausibles est devenu de moins en moins difficile, et notre confiance dans les images numériques est progressivement érodée par le développement des technologies modernes de l'information car des images falsifiées, pouvant facilement tromper les yeux humains, apparaissent de plus en plus fréquemment. Malheureusement, ce n'est pas toujours dans un but aussi innocent que le pur amusement visiblement ressenti par l'auteur de la figure A.1-(b). La falsification d'image peut aussi et surtout être malveillante, tant dans la sphère publique que dans la vie privée – et la frontière toujours plus floue entre les deux souligne l'acuité du problème. *Fourandsix Technologies, Inc.* maintient une galerie d'images[22], qui recense les faux les plus notables travers l'Histoire. Parmi ces faux, certains étaient assez dommageables pour causer de sévères pertes financières, ou même impacter négativement l'ensemble de la société.

Le doute qui devrait toujours légitimement entacher les images numériques a donné naissance à la **criminalistique**, qui essaye de rétablir une certaine confiance dans les images numériques. Les objectifs principaux de la criminalistique sont d'analyser une image numérique de manière à détecter si elle est authentique ou non, d'identifier son origine, de retracer l'histoire des traitements qu'elle a subis, ou de révéler les détails cachés invisibles pour des yeux humains [Fou].

Au cours de la dernière décennie, les chercheurs ont proposé différentes techniques en criminalistique d'image. Dans un premier temps, le tatouage fragile d'image était le choix le plus populaire pour l'authentification d'image. Dans la littérature, le tatouage fragile est en effet considéré comme relevant de la criminalistique *active* [Con11]. Le tatouage intègre les informations d'authentification dans l'image quand elle est capturée ou avant sa transmission. Par la suite, l'image peut être authentifiée si le filigrane extrait correspond à celui embarqué. L'image sera considérée comme fausse si l'extraction de filigrane échoue ou si l'information extraite ne correspond pas à celle qui avait été insérée. Dans ce scénario, un dispositif spécial d'acquisition d'image est nécessaire. De fait, l'idée de la caméra de confiance, équipée d'un système de tatouage, a été proposée dès 1993 [Fri93]. Cependant, son déploiement dans l'industrie a rencontré de nombreuses difficultés, qui restent actuellement encore à résoudre. Tout d'abord, il est apparemment difficile pour les différents fabricants d'appareils de prise de vue de parvenir à un accord sur un protocole standard commun. Par ailleurs, les consommateurs peuvent trouver qu'il est inacceptable de diminuer la qualité visuelle de l'image par l'insertion d'un filigrane. En outre, si le système de tatouage à l'intérieur de la caméra devait un jour être piraté, il deviendrait obsolète du jour au lendemain sans possibilité aisée d'assurer une quelconque rétro-compatibilité avec une nouvelle version corrigeant la faille de sécurité. Un exemple de tentative de réalisation de ce type de caméra dans l'industrie est l'Aigo V80PLUS [Aig], commercialisée par *Beijing Huaqi Digital Information Technology Co.,*

[22]Disponible à la page `http://www.fourandsix.com/photo-tampering-history/`.

Ltd en 2005. Dans cet appareil, un système de filigrane numérique est inclus, qui insère les informations d'authentification dans l'image au moment de l'enregistrement. Pourtant, l'usage de la caméra de confiance reste marginal, en raison des préoccupations décrites précédemment.

Conscient des limites de la criminalistique active, les chercheurs tournent progressivement leur attention vers la criminalistique *passive* [Far09a, Con11]. En comparaison avec l'authentification d'image par tatouage fragile, la criminalistique passive vise à évaluer de manière aveugle l'authenticité d'une image donnée, sans aucune autre information *a priori* (comme peut par exemple l'être un filigrane). L'hypothèse ici est que si l'on peut facilement créer de fausses images sans laisser de traces visuelles, la falsification d'image perturbera par contre très probablement les propriétés statistiques intrinsèques de l'image authentique. Par conséquent, nous choisissons de détecter une manipulation d'image en examinant l'incohérence ou la déviation de statistiques sous-jacentes d'une image. Dans la littérature, la " criminalistique passive " est souvent appelée " criminalistique ". Dans la suite de ce document, nous omettrons également l'épithète pour des raisons de concision. De la même manière, nous omettrons souvent le fait qu'il s'agisse ici systématiquement de criminalistique *d'image*, les éventuelles exceptions seront rendues évidentes par le contexte.

La criminalistique en imagerie numérique est devenue un sujet de recherche important au cours des dernières années. La figure A.2 montre le nombre annuel de publications depuis 2000 de l'IEEE [Iee] sur le sujet. Dans cette figure, la barre grise représente les publications avec le mot-clé " criminalistique ", alors que la barre hâchurée en gris clair se rapporte aux publications avec les mots-clés " criminalistique " et " image ". Si la criminalistique en général a reçu une attention croissante dans la dernière décennie, sa spécialisation pour l'image représente à elle seule environ 40% du total.

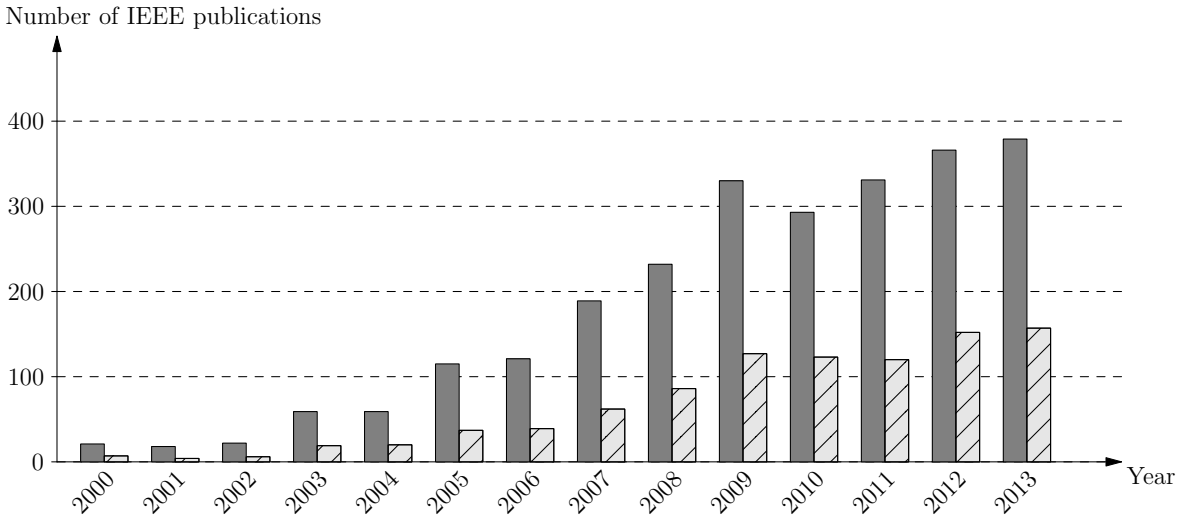


Figure A.2: Nombre annuel de publications de l'IEEE dont mots-clés incluent " criminalistique " (la barre grise), et à la fois " criminalistique " et " image " (la barre hâchurée en gris clair).

A.1.2 Anti-criminalistique en images numériques

Les deux faces d'une même pièce.

De même que la cryptographie (*resp.* la stéganographie) trouve son pendant nécessaire avec la cryptanalyse (*resp.* la stéganalyse), le "mauvais génie" de la criminalistique est l'anti-criminalistique. L'**anti-criminalistique** consiste donc en l'étude des limites des méthodes criminalistiques, dans le but ultime de contribuer à leur amélioration [BK13]. À ce titre, cet exercice est primordial et doit guider toute analyse du problème. L'objectif assigné à l'anti-criminalistique est alors d'effectuer un traitement sur l'image falsifiée, spécifiquement destiné à atténuer les traces laissées par la falsification de sorte qu'elle ne sera plus détectée par les méthodes criminalistiques.

Le développement de la criminalistique en est encore à son stade précoce, et l'anti-criminalistique est un sujet encore plus récent [KB07]. Dans la littérature, les publications sur la criminalistique sont d'ailleurs beaucoup plus nombreuses que celles relevant de l'anti-criminalistique. En outre, les méthodes anti-criminalistiques actuelles utilisent souvent des traitements simples pour dissimuler les traces laissées par une opération usuelle participant de la falsification, par exemple en utilisant un filtrage *ad-hoc* pour dissimuler les artefacts de compression JPEG [SL11], ou encore en ajoutant du bruit pour masquer ceux induits plus généralement par une opération de filtrage linéaire sur l'image [WSL13]. Il se trouve que ce type de méthodes anti-criminalistique peut réussir à tromper les détecteurs criminalistiques *ad-hoc*, mais elles peuvent être détectées par des détecteurs plus avancés. Par ailleurs, l'image anti-criminalistique créée par ces méthodes souffre souvent d'une qualité médiocre. C'est une question centrale, car une image de faible qualité (par exemple, une image floue/bruitée) peut spontanément éveiller les soupçons sur son authenticité.

En résumé, l'anti-criminalistique d'image a un double objectif: assurer une bonne indétectabilité criminalistique, ainsi qu'une haute qualité d'image [KR08]. Si ces deux buts devenaient antagonistes, le contexte de ce travail imposerait naturellement de privilégier l'indétectabilité sur la qualité.

A.1.3 Objectifs et contributions

Dans cette thèse, nous nous plaçons du côté de l'*anti-criminalistique*, et nous nous concentrons sur la *compression JPEG* et le *filtrage médian*. À partir d'une image compressée en JPEG ou filtrée par le filtre médian, si nous pouvons réussir à créer une image qui ressemble à celle non traitée, alors ce sera une tâche relativement facile que de falsifier l'histoire des traitements d'une image, ou même de changer la sémantique portée par l'image. À cette fin, nous employons certains aspects de la *restauration d'image*, en même temps que nous leur intégrons certains termes/stratégies spécifiquement dédiés à l'anti-criminalistique. De cette façon, nous espérons créer des images anti-criminalistiques avec un bon équilibre entre l'indétectabilité criminalistique et la qualité de l'image falsifiée.

A.1.3.1 Compression JPEG et filtrage médian

Tout d'abord, nous choisissons de mener nos recherches sur l'*anti-criminalistique de compression JPEG*, parce que le format JPEG est probablement le format d'image le plus commun en usage aujourd'hui sur Internet, et à ce titre souvent étudié dans de nombreux scénarios en criminalistique. Selon des statistiques sur l'utilisation des formats d'image sur Internet, au 8 décembre en 2014, le format JPEG était le plus utilisé, avec une utilisation de 68.7%, tous sites confondus [W3t].

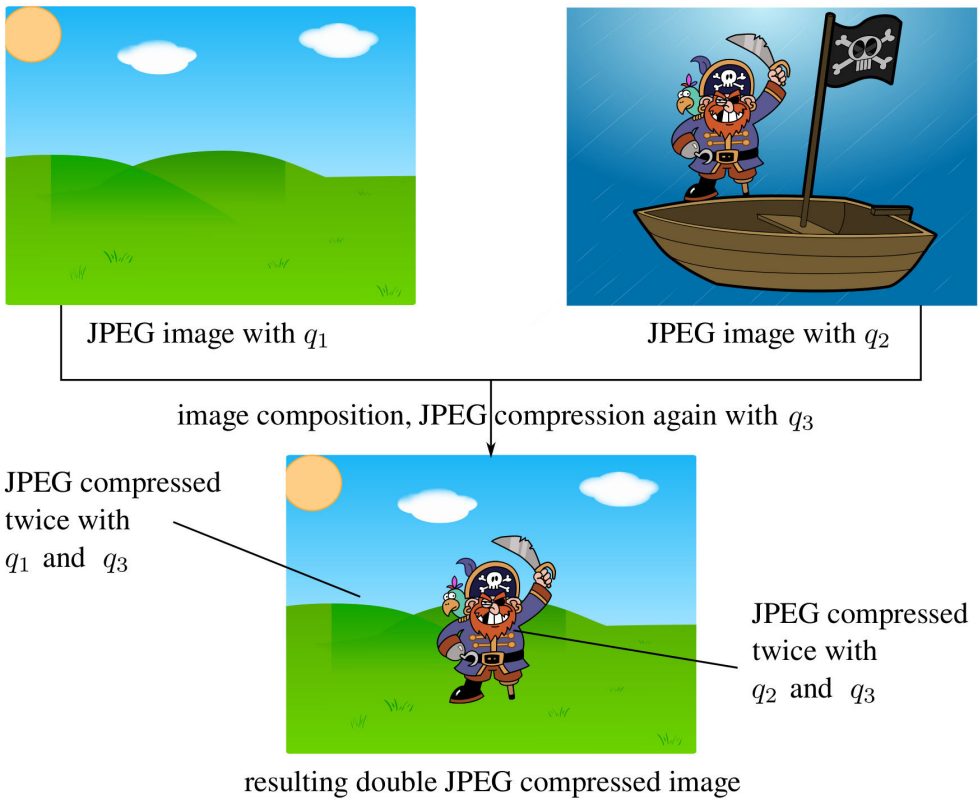


Figure A.3: Illustration de créer une image composite en JPEG. Ici, q_1, q_2 et q_3 sont trois probablement différents facteurs de qualité JPEG.

Pour illustrer l'intérêt d'étudier l'anti-criminalistique JPEG, nous nous appuyons sur le scénario suivant, illustré dans la figure A.3 : afin de faire croire que quelqu'un a participé à un certain événement, le faussaire créera très probablement une image composite avec la scène et la victime à partir de deux images JPEG avec différents facteurs de qualité[23]. L'image composite est enfin compressée en JPEG une dernière fois, avec un autre facteur de qualité JPEG, pour publication. Une modification habilement effectuée peut ne laisser aucune trace visible aux yeux humains, mais l'image falsifiée pourrait être confondue par la détection de différents artefacts de double compression JPEG présents dans les différentes zones de l'image composite [BP12b].

[23] Le facteur de qualité JPEG est un nombre entier entre 1 et 100. Plus le facteur de qualité JPEG est élevé, meilleure est la qualité de l'image, et plus la taille du fichier JPEG est grande.

D'une manière générale, l'objectif de l'anti-criminalistique JPEG est de *supprimer toute trace éventuelle de compression JPEG*, de sorte que l'image JPEG anti-criminalistique a l'air d'une image originale et non compressée. C'est typiquement ce que ferait un faussaire *réellement* habile, pour dissimuler au mieux son forfait, juste avant la dernière étape de compression avant publication [SL11].

Dans des travaux récents d'anti-criminalistique (sur la compression JPEG [SL11] ainsi que sur le rééchantillonnage [KR08]), le filtre médian est utilisé pour masquer efficacement les différentes de traitements effectués sur l'image. Cependant, le filtrage médian, vu comme un débruitage d'image et un opérateur de lissage, laisse des traces dans l'image. Elles peuvent être détectées par des méthodes criminalistiques de filtrage médian. La présence de traces de filtrage médian, non seulement donne à penser que l'image a été préalablement filtrée par le filtre médian, mais aussi implique la possibilité que d'autres opérations de traitement d'image ont pu être appliquées à l'image. Par conséquent, il est primordial d'étudier l'*anti-criminalistique du filtrage médian*, qui constitue ainsi le second sujet de recherche de cette thèse.

A.1.3.2 Anti-criminalistique d'image et restauration d'image

Dans une certaine mesure, l'anti-criminalistique d'image partage quelques similitudes avec la restauration d'image. En codage ou traitement d'image, leur objectif à tous les deux consiste à approcher au mieux les informations perdues pendant le processus de dégradation d'image, souvent en résolvant un problème mal posé. Notons d'emblée que pour certains scénarios anti-criminalistiques extrêmes (par exemple, la modification du processus physique ou géométrique [Far09a] de génération de l'image), cette similitude ne tient plus. Néanmoins, l'étude de ces scénarios nous semble relever davantage des effets spéciaux au cinéma, et nous nous concentrons principalement sur la compression JPEG et le filtrage médian. Par rapport à la restauration d'image (et à son seul objectif d'amélioration de la qualité de l'image), l'anti-criminalistique d'image a un autre objectif, encore plus impérieux, *i.e.* : être indétectable par la criminalistique.

Fondée sur des similitudes entre l'anti-criminalistique et la restauration d'images, cette thèse vise à créer, à partir d'une image compressée en JPEG ou filtrée par le filtre médian, une image " naturelle " qui semblera n'avoir jamais été traitée. À cette fin, l'estimateur du MAP (ou une de ses variantes) sera employé. Nous chercherons donc la meilleure image falsifiée, au double sens de l'indétectabilité et de la qualité, et cette recherche, comme il est de mise pour une estimation du MAP dans le cas général, fera intervenir plusieurs méthodes d'*optimisation numérique*.

A.1.3.3 Méthodologie

À la différence des méthodes anti-criminalistiques de l'état de l'art basées sur le traitement d'image simple [SL11, WSL13], cette thèse propose une nouvelle piste de recherche pour

falsifier des images. Nous proposons des méthodes anti-criminalistiques sophistiquées visant la compression JPEG et le filtrage médian, en nous appuyant sur des concepts/outils avancés de la restauration d'image, les statistiques d'image naturelle et l'optimisation numérique.

Les effets de bloc dans le domaine spatial et les artefacts de quantification dans le domaine DCT sont deux indices bien connus d'une compression JPEG [FD03]. Dans cette thèse, les méthodes suivantes sont développées pour l'anti-criminalistique JPEG :

- Tout d'abord, une minimisation contrainte basée sur la variation totale (TV) [ADF05] est employée pour éliminer les effets de bloc de la compression JPEG. En outre, afin d'assurer une haute qualité pour l'image falsifiée en bout de chaîne, une projection de type QCS [RS05] est utilisé.
- Pour supprimer les artefacts de quantification dans le domaine DCT, une méthode de lissage perceptuel de l'histogramme DCT est proposée, utilisant une loi de Laplace locale et de l'information en partie récupérée dans le domaine DCT par le déblocage TV précédemment proposé.
- Afin d'étudier l'impact d'un modèle d'image plus avancé que la TV sur l'anti-criminalistique JPEG, nous considérons aussi une mixture de gaussiennes (GMM) pour modéliser les *patchs* d'image [ZW11] allié à un terme de vraisemblance modélisant le processus de compression JPEG [RS05]. Par ailleurs, nous proposons également une nouvelle méthode non-paramétrique afin de lisser l'histogramme DCT, basée cette fois sur le calibrage [FGH02].

Quant à l'anti-criminalistique du filtrage médian, cette thèse propose une déconvolution variationnelle d'image, dans une certaine mesure inspiré par [KF09, KTF11]. La fonction de coût de ce problème d'optimisation est composée d'un terme de convolution qui approche le processus de filtrage médian, d'un terme de fidélité d'image qui conserve l'image traitée dans une certaine mesure proche de l'image qui a subi le filtrage médian, et d'un terme d'a priori basé sur une gaussienne généralisée qui régularise la dérivée de l'image obtenue.

Afin de valider l'efficacité des méthodes anti-criminalistiques proposées pour la compression JPEG et le filtrage médian, nous effectuons des tests à grande échelle. Les méthodes que nous proposons surpassent les méthodes anti-criminalistiques de l'état de l'art, à la fois au regard de leur capacité à résister aux différents criminalistiques simultanément, et de la qualité de l'image anti-criminalistique produite.

A.1.4 Organisation du résume

Le reste de ce résumé est organisé de la manière suivante :

La section A.2 introduit quelques notions de base sur la criminalistique et l'anti-criminalistique d'image, y compris la classification, les métriques d'évaluation, les ensembles

d'image utilisés dans les tests de cette thèse, et les méthodes d'optimisation numérique qui seront utilisées en résolvant les problèmes proposés.

La section A.3 présente les méthodes criminalistiques/anti-criminalistiques dans l'état de l'art sur la compression JPEG et le filtrage médian. Les méthodes de criminalistique constituent les détecteurs cibles nous attaquerons, alors que les méthodes d'anti-criminalistique constitueront nos concurrents pour les comparaisons expérimentales.

La section A.4 propose une méthode de déblocage de compression JPEG, en optimisant un problème de minimisation contrainte basée sur la TV, dont la fonction de coût est composé d'un terme de TV et d'un terme mesurant le blocage basé sur la TV. Outre un bon effet de déblocage, l'image JPEG anti-criminalistique atteint également une relativement bonne indétectabilité criminalistique, même contre les détecteurs examinant les artefacts de quantification. Cependant, ces artefacts de quantification dans le domaine DCT existent, et ils peuvent potentiellement être utilisés par d'autres détecteurs. Par conséquent, cette méthode sera encore améliorée dans la section A.5.

La section A.5 décrit une méthode améliorée pour l'anti-criminalistique JPEG basée sur le travail de la section A.4. Les artefacts de quantification restants dans le domaine DCT après le processus de déblocage TV sont explicitement éliminés par une procédure de lissage perceptuel de l'histogramme DCT.

La section A.6 présente une méthode visant à améliorer la qualité de l'image falsifiée, et qui dans le cadre du MAP, utilise une GMM pour modéliser les *patchs* d'image. Afin de produire l'image anti-criminalistique, un lissage de l'histogramme DCT est effectué en utilisant une méthode non-paramétrique basée sur le calibrage.

La section A.7, cette fois dans le cadre de l'anti-criminalistique du filtrage médian, propose une déconvolution variationnelle d'image qui modélise la différence de valeurs de pixels en utilisant la loi gaussienne généralisée.

La section A.8 conclut cette thèse, en résumant les contributions et en proposant plusieurs directions de recherche sur la criminalistique et l'anti-criminalistique d'image.

A.2 Préliminaires

A.2.1 Classification de la criminalistique et l'anti-criminalistique d'image

Dans la littérature, il existe différentes taxonomies en criminalistique et en anti-criminalistique d'image [Far09a, RTD11, Piv13, SWL13, BK13]. Pour la criminalistique d'image, celle proposée par Farid [Far09a] est probablement la plus reconnue. La criminalistique d'image fait l'hypothèse que le processus de création de fausses images perturbe certains propriétés intrinsèques de scène/d'image (par exemple, les propriétés statistiques, physiques ou géométriques). Dans ce contexte, Farid [Far09a] propose de diviser les différentes stratégie de criminalistique

d'image suivant les cinq catégories suivantes :

- La criminalistique d'image *à base de pixels* analyse les anomalies causées par la falsification au niveau du pixel. Les moyens de manipulation d'image fréquemment utilisés sont, par exemple, le copier-coller, l'épissage, le rééchantillonnage, le filtrage médian, *etc.* Afin de détecter chaque type d'opération sur l'image, de nombreuses méthodes criminalistiques ont déjà été proposées.
- La criminalistique d'image *basée sur le format* détecte le changement statistique introduit par une méthode de compression particulière. Les algorithmes populaires de compression d'image incluent, par exemple, le JPEG basé sur la transformée DCT, le SPIHT et le JPEG2000 basés sur la transformée en ondelettes, *etc.*
- La criminalistique d'image *basée sur la caméra* étudie les traces laissées *par* le dispositif de capture de l'image [Far06]. Les méthodes criminalistiques dans cette catégorie sont basées, par exemple, sur l'analyse de l'aberration chromatique, du CFA, du PRNU, *etc.*
- La criminalistique d'image *basée sur la physique* examine les anomalies de l'interaction entre les différents objets d'une scène, la lumière et la caméra dans le monde physique en 3 dimensions. Par exemple, les incohérences de directions de la lumière estimées à partir de différents objets physiques peuvent être utilisées comme critères aux fins de criminalistique.
- La criminalistique d'image *basée sur la géométrie* mesure les positions des objets physiques par rapport à la caméra. Par exemple, la manipulation d'image peut être détectée s'il existe des incohérences au point focal de l'image[24].

Pour l'anti-criminalistique d'image, Böhme et Kirchner [BK13] ont divisé les techniques de l'anti-criminalistiques d'image en deux grandes catégories, suivant les trois dimensions suivantes :

- *Robustesse versus sécurité.* En toute généralité, les faussaires peuvent exploiter les faiblesses de robustesse ou de sécurité des méthodes criminalistiques. La *robustesse* en criminalistique désigne la confiance que l'on a dans le diagnostic d'un traitement effectué sur l'image, et la *sécurité* désigne la capacité à détecter comme telles des images falsifiées, même si elles ont fait l'objet d'un traitement anti-criminalistique spécifique. En d'autres termes, et en réfléchissant au pire des cas comme nous y oblige le principe de Kerckhoffs en sécurité, la sécurité indique la capacité à résister à des attaques anti-criminalistiques.
- *Attaques de post-traitement versus traitement intégrés.* Les attaques dites de *post-traitement* sont une étape de traitement supplémentaire et spécifique dans le processus de falsification, afin qu'il ne reste plus de traces dans l'image qui puissent être détectées par la criminalistique. Les attaques *intégrées*, de leur côté, interfèrent directement avec le processus de falsification de l'image. Par définition, ces derniers ne sauraient être évalués pertinemment sous l'angle de la robustesse.

[24] Le point focal est la projection du centre de la caméra sur le plan de l'image.

- *Attaques ciblées versus attaques universelles.* Si une méthode anti-criminalistique exploite les faiblesses d'un outil criminalistique spécifique, elle est dite *ciblée* (ou *ad-hoc*). Une méthode *universelle* essaiera quant à elle de créer une image falsifiée dont les propriétés statistiques sont maintenues aussi plausibles que possible, de sorte que l'image falsifiée reste indétectable, même si elle devait être examinée par des outils criminalistiques encore inconnus aujourd'hui.

De la restauration d'image, cette thèse emprunte quelques méthodes d'estimation du MAP (ou une de ses variantes) et intègre certains termes/stratégies anti-criminalistiques. Selon la classification de Farid [Far09a], la criminalistique JPEG est basée sur le format, alors que la criminalistique du filtrage médian est à base de pixels. Selon la classification de Böhme et Kirchner [BK13], les méthodes anti-criminalistiques proposées à la fois pour la compression JPEG et le filtrage médian sont des *post-traitements* de l'image à falsifier, sont *ciblées*, et permettent d'analyser la *sécurité* des méthodes criminalistiques.

A.2.2 Métriques d'évaluation

A.2.2.1 Indétectabilité criminalistique

Comme indiqué dans la section A.1, l'anti-criminalistique d'image a un double objectif : une bonne indétectabilité criminalistique et une haute qualité de l'image anti-criminalistique. La détectabilité (pour le côté criminalistique) est une métrique qui n'a de sens qu'en rapport avec l'indétectabilité (pour l'anti-criminalistique). Ces deux métriques sont souvent exprimées par une courbe ROC, qui peut être générée en comparant les performances d'un détecteur criminalistique avec la vérité terrain, sur un ensemble d'images conséquent. On considère un ensemble de vraies images (des images originales, les échantillons *négatifs*) et de fausses images (des images traitées, les échantillons *positifs*). Pour chaque stratégie de classification du détecteur criminalistique, on peut calculer un *taux de vrais positifs* et un *taux de faux positifs*. Chaque couple de ces deux taux constitue un point de la courbe ROC, dont un exemple est présenté dans la figure A.4.

Pour une évaluation quantitative, nous utilisons la métrique AUC, en mesurant l'aire sous la courbe ROC. La valeur AUC varie dans l'intervalle $[0, 1]$. Lorsque la valeur AUC est proche de 1, cela signifie que la courbe ROC est proche du point de classification parfaite de l'espace ROC, indiquant une bonne détectabilité criminalistique et une mauvaise indétectabilité criminalistique. Lorsque la valeur AUC est proche de 0.5, cela signifie que la courbe ROC est proche de la *ligne de classification aléatoire* (à savoir la diagonale de l'espace ROC représentée par la ligne pointillée en rouge dans la figure A.4), indiquant une bonne indétectabilité criminalistique et une mauvaise détectabilité criminalistique. Lorsque la valeur AUC est proche de 0, cela ne veut pas dire que la détectabilité criminalistique du détecteur est faible, car il suffirait d'inverser les résultats de classification pour tendre vers 1. En toute généralité, l'objectif de l'anti-criminalistique d'image est d'amener la courbe ROC d'un détecteur criminalistique vers la ligne de classification aléatoire. Dans ce cas, la valeur AUC est proche de 0.5.

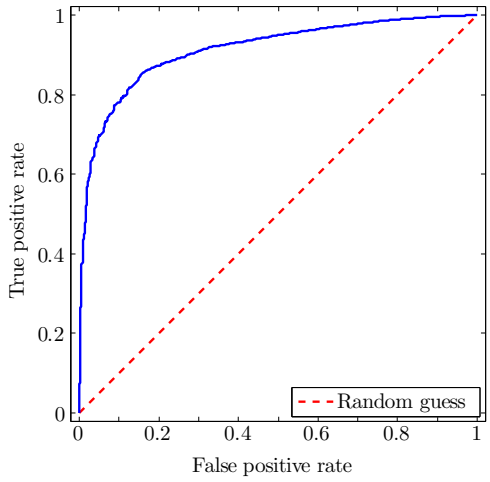


Figure A.4: Un exemple de la courbe ROC.

La validation d'un détecteur criminalistique nécessite deux ensembles d'images, des authentiques et des fausses, en plus du détecteur à valider. Pour notre procédure de validation criminalistique, nous utilisons un ensemble d'images originales (cf. la section A.2.3), sur chacune desquelles on vient appliquer le traitement à détecter afin de produire l'ensemble des positifs. Le test utilisera donc une quantité égale de positifs et de négatifs.

D'autre part, la caractéristique analysée par un détecteur criminalistique pour prendre sa décision peut être soit un scalaire soit un vecteur, selon la méthode considérée. Lorsque la caractéristique est un scalaire, on fera varier la valeur du seuil de décision afin de mesurer le comportement global du détecteur, et lorsque la caractéristique est un vecteur, la littérature retient souvent l'utilisation d'une machine à support vecteur (SVM).

Les détecteurs à base de SVM font l'hypothèse que la méthode anti-criminalistique à détecter est connue, et que l'on est capable de créer une grande quantité de fausses images pour l'entraînement du détecteur. Cela correspond à la configuration la plus accomoddante pour la criminalistique, et au *pire* scénario pour l'anti-criminalistique. Dans la pratique, et dans ce pire des scénarios, nous constatons également qu'il est très difficile de faire passer une image traitée (compressée en JPEG ou filtrée par le filtre médian dans cette thèse) pour une image originale lorsque le détecteur utilise un SVM. Actuellement, aucune méthode anti-criminalistique d'image ne peut tromper les détecteurs puissants à base de SVM.

Inspiré par le scénario de création d'une image composite (illustré dans la figure A.3) et la stéganographie, nous utilisons la configuration expérimentale suivante pour entraîner/tester les détecteurs à base de SVM. Comme illustré dans la figure A.5, pour chaque image originale, la partie centrale de l'image est remplacée par son image traitée correspondante (compressée en JPEG ou filtrée par le filtre médian, et avec ou sans l'application d'un traitement anti-criminalistique) avec un taux de remplacement prenant les valeurs $\{0.05, 0.1, 0.15, 0.2, 0.3, 0.4, 0.5, 0.6, 0.8, 1\}$ pour créer les images traitées. Pour chaque taux de remplacement, les fausses images composites et les images originales sont ensuite mélangées ensemble afin de entraîner/tester les détecteurs à base du SVM. Pour le côté anti-criminalistique,

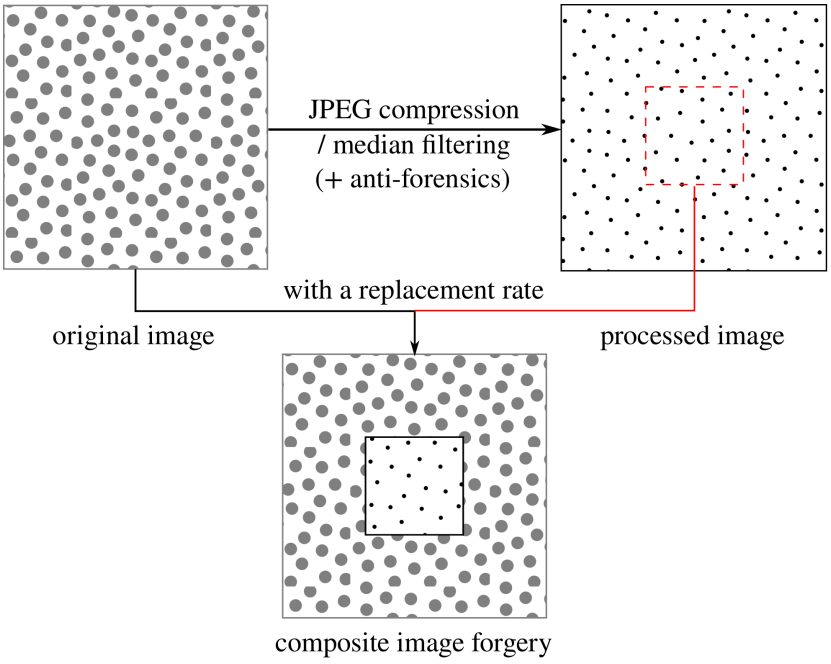


Figure A.5: Illustration de la création d'une fausse image composite afin d'entraîner/tester un détecteur à base de SVM.

cette thèse considère essentiellement les résultats obtenus avec un taux de remplacement relativement faible. Nous restons réservés sur l'indétectabilité criminalistique d'un traitement appliqué sur la totalité de la surface de l'image originale. Cependant, on peut déjà trouver de nombreuses applications dans de nombreux scénarios anti-criminalistiques. Par exemple, une image falsifiée peut être créée par le remplacement de la tête d'une personne dans la photo, sans être détectée.

A.2.2.2 Qualité d'image

Les méthodes d'évaluation de la qualité d'image peuvent être groupées en deux catégories : les métriques *sans référence* et celles *avec référence*, selon que l'image originale sans déformation est connue ou non [WB06]. Dans les scénarios du monde réel, l'image originale est habituellement inaccessible. Cependant, nous disposons souvent la vérité terrain dans la recherche scientifique. Dans ce cas, l'évaluation de la qualité avec référence est possible. Dans cette thèse, nous ne considérons que les images en niveau de gris 8 bits. Nous utilisons deux métriques très connues avec référence pour évaluer la qualité d'image : le PSNR et le SSIM [Wan+04]. L'image originale est utilisée pour calculer les valeurs PSNR et SSIM, afin d'évaluer quantitativement la qualité de l'image anti-criminalistique. Plus ces deux valeurs sont élevées, plus la qualité de l'image traitée est bonne.

A.2.2.3 Comparaison d'histogrammes

Souvent, afin de créer l'image anti-criminalistique d'une image compressée en JPEG ou filtrée par le filtre médian, nous sommes amenés à traiter des histogrammes : l'histogramme des coefficients DCT pour la compression JPEG, et l'histogramme des différences de valeurs de pixels pour le filtre médian. Afin d'évaluer la restauration de l'histogramme, nous adoptons la divergence de Kullback-Leibler [KL51] pour mesurer la ressemblance entre l'histogramme de l'image anti-criminalistique et celui de l'image originale. L'objectif du faussaire est de diminuer cette valeur de divergence vers 0.

A.2.3 Ensembles d'images naturelles

Dans la littérature de l'anti-criminalistique de la compression JPEG [Sta+10a, Sta+10b, VTT11, SS11], l'ensemble UCID [SS04] est le plus utilisé. Nous choisissons donc d'utiliser cet ensemble d'images. Il contient au total 1338 images de taille 384×512 et au format TIFF qui n'ont jamais été compressées. Puis ces images en mode RGB sont transformées en niveau de gris à l'aide de la fonction Matlab `rgb2gray`.

Pour différents usages lors du test de l'anti-criminalistique de la compression JPEG, les ensembles suivants sont créés à partir de l'ensemble UCID :

- UCIDLearn, les 338 dernières images de l'ensemble UCID, pour apprendre un modèle d'image et les matrices de covariance du bruit de compression JPEG dans la domaine spatial.
- UCIDTest, les 1000 premières images de l'ensemble UCID, pour le test anti-criminalistique en utilisant les détecteurs à base de caractéristique scalaire, et aussi pour l'évaluation de la qualité de l'image traitée et de la restauration de l'histogramme DCT.
  - UCIDTR, 500 images choisies de façon aléatoire parmi les images UCIDTest, pour tester les détecteurs à base de SVM.
  - UCIDTE, les 500 autres images dans l'ensemble UCIDTest qui ne sont pas choisies pour l'ensemble UCIDTR, pour entraîner les détecteurs à base de SVM.
  - UCIDTest92, 92 images choisies de façon aléatoire dans l'ensemble UCIDTest, pour régler les paramètres de la méthode proposée ou pour analyser rapidement la performance de la méthode proposée.
  - UCIDTest100, 100 images choisies de façon aléatoire de l'ensemble UCIDTest, pour l'application anti-criminalistique de la double compression JPEG.

Pour le test criminalistique du filtrage médian, les ensembles d'images sont créés à partir de 545 images[25] de haute résolution et au format TIFF qui n'ont jamais été compressées, ni

[25] Ces images peuvent être téléchargées depuis : `ftp://firewall.teleco.uvigo.es:27244/DS_01_UTFI.zip` et `ftp://lesc.dinfo.unifi.it/pub/Public/JPEGloc/dataset/`.

échantillonnées, ni filtrées. Trois images de taille 512×512 sont extraites depuis le centre de l'image de haute résolution, puis elles sont transformées en niveaux de gris à l'aide de la fonction Matlab `rgb2gray`. Les images ainsi extraites qui n'ont pas de valeurs valides pour les algorithmes criminalistiques du filtrage médian [KF10, Cao+10, Yua11] sont exclues. *In fine*, nous avons au total 1607 images qui sont ensuite divisées en trois ensembles : le MFTR, le MFTE et le MFPE. En résumé, les quatre ensembles d'images suivants sont utilisés dans les tests criminalistiques dans notre travail sur le filtrage médian :

- MFTR, 500 images pour entraîner les détecteurs criminalistiques à base de SVM.
- MFTE, 1000 images pour le test criminalistique en utilisant les détecteurs basés sur la criminalistique scalaire et ceux à base de SVM, et aussi pour l'évaluation de la qualité de l'image traitée et de la restauration de l'histogramme des différences de valeurs de pixels.
- MFPE, 107 images pour l'estimation des paramètres.
- MFTE100, 100 images choisies de façon aléatoire dans l'ensemble MFTE, pour régler les paramètres de la méthode proposée.

A.2.4 Algorithmes pertinents d'optimisation

Pour toutes les méthodes anti-criminalistiques proposées dans cette thèse, plusieurs méthodes d'optimisation numérique sont utilisées de manière à estimer l'image traitée " la meilleure " dans un certain sens. Plus précisément, nous utiliserons la méthode de sous-gradient [BXM03], l'algorithme des Hongrois [Kuh55], et le " Half Quadratic Splitting " [GY95] et la méthode dite *split Bregman* [GO09].

La méthode de sous-gradient, itérative, résout un problème de minimisation convexe, selon un sous-gradient de la solution actuelle [BXM03]. Une extension importante de la méthode de sous-gradient est la méthode de sous-gradient projeté, qui permet de résoudre des problèmes d'optimisation sous contraintes. Nous utiliserons la méthode de sous-gradient dans les sections A.4, A.5 et A.6. L'algorithme des Hongrois [Kuh55] est une méthode d'optimisation discrète bien connue pour résoudre le *problème d'affectation* en temps polynomial, cubique. Cette méthode sera utilisée pour résoudre un problème de mise en correspondance des coefficients DCT pour lisser perceptuellement l'histogramme DCT dans la section A.5. De nombreux problèmes avec régularisation en traitement d'image sont difficiles car souvent non-convexes. Le " Half Quadratic Splitting " et la méthode *split Bregman* ont donc été proposés pour faciliter la résolution de ce type de problème d'optimisation complexe, en divisant le problème initial en deux sous-problèmes plus simples [GY95, GO09]. Ces deux techniques d'optimisation seront utilisées dans la sections A.6 et A.7.

A.3 État de l'art en (anti-)criminalistique de compression JPEG et de filtrage médian

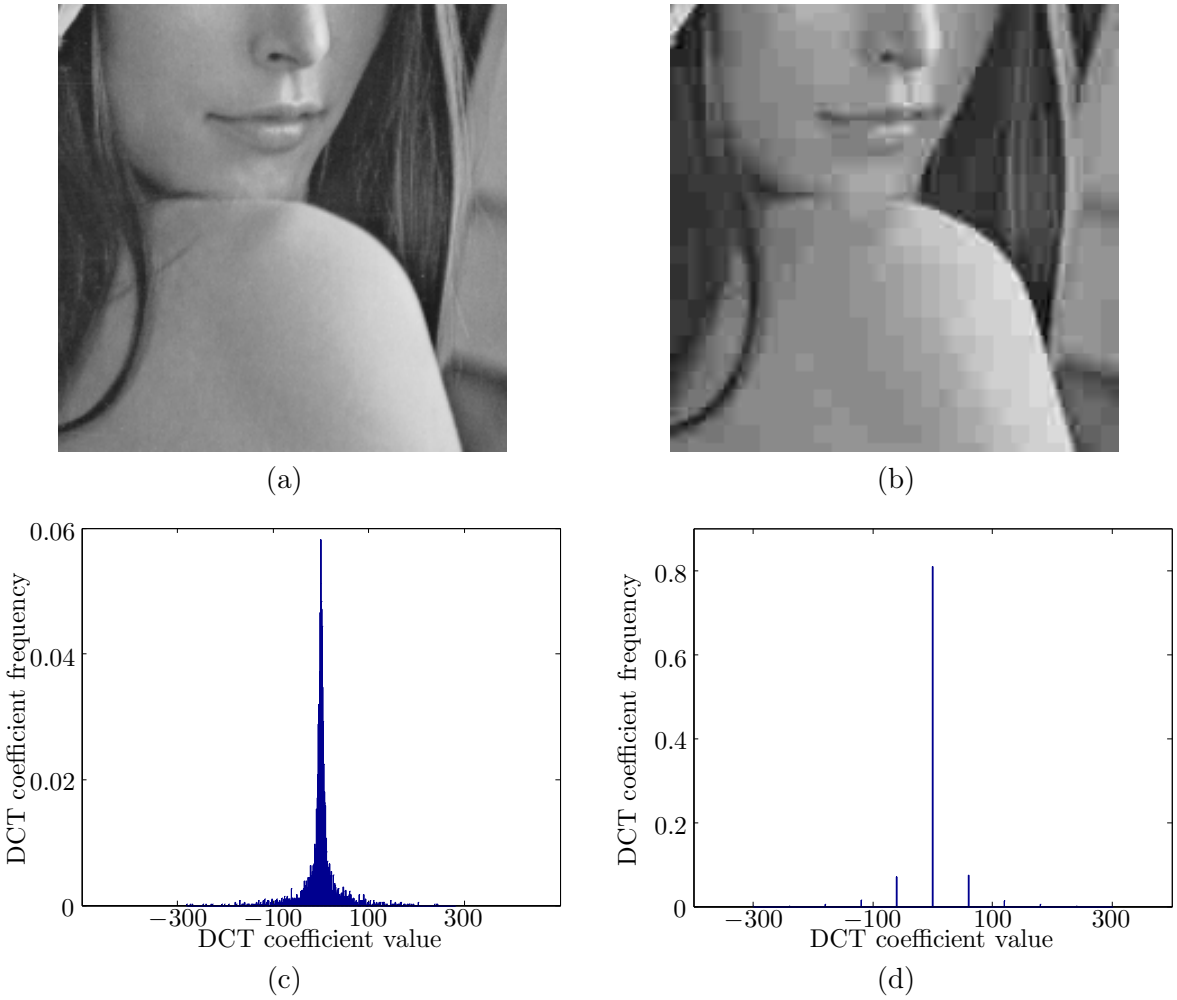


Figure A.6: Exemples de les artefacts de la compression JPEG. Ici, l'image JPEG est compressée de l'image " Lena " avec le facteur de qualité JPEG 10.

A.3.1 (Anti-)Criminalistique de compression JPEG

Après une compression JPEG, deux types bien connus d'artefacts apparaissent, trahissant l'histoire de compression JPEG d'une image. Ce sont, respectivement, les *effets de bloc* dans le *domaine spatial*, et les *artefacts de quantification* dans le *domaine DCT* [FD03]. Un exemple d'effet de bloc peut être trouvé dans la figure A.6-(b), en comparant à -(a). Ceci est principalement à cause de ce que la compression JPEG traite individuellement chaque bloc de taille 8×8. La présence d'effet de bloc est également liée à la base de la DCT [RS05]. Un exemple des artefacts de quantification peut être trouvé dans la figure A.6-(d), en comparant

à -(c). Ceci est à cause de ce que la quantification de compression JPEG est conduite dans le domaine DCT. Afin de créer l'image anti-criminalistique d'une image compressée en JPEG, il est important d'éliminer ces deux types d'artefacts JPEG.

Table A.1: Détecteurs criminalistiques de compression JPEG.

K_F^Q	Détecteur criminalistique basé sur l'estimation de la matrice de quantification, proposé par Fan et De Queiroz [FD03] ;
K_F	Détecteur criminalistique basé sur l'analyse des effets de bloc, proposé par Fan et De Queiroz [FD03] ;
K_{Luo}	Détecteur criminalistique JPEG proposé par Luo *et al.* [LHQ10] ;
K_{Luo}^Q	Détecteur criminalistique basé sur l'estimation du pas de quantification, proposé par Luo *et al.* [LHQ10] ;
K_V	Détecteur criminalistique basé sur la TV, proposé par Valenzise *et al.* [Val+11, VTT13] ;
K_L	Détecteur criminalistique basé sur le calibrage, proposé par Lai et Böhme [LB11] ;
K_U^1, K_U^2	Deux détecteurs proposés par nous et basés sur l'analyse des effets de bloc ;
K_{Li}^{S100}	Détecteur criminalistique basé sur une caractéristique en 100 dimensions qui exploite la corrélation *inter*- ou *intra*-bloc [CS08], proposé par Li *et al.* [LLH12] ;
K_P^{S686}	Détecteur criminalistique basé sur la caractéristique SPAM en 686 dimensions, proposé par Pevný *et al.* [PBF10].

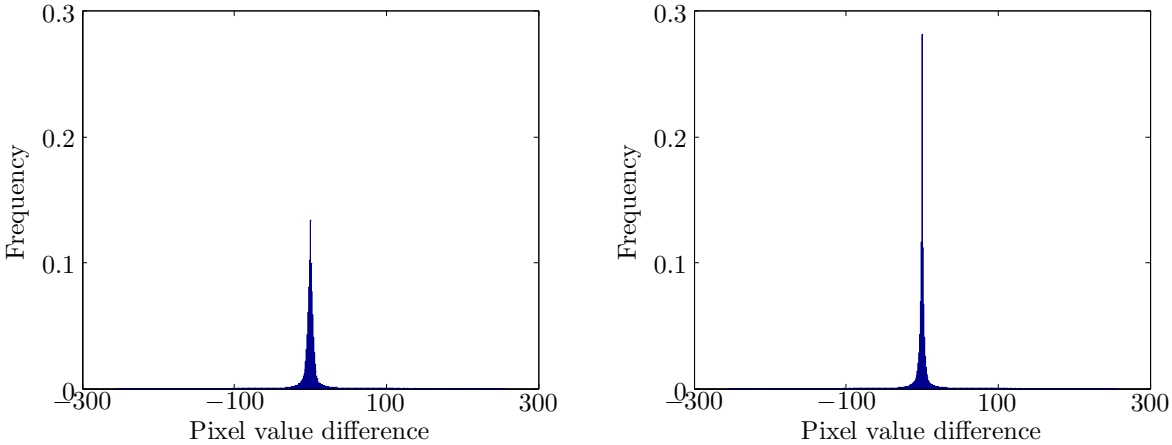
Table A.2: Notations pour l'image originale, compressée JPEG, et anti-criminalistique dans l'état de l'art.

$\mathcal{I}$	L'image originale, qui n'a jamais été compressée ;
$\mathcal{J}$	L'image JPEG, composée à partir de l'image originale $\mathcal{I}$;
$\mathcal{F}_{S_q}^J$	L'image anti-criminalistique, créée à partir de l'image JPEG $\mathcal{J}$, par un procédé de tramage, proposé par Stamm *et al.* [Sta+10a, SL11] ;
$\mathcal{F}_{S_qS_b}^J$	L'image anti-criminalistique, créée à partir de l'image $\mathcal{F}_{S_q}^J$, par un procédé de déblocage basé sur le filtrage médian, proposé par Stamm *et al.* [Sta+10b, SL11] ;
$\mathcal{F}_V^J$	L'image anti-criminalistique, créée à partir de l'image JPEG $\mathcal{J}$, par un procédé de tramage perceptif, proposé par Valenzise *et al.* [VTT11] ;
$\mathcal{F}_{Su}^J$	L'image anti-criminalistique, créée à partir de l'image JPEG $\mathcal{J}$, par une attaque de SAZ, proposé par Sutthiwan et Shi [SS11].

Les détecteurs de compression JPEG utilisés dans cette thèse sont listés dans la table A.1. Parmi tous les détecteurs de compression JPEG en considération, nous en introduisons un nouveau : K_U^p (avec les paramètres $p = 1$, et $p = 2$) qui se fonde sur l'analyse des effets

de bloc. Les autres sont les détecteurs classiques dans la littérature de la criminalistique en compression JPEG. Dans la table A.1, les huit premiers détecteurs sont basés sur des caractéristiques scalaires, alors que les deux derniers utilisent un SVM. Les méthodes anti-criminalistiques en JPEG de l'état de l'art sont listées dans la table A.2. Elles nous seront utiles pour comparer les performances des méthodes que nous proposons.

A.3.2 (Anti-)Criminalistique du filtrage médian



(a) Image originale (b) MFI

Figure A.7: Exemples de l'histogramme de différence de valeurs de pixels au premier ordre de l'image originale et celui de l'image MF, respectivement.

Pour des raisons de concision, l'abréviation " MFI " désigne dans cette thèse l'image qui a été filtrée par le filtre médian. La fenêtre du filtre médian peut avoir de nombreuses formes et tailles différentes. Cette thèse utilise la fenêtre carrée de taille 3×3, qui est aussi la valeur retenue dans l'état de l'art [FB12, WSL13, DN+13]. Dans un certain voisinage de la MFI, la probabilité que les valeurs de pixels proviennent des mêmes pixels de l'image originale est élevée. Cet effet est connu sous le nom d'artefact dit de " *streaking* " [Bov87]. Les artefacts *streaking* peuvent également se refléter dans la modification de la forme de l'histogramme des différences des valeurs de pixels après filtrage médian. En comparant la figure A.7-(b) avec -(a), on peut voir que le pic autour la classe 0 de différence de valeurs de pixels au premier ordre devient plus haut après que l'image est filtrée par le filtre médian.

Dans cette thèse, les détecteurs de l'état de l'art relatifs au filtrage médian sont listés dans la table A.3. Dans cette table, les quatre premiers détecteurs sont basés sur des caractéristiques scalaires, alors que les cinq derniers utilisent un SVM. Les méthodes anti-criminalistiques de l'état de l'art pour le filtre médian sont listées dans la table A.4. Elles seront utilisées pour la comparaison expérimentale avec les nôtres.

Table A.3: Détecteurs criminalistiques du filtrage médian.

K_K	Détecteur basé sur l'analyse de la différence de valeurs de pixels au premier ordre, proposé par Kirchner et Fridrich [KF10] ;
$\hat{K}_K$	Version améliorée de K_K [KF10] ;
K_C	Détecteur basé sur l'analyse de la différence des valeurs de pixels au premier ordre, proposé par Cao *et al.* [Cao+10] ;
K_Y	Détecteur basé sur une caractéristique fusionnée, proposé par Yuan [Yua11] ;
K_{SPAM}^{S686}	Détecteur criminalistique basé sur la caractéristique SPAM en 686 dimensions, proposé par Pevný *et al.* [PBF10];
K_{MFF}^{S44}	Détecteur criminalistique basé sur la caractéristique MFF en 44 dimensions, proposé par Yuan [Yua11] ;
K_{GLF}^{S56}	Détecteur criminalistique basé sur la caractéristique GLF en 56 dimensions, proposé par Chen *et al.* [Che+12, CNH13] ;
K_{AR}^{S10}	Détecteur criminalistique basé sur la caractéristique MFRAR en 10 dimensions, proposé par Kang *et al.* [Kan+12, Kan+13] ;
K_{LTP}^{S220}	Détecteur criminalistique basé sur la caractéristique MFLTP en 220 dimensions, proposé par Zhang *et al.* [Zha+14].

Table A.4: Notations pour l'image originale, filtrée médian, et anti-criminalistique dans l'état de l'art.

$\mathcal{I}$	L'image originale ;
$\mathcal{M}$	L'image filtrée à partir de l'image originale $\mathcal{I}$ par le filtre médian, aussi appelée MFI dans le texte ;
$\mathcal{F}_W^M$	L'image anti-criminalistique, créée à partir de la MFI $\mathcal{M}$, par un procédé de tramage, proposé par Wu *et al.* [Sta+10b, SL11] ;
$\mathcal{F}_D^M$	L'image anti-criminalistique, créée à partir de la MFI $\mathcal{M}$, par une injection de bruit, proposé par Dang-Nguyen *et al.* [DN+13].

A.4 Anti-criminalistique de compression JPEG basée sur la TV

A.4.1 Introduction et motivation

Afin de créer l'image anti-criminalistique d'une image JPEG, il faut supprimer les deux types d'artefacts présents à la fois dans le domaine DCT et dans le domaine spatial. Dans la pratique, nous trouvons qu'il est extrêmement difficile de mener une attaque en une seule étape pour tromper plusieurs détecteurs criminalistiques qui travaillent dans deux différents domaines, tout en conservant une haute qualité d'image anti-criminalistique. Par conséquent, cette thèse supprime les artefacts JPEG alternativement dans le domaine DCT et dans le

domaine spatial. Dans cette section, nous nous concentrons sur la suppression des effets de bloc d'une image JPEG au niveau anti-criminalistique. Nous laissons la suppression des artefacts de quantification dans le domaine DCT suppression pour la section A.5.

Suivant la piste de recherche décrite dans la section A.1.3.3 et visant à utiliser les concepts de la restauration, cette section présente notre méthode pour supprimer les effets de bloc d'une image JPEG (basée sur la régularisation de la TV). L'idée de régulariser de la TV (variation totale) a été proposé par Rudin *et al.* [ROF92] en 1992. Depuis lors, elle a été largement utilisée dans la restauration d'image, pour de nombreuses applications comme la réduction du bruit, la déconvolution, l'*inpainting*, *etc.* Les effets de bloc conduisent à une valeur plutôt haute de la TV. Par conséquent, le déblocage JPEG peut être effectué en minimisant une fonction d'énergie basée sur la TV [ADF05]. Traditionnellement, son objectif principal est d'améliorer la qualité de l'image JPEG, mais cela ne suffit pas pour obtenir une bonne indétectabilité criminalistique. Aux fins de criminalistique, nous ajoutons à la fonction de minimisation un autre terme de mesure du blocage JPEG basé sur la TV. En résolvant un nouveau problème de minimisation basé sur la TV, nous pouvons créer l'image anti-criminalistique avec une bonne indétectabilité criminalistique en sacrifiant un tout petit peu de la qualité d'image.

A.4.2 Déblocage JPEG en minimisant un problème contraint basé sur la TV

Pour image $\mathbf{U}$ de la taille $H \times W$, nous définissons le terme de TV comme :

$$\mathrm{TV}_b(\mathbf{U}) = \sum_{1 \leq i \leq H, 1 \leq j \leq W} \nu_{i,j}, \tag{A.1}$$

avec la variation à l'emplacement de (i, j) :

$$\nu_{i,j} = \sqrt{(\mathbf{U}_{i-1,j} + \mathbf{U}_{i+1,j} - 2\mathbf{U}_{i,j})^2 + (\mathbf{U}_{i,j-1} + \mathbf{U}_{i,j+1} - 2\mathbf{U}_{i,j})^2}, \tag{A.2}$$

où $\mathbf{U}_{i,j}$ est la valeur du (i, j)-ème pixel de l'image $\mathbf{U}$.

Afin d'éliminer les effets de bloc, nous définissons un second terme qui mesure le blocage JPEG. L'idée est très simple : s'il n'y a pas de compression JPEG, statistiquement la somme de l'énergie de la variation des pixels près des *frontières* du bloc devrait être proche de celle dans le centre du bloc. Nous divisons donc tous les pixels de l'image en deux groupes : $\mathcal{A}$ et $\mathcal{B}$, en fonction de leur position dans le bloc, comme il est illustré dans la figure A.8. Selon cette classification des pixels, le second terme de l'énergie est défini comme :

$$C(\mathbf{U}) = \left| \sum_{\mathbf{U}_{i,j} \in \mathcal{A}} \nu_{i,j} - \sum_{\mathbf{U}_{i,j} \in \mathcal{B}} \nu_{i,j} \right|. \tag{A.3}$$

Le problème contraint final basé sur la TV est composé d'un terme de TV et d'un terme

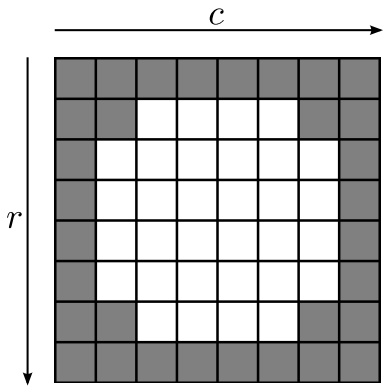


Figure A.8: Classification des pixels en fonction de leur position dans le bloc de taille 8×8.

mesurant le blocage basé sur la TV :

$$\mathbf{U}^* = \arg\min_{\mathbf{U} \in \mathcal{S}} E(\mathbf{U}) = \arg\min_{\mathbf{U} \in \mathcal{S}} \left(\mathrm{TV}_b(\mathbf{U}) + \alpha C(\mathbf{U})\right), \tag{A.4}$$

où $\alpha > 0$ est un paramètre de régularisation, et $\mathcal{S}$ (paramétrée par le paramètre μ) est un espace contraint d'images qui évite une trop grande modification des coefficients DCT.

A.4.3 Décalibrage

Dans la pratique, la méthode proposée pour le déblocage JPEG dans la section A.4.2 est capable de créer des images anti-criminalistiques qui peuvent tromper les huit premiers détecteurs basés sur des caractéristiques scalaires dans la table A.1. Mais la sortie du détecteur K_L [LB11] (basé sur la caractéristique de calibrage) reste difficile à diminuer sans affecter sérieusement la qualité de l'image. Afin de résoudre ce problème, l'image obtenue après le traitement du déblocage est soumise au problème d'optimisation suivant :

$$\mathbf{U}^* = \arg\min_{\mathbf{U}} \sum_{k=1}^{28} \left| \mathrm{var}(\mathbf{D}_k \mathbf{U}) - \mathrm{var}(\mathbf{D}_k \mathbf{U}_{cal}) \right|, \tag{A.5}$$

dont la fonction de coût a une forme similaire à la fonction de calcul de la caractéristique de K_L [LB11].

A.4.4 Quelques résultats expérimentaux

De chaque image originale $\mathcal{I}$, une image JPEG $\mathcal{J}$ est obtenue en compressant $\mathcal{I}$ avec un facteur de qualité JPEG choisi de façon aléatoire parmi $\{50, 51, \cdots, 95\}$. De l'image $\mathcal{J}$, nous créons notre image anti-criminalistique $\mathcal{F}_0^J$ par la procédure du déblocage décrite dans la section A.4.2 et le procédé du décalibrage décrit dans la section A.4.3. La figure A.9 rapporte les courbes ROC obtenues pour différentes images contre différents détecteurs criminalistiques. Les courbes obtenues par notre méthode sont les plus proches de la ligne de classification

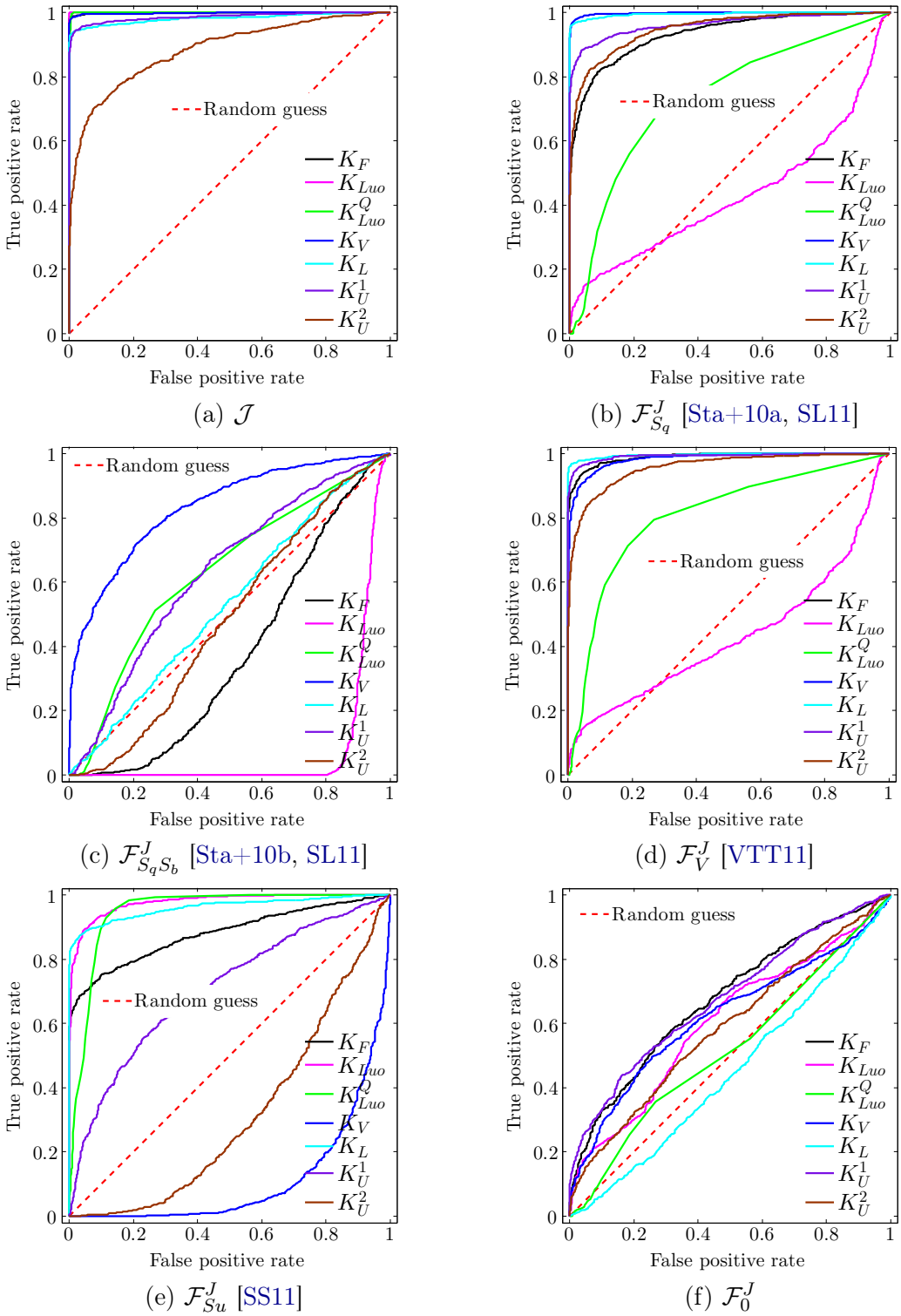


Figure A.9: Les courbes ROC obtenues pour différentes images contre différents détecteurs criminalistiques. Les résultats sont obtenus à partir de l'ensemble UCIDTest.

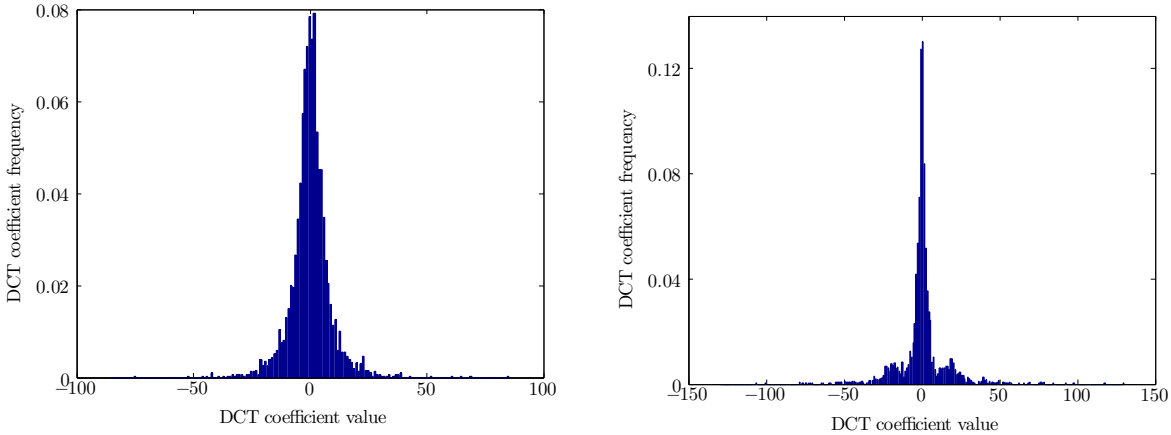
Table A.5: La comparaison de la qualité d'image, où les valeurs PSNR/SSIM sont calculées avec $\mathcal{I}$ comme référence. Les résultats sont obtenus à partir de l'ensemble UCIDTest.

	$\mathcal{J}$	$\mathcal{F}^J_{S_q}$	$\mathcal{F}^J_{S_qS_b}$	$\mathcal{F}^J_V$	$\mathcal{F}^J_{Su}$	$\mathcal{F}^J_0$
PSNR	37.0999	33.4061	30.4591	33.2890	31.6552	35.4814
SSIM	0.9919	0.9756	0.9509	0.9802	0.9719	0.9843

aléatoire, en comparant avec les méthodes anti-criminalistique de l'état de l'art. La comparaison de la qualité d'image est explicitée dans la table A.5. La qualité de $\mathcal{F}^J_0$ est la meilleure parmi toutes les images anti-criminalistiques. De ces résultats, on peut conclure que la méthode proposée surpasse les méthodes anti-criminalistiques de l'état de l'art [Sta+10a, Sta+10b, SL11, VTT11, SS11], au regard tant de la capacité à résister à la détection des méthodes criminalistiques, que de la qualité de l'image anti-criminalistique produite.

A.5 Anti-criminalistique de compression JPEG avec un lissage perceptuel de l'histogramme DCT

A.5.1 Introduction et motivation



(a) $\mathcal{F}^J_{S_qS_b}$ [Sta+10b, Sta+10a, SL11] (b) $\mathcal{F}^J_0$

Figure A.10: Exemples d'histogrammes DCT de la sous-bande $(4, 4)$, de $\mathcal{F}^J_{S_qS_b}$ [Sta+10a, Sta+10b, SL11] et de $\mathcal{F}^J_0$, respectivement.

La figure A.10 compare un exemple de l'histogramme DCT de l'image anti-criminalistique $\mathcal{F}^J_{S_qS_b}$ [Sta+10b, Sta+10a, SL11] et celui de notre image anti-criminalistique $\mathcal{F}^J_0$ créée par la méthode décrite dans la section A.4. On peut voir que les artefacts de quantification dans le domaine DCT persistent encore, dans une certaine mesure, dans $\mathcal{F}^J_0$, généralement dans les sous-bandes de fréquence moyenne de l'image. Cette faiblesse peut potentiellement être utilisée par les détecteurs criminalistiques avancés. Afin de résoudre ce problème, dans notre seconde

méthode proposée pour l'anti-criminalistique d'image de compression JPEG, une procédure de lissage perceptuel de l'histogramme DCT est mis au point. Ce lissage d'histogramme est estiné à éliminer les artefacts de quantification restants dans l'image traitée par la méthode de déblocage (mais avec un paramètre différent par rapport à la génération de $\mathcal{F}_0^J$).

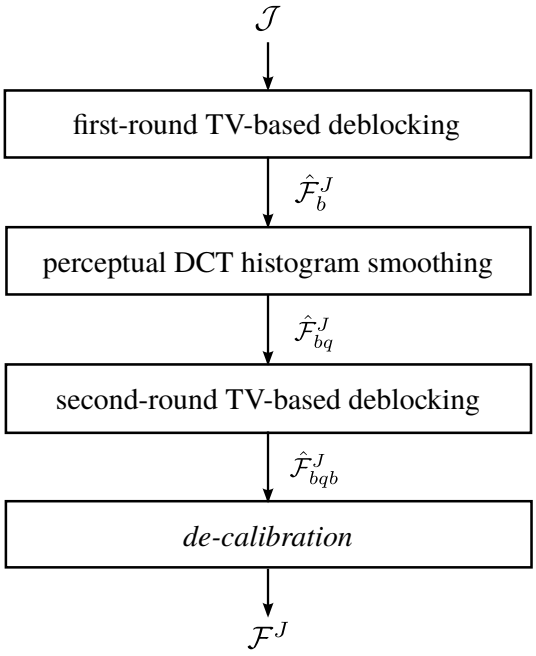


Figure A.11: Le processus proposé pour créer l'image anti-criminalistique $\mathcal{F}^J$.

À titre d'amélioration de la méthode dans la section A.4, nous proposons une méthode en quatre étapes pour créer l'image anti-criminalistique de l'image JPEG, comme illustré dans la figure A.11 et résumées dans ce qui suit :

- La première étape est le déblocage JPEG en minimisant un problème contraint basé sur la TV dans le domain spatial. Cette méthode est identique à celle décrite dans la section A.4.2, mais avec différents paramètres. Après cette étape, l'image générée à partir de l'image JPEG $\mathcal{J}$ est dénotée par $\hat{\mathcal{F}}_b^J$.
- Les artefacts de quantification dans l'image $\hat{\mathcal{F}}_b^J$ ne sont plus aussi évidents que ceux présents dans l'image JPEG $\mathcal{J}$. Sous l'hypothèse que les informations DCT récupérées en partie sont fiables, la prochaine étape sera naturellement de remplir les vides restants dans l'histogramme DCT. Cette tâche est remplie par la procédure de lissage perceptuel de l'histogramme DCT, qui sera décrite dans la section A.5.2. $\hat{\mathcal{F}}_{bq}^J$ désigne l'image intermédiaire créée à partir de $\hat{\mathcal{F}}_b^J$ pour cette étape.
- Dans le domaine spatial, afin de supprimer le bruit anormal introduit par la seconde étape, on effectue un second déblocage JPEG. L'image intermédiaire résultante est notée $\hat{\mathcal{F}}_{bqb}^J$.
- Enfin, l'image $\hat{\mathcal{F}}_{bqb}^J$ est traitée par le décalibrage, qui est identique à celui décrit dans la section A.4.3, afin de générer notre image anti-criminalistique finale $\mathcal{F}^J$.

Les méthodes dans les première, troisième, et dernière étapes dans la figure A.11 sont décrites dans les sections A.4.2 et A.4.3, mais avec différents paramètres. Pour le premier déblocage JPEG, le paramétrage suivant est utilisé : $\alpha = 1.5$, $\mu = 0.5$ et $t_k = 1/k$ ($k = 1, 2, \cdots, 50$). Pour le deuxième déblocage JPEG, le paramétrage suivant est utilisé : $\alpha = 0.9$, $\mu = 1.5$ et $t_k = 1/(k+1)$ ($k = 1, 2, \cdots, 30$). La section A.5.2 présentera la méthode permettant de lisser perceptuellement l'histogramme DCT de l'image $\hat{\mathcal{F}}_b^J$ dans la deuxième étape de la figure A.11.

A.5.2 Lissage perceptuel de l'histogramme DCT

Afin de lisser l'histogramme DCT, il est important de trouver un bon modèle pour les coefficients DCT. Dans la littérature, la loi de Laplace est un choix très populaire. Son kurtosis vaut 6. Nous avons calculé le kurtosis pour chaque sous-bande AC de chaque image de l'ensemble UCID [SS04]. La valeur moyenne est 19.99, qui est beaucoup plus grand que 6, et qui indique que la distribution réelle des coefficients DCT a généralement un pic beaucoup plus élevé que la loi de Laplace. D'autre part, 93.68% des valeurs de kurtosis calculés song plus grandes que 6. Ces chiffres montrent que la loi de Laplace n'est peut-être pas très appropriée pour modéliser les coefficients DCT (un exemple peut être trouvé dans la figure A.12-(a)). En outre, Robertson et Stevenson [RS05] ont souligné que la loi de Laplace fonctionne effectivement bien pour la classe de quantification 0, mais la loi uniforme est un modèle plus approprié pour les autres classes de quantification.

Sur la figure A.12-(c), on peut voir que les informations dans le domaine DCT sont récupérées en partie par le premier déblocage JPEG. Mais les artefacts de quantification persistent encore, dans un certain mesure, dans $\hat{\mathcal{F}}_b^J$. Avec ces informations DCT récupérées, et d'après l'analyse ci-dessus, nous proposons un modèle adaptatif local laplacien afin de modéliser les coefficients DCT. L'idée principale est de rechercher le paramètre λ_b de la loi de Laplace pour chaque classe de quantification b, dès la classe 0, en résolvant le problème contraint des moindres carrés pondérés suivant :

$$\lambda_b = \underset{\lambda_b^- \leq \lambda \leq \lambda_b^+}{\arg\min} \sum_{k=B_{r,c}^- \mathbf{Q}_{r,c} - \lfloor \frac{\mathbf{Q}_{r,c}}{2} \rfloor}^{B_{r,c}^+ \mathbf{Q}_{r,c} + \lfloor \frac{\mathbf{Q}_{r,c}}{2} \rfloor} w_k \times \left(H_{r,c}^{\mathbf{U}}(k) - P(Y = k) \right)^2, \tag{A.6}$$

où $\mathbf{Q}_{r,c}$ est le pas de quantification pour la sous-bande AC actuelle, $B_{r,c}^-$ (resp. $B_{r,c}^+$) est la classe de quantification non vide avec la valeur la plus petite (resp. grande) au centre de la classe, w_k est le poids, $H_{r,c}^{\mathbf{U}}$ est l'histogramme DCT normalisé, et P est la loi discrète de Laplace. On peut voir ce problème comme celui d'un ajustement local des coefficients DCT pour chaque classe de quantification b. Si un valeur valide est trouvée λ_b, la loi de Laplace sera utilisée par la classe b. Sinon, c'est la loi uniforme qui sera utilisée pour la classe actuelle b et les suivantes. Pour la sous-bande DC, la loi de Laplace est utilisée pour toutes les classes de quantification.

En construisant le modèle adaptatif de Laplace local comme décrit ci-dessus, on peut obtenir la distribution cible pour les coefficients DCT de chaque sous-bande de l'image. Un

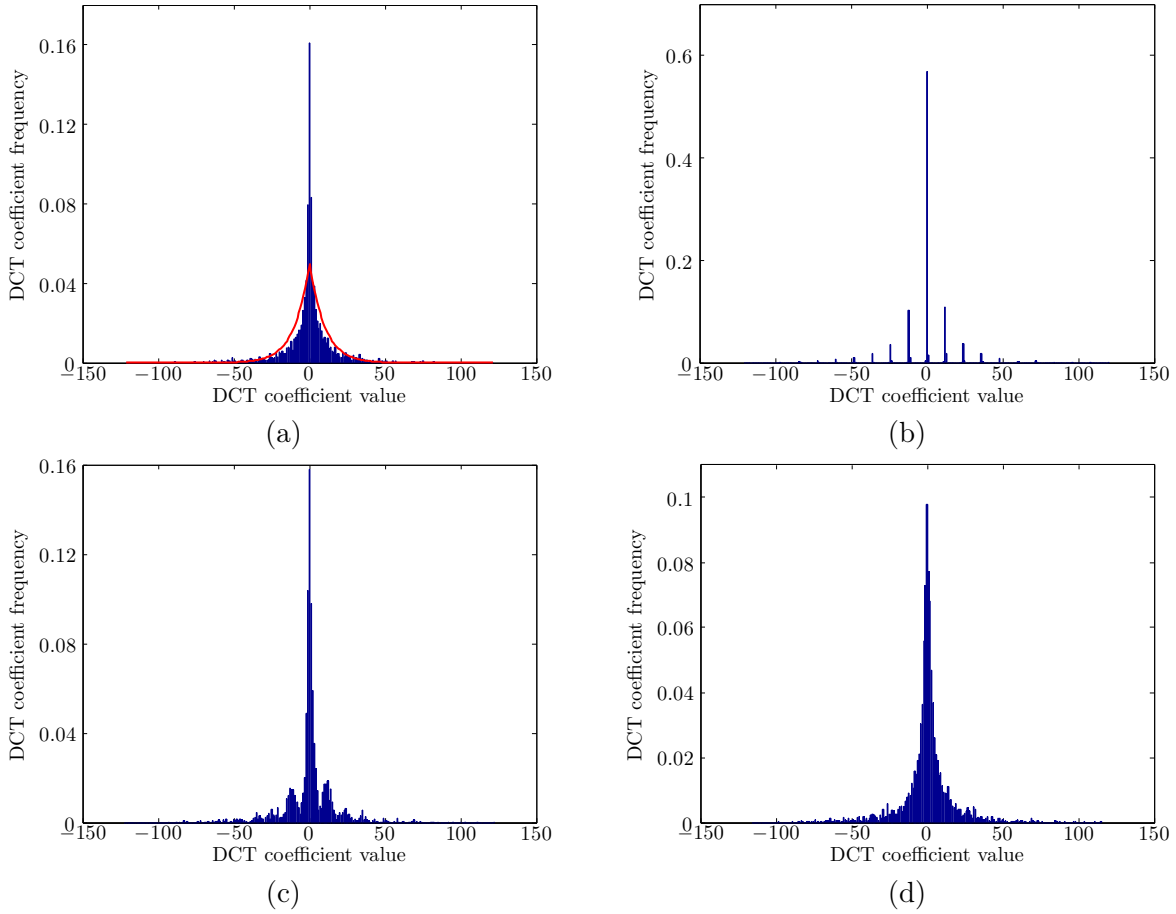


Figure A.12: (a) est un histogramme DCT de la sous-bande $(2, 2)$ d'une image originale de l'ensemble UCID [SS04], et la courbe en rouge est le résultat d'ajustement en utilisant la loi discrète de Laplace. Cette image est ensuite compressée avec un certain facteur de qualité JPEG 50 afin de générer l'image JPEG, elle est ensuite traitée par la méthode proposée. (b), (c), et (d) sont les histogrammes correspondants à l'histogramme (a) de l'image JPEG $\mathcal{J}$, de l'image $\hat{\mathcal{F}}_b^J$ obtenue à partir de $\mathcal{J}$ par le premier déblocage JPEG, et d'image obtenue à partir de $\hat{\mathcal{F}}_b^J$ par l'injection du signal de tramage local adaptatif, respectivement.

exemple est montré dans la figure A.12-(d). L'histogramme DCT apparaît bien lissé. À présent, nous décrivons comment modifier les coefficients DCT afin que leur distribution approche la distribution cible (cf. figure A.12-(d)) et tout en minimsant la distorsion infligée à l'image par ce traitement précis. Pour ce problème, nous nous proposons un problème d'affectation et souhaitons trouver une bijection $f : O^b \to T^b$ de sorte que la fonction de coût :

$$\sum_{o \in O^b} W(o, f(o)), \tag{A.7}$$

soit minimisée. Ici, pour chaque classe de quantification b, O^b est l'ensemble coefficients DCT originaux, et T^b est celui des valeurs cible des coefficient DCT. La fonction de poids $W : O^b \times T^b \to \mathbb{R}$ est définie comme la perte de valeur SSIM à cause de la modification des coefficients DCT. Après la résolution d'un problème d'affectation simplifiée pour chaque classe

de quantification, nous pouvons lisser l'histogramme DCT perceptuellement avec un minimum de distorsion introduite dans le domaine spatial, puis enfin créer l'image $\hat{\mathcal{F}}^J_{bq}$.

A.5.3 Quelques résultats expérimentaux

A.5.3.1 Comparaison de différentes méthodes de tramage

Nous comparons notre méthode de tramage basée sur le modèle adaptatif local de Laplace avec celle basée sur la loi de Laplace proposée par Stamm *et al.* [Sta+10a, SL11]. La table A.6 rapporte la différence entre la divergence K-L obtenue par $\mathcal{F}^J_{S_q}$ [Sta+10a, SL11] et celle obtenue par $\hat{\mathcal{F}}^J_{bq}$. Toutes les valeurs de différence listées dans la table A.6 indiquent que la méthode proposée surpasse constamment la méthode de Stamm *et al.* [Sta+10a, SL11] pour toutes les sous-bandes DCT

Table A.6: La différence entre la divergence K-L obtenue par $\mathcal{F}^J_{S_q}$ [Sta+10a, SL11] et celle obtenue par $\hat{\mathcal{F}}^J_{bq}$ pour les 64 sous-bandes DCT. La valeur moyenne de la différence est 0.0552 avec une déviation standard 0.0249. Pour une comparaison équitable avec $\mathcal{F}^J_{S_q}$, les sous-bandes dont les coefficients DCT sont tous quantifiés à 0 dans l'image JPEG $\mathcal{J}$, ne sont pas comptés. Les résultat sont obtenus à partir de l'ensemble UCIDTest.

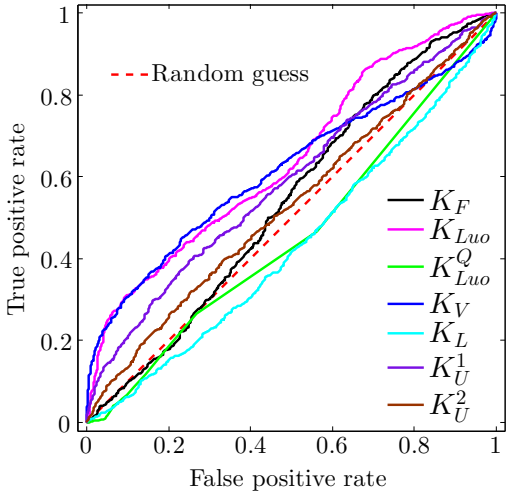
r \ c	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
1	0.0001	0.0065	0.0118	0.0278	0.0493	0.0634	0.0663	0.0656
2	0.0042	0.0166	0.0229	0.0363	0.0447	0.0565	0.0504	0.0369
3	0.0161	0.0208	0.0291	0.0442	0.0573	0.0665	0.0634	0.0497
4	0.0200	0.0317	0.0409	0.0470	0.0658	0.0802	0.0553	0.0446
5	0.0357	0.0395	0.0522	0.0678	0.0764	0.0930	0.0927	0.0856
6	0.0441	0.0383	0.0642	0.0610	0.0726	0.0769	0.0806	0.0957
7	0.0538	0.0442	0.0678	0.0595	0.0879	0.0809	0.0948	0.0975
8	0.0619	0.0545	0.0697	0.0528	0.0927	0.0880	0.0854	0.0722

A.5.3.2 Contrer les détecteurs criminalistiques de compression JPEG

Partant d'une image JPEG $\mathcal{J}$, nous la traitons suivant les quatre étapes de la méthode anti-criminalistique proposée (cf. figure A.11) afin de créer l'image anti-criminalistique $\mathcal{F}^J$. La figure A.13 montre les courbes ROC obtenues par $\mathcal{F}^J$ contre des détecteurs scalaires basés, qui sont toutes proches de la ligne de classificateur aléatoire. Cela signifie que l'image anti-criminalistique créée par la méthode proposée est capable d'atteindre une bonne indétectabilité criminalistique. Cette information se reflète aussi par les valeurs AUC atteintes par $\mathcal{F}^J$ (cf. table A.7), en comparant aux autres méthodes de l'état de l'art. En outre, $\mathcal{F}^J$ est capable d'atteindre des valeurs AUC plus petites que les autres images anti-criminalistiques de l'état de l'art en variant le taux de remplacement d'image. Selon les résultats de la figure A.14, quand le

Table A.7: De la deuxième colonne à la huitième colonne, sont listées les valeurs AUC obtenues pour différentes images contre des détecteurs scalaires ; la comparaison de la qualité d'image (avec $\mathcal{I}$ comme référence) est rapportée dans les deux dernières colonnes. Les résultats sont obtenus à partir de l'ensemble UCIDTest.

	K_F	K_{Luo}	K_{Luo}^Q	K_V	K_L	K_U^1	K_U^2	PSNR	SSIM
$\mathcal{J}$	0.9991	1.0000	0.9996	0.9976	0.9811	0.9860	0.8840	37.0999	0.9919
$\mathcal{F}_{S_q}^J$	0.9332	0.4328	0.7328	0.9977	0.9946	0.9633	0.9483	33.4061	0.9756
$\mathcal{F}_{S_qS_b}^J$	0.3783	0.0806	0.6288	0.8337	0.5338	0.6309	0.4854	30.4591	0.9509
$\mathcal{F}_V^J$	0.9889	0.4330	0.8066	0.9834	0.9958	0.9916	0.9574	33.2890	0.9802
$\mathcal{F}_{Su}^J$	0.8802	0.9772	0.9475	0.1115	0.9610	0.7052	0.3149	31.6552	0.9719
$\mathcal{F}_0^J$	0.6756	0.6046	0.5194	0.6210	0.4490	0.6772	0.5880	35.4814	0.9843
$\hat{\mathcal{F}}_b^J$	0.7590	0.4830	0.4354	0.8542	0.9588	0.6813	0.5650	36.7405	0.9891
$\hat{\mathcal{F}}_{bq}^J$	0.9170	0.4050	0.5244	0.8435	0.9874	0.8081	0.6531	35.9926	0.9872
$\mathcal{F}^J$	0.5398	0.6425	0.4598	0.6159	0.4344	0.5894	0.5317	35.9855	0.9866

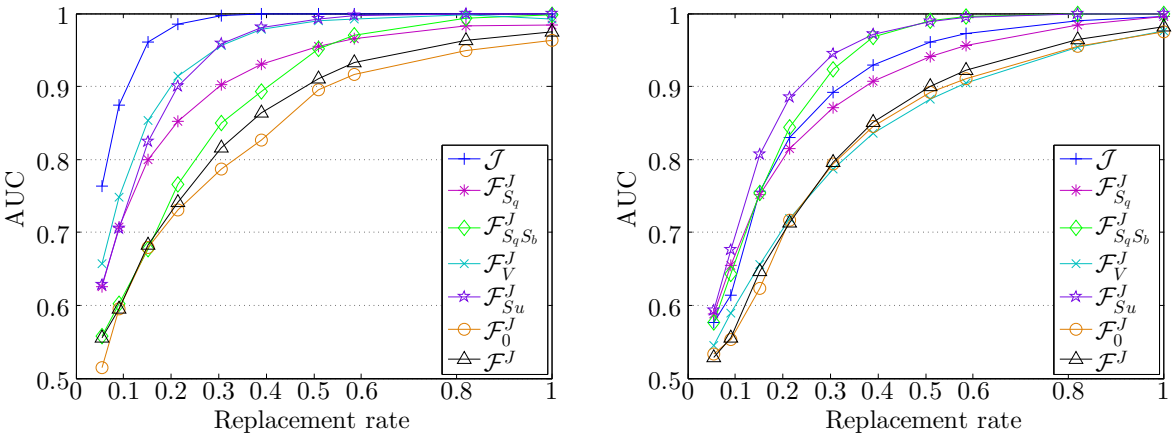


(b) $\mathcal{F}^J$

Figure A.13: Les courbes ROC obtenues par $\mathcal{F}^J$ contre des détecteurs criminalistiques. Les résultat sont obtenus à partir de l'ensemble UCIDTest.

taux de replacement est relativement petit, notre image anti-criminalistique reste indétectable contre les deux détecteurs à base de SVM [LLH12, PBF10]. De plus, $\mathcal{F}^J$ surpasse les autres images anti-criminalistiques au point de vue de la qualité de l'image, en l'évaluant cette qualité tant par le PSNR que par le SSIM.

Par rapport à la méthode proposée dans la section A.4, le lissage perceptuel d'histogramme DCT remplit explicitement les lacunes dans l'histogramme DCT de l'image JPEG. Cela augmente la performance anti-criminalistique de $\mathcal{F}^J$. Enfin, $\mathcal{F}^J$ présente également une meilleure qualité d'image que $\mathcal{F}_0^J$.



(a) K_{Li}^{S100} [CS08, LLH12] (b) K_P^{S686} [PBF10]

Figure A.14: La valeur AUC en fonction du taux de remplacement d'image pour différents types d'images, lorsqu'elles sont testées en utilisant les détecteurs à base de SVM. Les résultats sont obtenus à partir de l'ensemble UCIDTR pour l'entraînement, et à partir de l'ensemble UCIDTE pour les tests.

A.6 Amélioration de qualité et anti-criminalistique de l'image JPEG basée sur un modèle d'image avancé

A.6.1 Introduction et motivation

Les deux méthodes proposées dans les sections A.4 et A.5 utilisent la TV pour le déblocage JPEG. La TV peut être considérée comme un modèle a priori d'image simple. Puisque cette thèse suit la ligne de recherche décrite dans la section A.1.3.3, à savoir concevoir des méthodes anti-criminalistiques en s'appuyant sur la restauration d'image, cette section traite la possibilité d'utiliser un modèle a priori d'image plus avancé que la TV. Dans notre troisième méthode d'anti-criminalistique JPEG, présentée dans cette section, nous essayons tout d'abord d'améliorer la qualité de l'image JPEG $\mathcal{J}$. De cette façon, nous espérons que non seulement la qualité de l'image JPEG sera améliorée, mais aussi que le coût pour créer l'image anti-criminalistique avec une bonne indétectabilité criminalistique sera également réduit par cette étape de restauration d'image.

A.6.2 Amélioration de qualité de l'image JPEG

Dans la littérature de la restauration d'image, afin d'améliorer la qualité de l'image JPEG, il est courant de modéliser le processus de la compression JPEG comme une addition de bruit dans le domaine spatial $\mathbf{n}_q$ à l'image originale $\mathbf{x}$ pour générer l'image JPEG $\mathbf{y}$: $\mathbf{y} = \mathbf{x} + \mathbf{n}_q$. Le bruit de compression $\mathbf{n}_q$ est supposé être une quantité aléatoire, qui peut être modélisé en utilisant une gaussienne multivariée de moyenne nulle. De plus, $\mathbf{n}_q$ et $\mathbf{x}$ sont supposés être indépendants. L'estimateur du MAP est souvent employé afin d'estimer l'image restaurée $\hat{\mathbf{x}}$.

Dans la méthode proposée, le problème d'optimisation basé sur le MAP est :

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{u}} \left\{ \sum_{k=1}^{64} \sum_{\mathbf{P}^i \in \mathcal{S}_k} \frac{1}{2} \left(\mathbf{P}^i(\mathbf{y}-\mathbf{u})\right)^t (\mathbf{C}^k)^{-1} \mathbf{P}^i(\mathbf{y}-\mathbf{u}) - \sum_i \log p(\mathbf{P}^i\mathbf{u}) \right\}, \tag{A.8}$$

dont le premier terme utilise la gaussienne multivariée de moyenne nulle pour modéliser le bruit de compression, et le deuxième terme utilise le modèle de GMM dans le cadre EPLL [ZW11] pour modéliser les *patches* d'image. Ici, $\mathbf{P}^i\mathbf{u}$ est le i-ème *patch* extrait de l'image $\mathbf{u}$.

La table A.8 compare l'image obtenue en résolvant le problème d'optimisation de l'équation (A.8), à celle traitée par une méthode de l'état de l'art [SC07]. On peut voir que la méthode proposée est très compétitive en terme de gain en PSNR. De plus, il est environ dix fois plus rapide de traiter une image par la méthode proposée que par la méthode proposée par [SC07].

Table A.8: Valeurs de PSNR (avec l'image originale comme référence) pour quatre images classiques et trois différentes matrices de quantification fournies par [SC07].

		Image JPEG	Basée sur le FoE [SC07]	Proposée
Lena	Q1	30.71	31.95	**32.06**
	Q2	30.08	31.44	**31.48**
	Q3	27.45	28.83	**28.94**
Peppers	Q1	30.72	32.04	**32.09**
	Q2	30.17	**31.61**	31.59
	Q3	27.66	29.35	**29.40**
Barbara	Q1	25.95	26.65	**26.94**
	Q2	25.60	26.31	**26.56**
	Q3	24.05	24.86	**25.00**
Baboon	Q1	24.32	24.77	**24.84**
	Q2	24.14	24.62	**24.68**
	Q3	22.14	**22.61**	**22.61**

A.6.3 Anti-criminalistique de compression JPEG

Afin de supprimer les artefacts de quantification de l'image JPEG, il convient de lisser l'histogramme DCT. À cette fin, nous proposons la méthode suivante basée sur le calibrage de l'image JPEG $\mathcal{J}$ de facteur de qualité q :

- Récupérer l'image $\hat{\mathcal{I}}^J$ de $\mathcal{J}$ en résolvant le problème d'optimisation de l'équation (A.8) ;
- Découper $\hat{\mathcal{I}}^J$ à partir du 1ème pixel diagonal afin d'obtenir l'image calibrée $\hat{\mathcal{I}}_c$;

- Soustraire les coefficients DCT de l'image $\hat{\mathcal{J}}_c$ à ceux de $\hat{\mathcal{I}}_c$, afin d'estimer le bruit de quantification DCT $\hat{\mathcal{N}}_q$;
- Ajouter $\hat{\mathcal{N}}_q$ à $\hat{\mathcal{I}}^J$ dans le domaine DCT, afin de créer l'image $\hat{\mathcal{F}}_c^J$ avec l'histogramme DCT lissé.

Nous comparons cette méthode de lissage de l'histogramme DCT basée sur le calibrage, à celle proposée dans la section A.5.2. La table A.9 rapporte la divergence K-L obtenue par $\hat{\mathcal{F}}_{bq}^J$ et celle obtenue par $\hat{\mathcal{F}}_c^J$. Les valeur positives dans la table A.9 indiquent que le lissage d'histogramme basé uniquement sur le calibrage surpasse notre précédente méthode (cf. section A.5.2) dans les sous-bandes de basses fréquences. Mais cette nouvelle méthode basée sur le calibrage est beaucoup plus rapide pour créer l'image avec l'histogramme DCT lissé, et est conceptuellement plus simple.

Table A.9: La différence entre la divergence K-L obtenue par $\hat{\mathcal{F}}_{bq}^J$ et celle obtenue par $\hat{\mathcal{F}}_c^J$ pour les 64 sous-bandes DCT. La valeur moyenne de la différence est -0.0239 avec une déviation standard de 0.0431. Les résultat sont obtenus à partir de l'ensemble UCIDTest.

r \ c	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
1	-0.0081	-0.0001	0.0096	0.0121	0.0075	0.0126	0.0171	-0.0265
2	0.0037	0.0081	0.0127	0.0025	-0.0013	0.0201	0.0056	-0.0393
3	0.0171	0.0120	0.0098	-0.0036	-0.0062	-0.0050	-0.0099	-0.0663
4	0.0133	0.0046	-0.0013	-0.0110	-0.0194	-0.0086	-0.0097	-0.0708
5	0.0106	0.0022	-0.0037	-0.0159	-0.0291	-0.0237	-0.0591	-0.1173
6	0.0090	0.0114	-0.0104	-0.0179	-0.0323	-0.0253	-0.0692	-0.1229
7	0.0353	0.0223	-0.0124	-0.0079	-0.0660	-0.0690	-0.0936	-0.1067
8	-0.0050	-0.0332	-0.0730	-0.0720	-0.1385	-0.1372	-0.1235	-0.0357

Dans l'image $\hat{\mathcal{F}}_c^J$, il faut encore atténuer le bruit introduit par la procédure de lissage de l'histogramme DCT. À cette fin, nous proposons le problème suivant :

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{u}} \left\{ \frac{\lambda}{2}\|\mathbf{u}-\mathbf{y}\|^2 + \alpha \times \iota(\mathbf{u}) + \beta \sum_{k=1}^{28}\sum_{c=0}^{7} \left|\nu_k(\mathbf{u}_c) - \hat{\sigma}_k^2\right| + \sum_i \frac{\gamma}{2}\|\mathbf{P}^i\mathbf{u} - \mathbf{z}^i\|^2 - \log p(\mathbf{z}^i) \right\}. \quad \text{(A.9)}$$

Cette fonction de coût est composée de trois types de termes. Le premier terme est relatif à la fidélité à l'image JPEG $\mathbf{y}$. Les deuxième et troisième termes sont destinés aux fins anti-criminalistiques. La dernière partie de cette fonction se rapporte au modèle a priori d'image, en utilisant le GMM dans le cadre EPLL [ZW11]. Ici, λ, α, β et γ sont des paramètres de régularisation, $\mathbf{u}_c$ est l'image calibrée obtenue en coupant $\mathbf{u}$ au $c^{\text{ème}}$ pixel diagonal [FGH02, LB11], $\{\mathbf{z}^i\}$ est un ensemble de variables auxiliaires pour faciliter l'optimisation [ZW11], $\iota(\cdot)$ représente la TV de l'image donnée, $\nu_k(\cdot)$ renvoie la variance de la k-ème sous-bande de hautes

fréquences (définie dans [LB11]) de l'image donnée, tandis que $\hat{\sigma}_k^2$ est la variance estimée de l'image originale de la k-ème sous-bande de hautes fréquences de $\hat{\mathcal{F}}_c^J$.

$\mathcal{F}_1^J$ désigne notre image anti-criminalistique finale créée par la méthode proposée dans cette section. La figure A.15 montre les courbes ROC obtenues par $\mathcal{F}_0^J$ contre des détecteurs scalaires. En comparant la figure A.15 à la figure A.9-(f) et la figure A.13, on constate que la méthode proposée dans cette section ne surpasse pas les deux méthodes proposées précédemment. Cela apparaît également dans la table A.10. Une raison possible est que l'estimation du bruit de quantification $\hat{\mathcal{N}}_q$ est peut-être encore trop rudimentaire : cela introduit un bruit dans le domaine spatial, qui réduit non seulement l'indétectabilité criminalistique mais aussi la qualité de l'image anti-criminalistique finale. La performance de la méthode proposée ici peut être améliorée par une estimation du bruit de quantification précise et l'intégration de davantage de termes anti-criminalistiques dans l'équation (A.9). Par ailleurs, il est à noter que même si la méthode décrite dans cette section n'est pas la meilleure parmi les trois méthodes anti-criminalistiques pour la compression JPEG, elle surpasse encore largement l'état de l'art (cf. la table A.10).

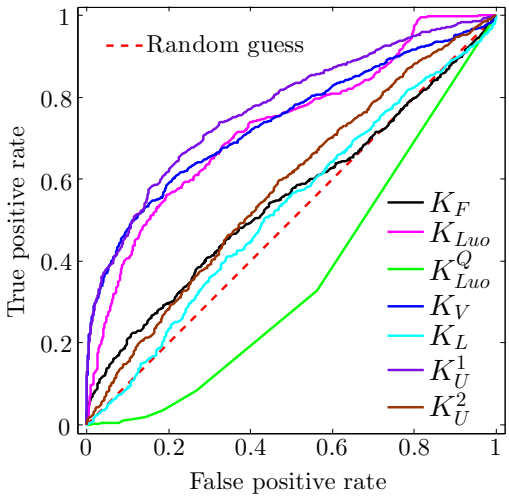


Figure A.15: Les courbes ROC obtenues par $\mathcal{F}_1^J$ contre des détecteurs criminalistiques. Les résultat sont obtenus à partir de l'ensemble UCIDTest.

A.7 Amélioration de la qualité et anti-criminalistique de l'image filtrée par le filtre médian à l'aide d'une déconvolution variationnelle d'image

A.7.1 Introduction et motivation

Nous suivons la ligne de recherche décrite dans la section A.1.3.3, à savoir concevoir des méthodes anti-criminalistiques d'image en s'appuyant sur la restauration d'image. À cette fin, il faut un modèle pour décrire le processus de filtrage médian ainsi qu'un modèle a priori

Table A.10: De la deuxième colonne à la huitième colonne, sont listées les valeurs AUC pour différentes images contre des détecteurs scalaires ; la comparaison de la qualité d'image (avec $\mathcal{I}$ comme référence) est rapportée dans les deux dernières colonnes. Les résultats sont obtenus à partir de l'ensemble UCIDTest.

	K_F	K_{Luo}	K_{Luo}^Q	K_V	K_L	K_U^1	K_U^2	PSNR	SSIM
$\mathcal{J}$	0.9991	1.0000	0.9996	0.9976	0.9811	0.9860	0.8840	37.0999	0.9919
$\mathcal{F}_{S_qS_b}^J$	0.3783	0.0806	0.6288	0.8337	0.5338	0.6309	0.4854	30.4591	0.9509
$\hat{\mathcal{I}}^J$	0.9997	0.9982	0.9528	0.7851	0.9698	0.9878	0.8779	37.8930	0.9927
$\hat{\mathcal{F}}_c^J$	0.9994	0.8147	0.5949	0.7383	0.9868	0.9868	0.8944	35.3209	0.9876
$\mathcal{F}_1^J$	0.5522	0.7291	0.3594	0.7394	0.5272	0.7750	0.5787	35.2568	0.9832
$\mathcal{F}_0^J$	0.6756	0.6046	0.5194	0.6210	0.4490	0.6772	0.5880	35.4814	0.9843
$\mathcal{F}^J$	0.5398	0.6425	0.4598	0.6159	0.4344	0.5894	0.5317	35.9855	0.9866

d'image approprié afin de construire un problème d'optimisation basé sur le MAP pour le problème inverse du filtrage médian.

De fait, le processus de filtrage médian pourrait être décrit comme une procédure de convolution linéaire spatialement hétérogène. Pour une fenêtre locale, le noyau de convolution est une matrice de taille 3×3 dont un seul des 9 éléments prendrait la valeur 1, et les 8 autres éléments seraient tous nuls. La forme du noyau est alors complètement dépendante des statistiques d'ordre des pixels encadrés par la fenêtre du filtre. Pourtant, cette information est définitivement perdue pendant le processus de filtrage médian. Afin de simplifier ce modèle, la convolution avec un noyau spatialement homogène $\mathbf{k}$ est utilisée pour se rapprocher du processus de filtrage médian. L'analyse développée ici donnera le premier terme de l'équation (A.10) dans la section A.7.2.

Dans les méthodes criminalistiques proposées dans l'état de l'art [KF10, Cao+10, Yua11], la différence de valeurs de pixels d'image est souvent la caractéristique de base des détecteurs. En général, l'histogramme de différence de valeurs de pixels (à savoir la dérivée de l'image) a un pic plus haut dans l'image MF que celui de l'image originale, autour de la classe 0 (cf. les exemples dans la figure A.7). Pour régulariser la différence de valeurs de pixels, la loi gaussienne généralisée (paramétrée par le paramètre d'échelle α et le paramètre de forme β) de moyenne nulle est utilisée pour la modéliser. L'analyse de ce paragraphe donnera le dernier terme de l'équation (A.10) dans la section A.7.2.

A.7.2 Déconvolution variationnelle d'image

D'une manière similaire à celle utilisée en anti-criminalistique de compression JPEG [Fan+13b], nous proposons un problème de déconvolution variationnelle d'image afin de créer une image MF anti-criminalistique, basé sur l'analyse développée dans la section A.7.1, et

dans une certaine mesure inspiré par [KF09, KTF11] :

$$\arg\min_{\mathbf{u}} \left(\frac{\lambda}{2} \left(\|\mathbf{Ku} - \mathbf{y}\|_2^2 + \omega \|\mathbf{u} - \mathbf{y}\|_2^2 \right) + \sum_{j=1}^{J} \left\| \frac{\mathbf{F}^j \mathbf{u}}{\alpha_j} \right\|_{\beta_j}^{\beta_j} \right), \tag{A.10}$$

Dans cette équation, $\mathbf{y}$ contient les valeurs des pixels de l'image MF sous forme vectorisée, cette image MF ayant été produite à partir de l'image originale $\mathbf{x}$. La variable $\mathbf{u}$ est utilisée pour représenter génériquement une image. $\mathbf{Ku}$ désigne la multiplication matricielle correspondant à une convolution d'image avec le noyau $\mathbf{k}$. J filtre(s) de dérivée d'image sont utilisés afin de calculer différents types de dérivées d'image. $\mathbf{F}^j\mathbf{u}$ est la multiplication matricielle servant à calculer la j-ème ($j = 1, 2, \cdots, J$) dérivée d'image, en utilisant le j-ème filtre linéaire avec le noyau $\mathbf{f}^j$. α_j et β_j sont les deux paramètres de la loi gaussienne généralisée de moyenne nulle utilisée pour modéliser la j-ème dérivée de l'image. λ et ω sont deux paramètres de régularisation pour équilibrer l'énergie des différents termes.

Il y a trois termes dans l'équation (A.10). Comme discuté ci-dessus, le processus de filtrage médian est simplifié, et approché par une convolution d'image avec un noyau spatialement homogène – ce qui constitue le premier terme de l'équation (A.10). Le deuxième terme est conçu pour que l'image MF anti-criminalistique produite soit dans une certaine mesure proche de l'image MF. Le troisième terme est l'a priori d'image, qui sert à régulariser la dérivée de l'image en utilisant la loi gaussienne généralisée de moyenne nulle. Dans la pratique, on utilise les filtres suivants ($J = 4$) de dérivées d'image: $\mathbf{f}^1 = [1, -1]$, $\mathbf{f}^2 = [1, -1]^T$, $\mathbf{f}^3 = [1, 0, -1]$, et $\mathbf{f}^4 = [1, 0, -1]^T$. D'autres filtres pourraient être ajoutés dans l'équation (A.10), au prix d'une complexité de calcul plus élevée, mais pour un impact mineur sur les résultats finaux.

Pour le terme d'a priori d'image, il faut connaître les valeurs des deux paramètres α et β de la loi gaussienne généralisée de moyenne nulle. En fait, ces deux paramètres sont directement reliés à la variance σ^2 et au kurtosis κ de la loi :

$$\sigma^2 = \frac{\alpha^2 \Gamma(3/\beta)}{\Gamma(1/\beta)}, \quad \kappa = \frac{\Gamma(5/\beta)\Gamma(1/\beta)}{\Gamma(3/\beta)^2}, \tag{A.11}$$

où $\Gamma(\cdot)$ est la fonction gamma. Donc, si la variance et le kurtosis sont connus, les deux paramètres $\hat{\alpha}$ et $\hat{\beta}$ peuvent être estimés en utilisant une méthode numérique [BS99]. Par conséquent, il faut estimer la variance $\hat{\sigma}^2$ et la kurtosis $\hat{\kappa}$ de l'image originale à partir de l'image MF. À cette fin, une régression linéaire est employée pour leur estimation, en utilisant l'image MF donnée et ses versions encore plusieurs fois filtrées par le filtre médian. Pour le j-ème type de filtre de dérivée d'image, la variance $\hat{\sigma}^2(\mathbf{F}^j\mathbf{x})$ et le kurtosis $\hat{\kappa}(\mathbf{F}^j\mathbf{x})$ de l'image originale sont estimés par :

$$\begin{cases} \hat{\sigma}^2(\mathbf{F}^j\mathbf{x}) = \mathbf{c}_1^{\sigma^2} + \sum_{m=1}^{M} \mathbf{c}_{m+1}^{\sigma^2} \times \hat{\sigma}^2 \left(\mathbf{F}^j \mathcal{MF}^{(m)}(\mathbf{x}) \right), \\ \hat{\kappa}(\mathbf{F}^j\mathbf{x}) = \mathbf{c}_1^{\kappa} + \sum_{m=1}^{M} \mathbf{c}_{m+1}^{\kappa} \times \hat{\kappa} \left(\mathbf{F}^j \mathcal{MF}^{(m)}(\mathbf{x}) \right), \end{cases} \tag{A.12}$$

où $\hat{\sigma}^2(\cdot)$ et $\hat{\kappa}(\cdot)$ sont respectivement la variance et le kurtosis de l'echantillon, et $\mathcal{MF}^{(m)}(\cdot)$ représente m applications successives du filtrage médian. $\mathbf{c}_m^{\sigma^2}$ et $\mathbf{c}_m^{\kappa}$ sont les coefficients de la

régression linéaire : ils peuvent être obtenus depuis un ensemble d'images par une procédure d'apprentissage. Dans la pratique, fixer M à 5 donne de bons résultats.

L'optimisation directe de la fonction de coût dans l'équation (A.10) n'est pas triviale. En pratique, elle peut être résolue en utilisant la méthode " Half Quadratic Splitting " [KF09] et la méthode de Bregman [GO09, KTF11]. Après l'introduction d'un ensemble de variables auxiliaires $\{\mathbf{w}^j\}_{j=1}^J$, le problème d'optimisation dans l'équation (A.10) peut être ré-écrit sous la forme suivante :

$$\min_{\mathbf{u},\{\mathbf{w}^j\}} \left(\frac{\lambda}{2} \left(\|\mathbf{K}\mathbf{u} - \mathbf{y}\|_2^2 + \omega \|\mathbf{u} - \mathbf{y}\|_2^2 \right) + \sum_{j=1}^{J} \left(\frac{\gamma}{2} \left\|\mathbf{F}^j\mathbf{u} - \alpha_j \mathbf{w}^j\right\|_2^2 + \left\|\mathbf{w}^j\right\|_{\beta_j}^{\beta_j} \right) \right). \tag{A.13}$$

où γ est un paramètre de régularisation. Ensuite, l'itération de Bregman est appliquée à l'équation (A.13). La méthode de Bregman résout le problème à la $(k+1)$-ème $(k = 0, 1, 2, \cdots)$ itération par les formules de l'équation (A.14), où $\{\mathbf{b}^j\}_{j=1}^J$ sont les variables de Bregman, et $(\mathbf{b}^j)^{(0)} = \mathbf{0}$.

$$\begin{cases} \left(\mathbf{u}^{(k+1)}, \{\left(\mathbf{w}^j\right)^{(k+1)}\}\right) = \arg\min\limits_{\mathbf{u},\{\mathbf{w}^j\}} \left(\frac{\lambda}{2} \left(\|\mathbf{K}\mathbf{u} - \mathbf{y}\|_2^2 + \omega \|\mathbf{u} - \mathbf{y}\|_2^2 \right) \right. \\ \qquad\qquad\qquad\qquad \left. + \sum\limits_{j=1}^{J} \left(\frac{\gamma}{2} \left\|\mathbf{F}^j\mathbf{u} + \left(\mathbf{b}^j\right)^{(k)} - \alpha_j \mathbf{w}^j\right\|_2^2 + \left\|\mathbf{w}^j\right\|_{\beta_j}^{\beta_j} \right) \right), \\ (\mathbf{b}^j)^{(k+1)} = (\mathbf{b}^j)^{(k)} + \left(\mathbf{F}^j\mathbf{u}^{(k+1)} - \alpha_j (\mathbf{w}^j)^{(k+1)}\right). \end{cases} \tag{A.14}$$

A.7.3 Amélioration de qualité de l'image MF

En paramétrant de manière appropriée, nous pouvons créer une image de meilleure qualité à partir de l'image MF, en résolvant le problème d'optimisation dans l'équation (A.14). Pour estimer $\mathbf{k}$ dans l'équation (A.10), 6 types de noyaux ont été étudiés expérimentalement sur l'ensemble MFTE100 : le noyau moyenne, le noyau gaussien, le noyau inspiré par [Yua11], et, pour chaque image, les trois noyaux estimés en utilisant les trois noyaux précédents comme estimation initiale pour la méthode [KTF11]. Les résultats expérimentaux montrent qu'un bon choix pour $\mathbf{k}$ est le noyau estimé par [KTF11] en utilisant le noyau moyenne comme estimation initiale. Pour l'amélioration de qualité de l'image MF, le paramétrage suivant est utilisé : $\omega = 0.4$, $\lambda = 1000$ et $\gamma = 1200$. Ces paramètres sont obtenus depuis l'ensemble MFTE100.

La table A.11 compare la qualité d'image de l'image bruitée par le bruit sel et poivre, celle puis filtrée par le filtre médian, puis à celle traitée par la méthode proposée. L'efficacité de la déconvolution variationnelle d'image décrite dans la section A.7.2 est bien validée : la qualité de l'image MF est améliorée en utilisant les métriques d'évaluation PSNR et SSIM. Ce fait

est aussi reflété par les valeurs PSNR/SSIM pour l'image $\mathcal{M}^p$ listées dans la table A.12, en comparaison avec l'image MF $\mathcal{M}$.

Table A.11: Les valeurs moyennes du PSNR et du SSIM de l'image bruitée par le bruit sel et poivre, celle puis filtrée par le filtre médian, et enfin celle traitée par la méthode proposée, respectivement. La densité de bruit varie de 1% à 7%. Les résultats sont obtenus à partir de l'ensemble MFTE.

		Bruitée	Filtrée	Améliorée
1%	PSNR	25.1365	37.1336	38.0723
	SSIM	0.8308	0.9827	0.9892
2%	PSNR	22.1257	36.9719	37.8236
	SSIM	0.7185	0.9822	0.9885
3%	PSNR	20.3642	36.7957	37.5155
	SSIM	0.6388	0.9818	0.9876
4%	PSNR	19.1161	36.6114	37.2058
	SSIM	0.5793	0.9813	0.9867
5%	PSNR	18.1466	36.4031	36.7914
	oSSIM	0.5327	0.9807	0.9850
6%	PSNR	17.3540	36.1758	36.2933
	SSIM	0.4948	0.9801	0.9828
7%	PSNR	16.6851	35.8924	35.7542
	SSIM	0.4632	0.9793	0.9803

Table A.12: Du deuxième au troisième rang, sont listées les valeurs moyennes de PSNR/SSIM; les valeurs AUC pour différentes images contre des détecteurs scalaires sont rapportées dans les quatre rangs suivants ; les divergences K-L pour différents histogrammes de différence de valeurs de pixels sont listées dans les deux derniers rangs. Les résultats sont obtenus à partir de l'ensemble UCIDTest.

		$\mathcal{M}$	$\mathcal{M}^p$	$\mathcal{F}_W^M$	$\mathcal{F}_D^M$	$\mathcal{F}^M$
Qualité d'image	PSNR	37.2847	38.2953	33.6033	33.4272	37.5184
	SSIM	0.9831	0.9896	0.9552	0.9714	0.9901
Performance anti-criminalistique	K_K	0.9722	0.7839	0.4592	0.5347	0.5595
	$\hat{K}_K$	0.9824	0.8375	0.6586	0.4635	0.5061
	K_C	0.9938	0.8213	0.6668	0.7479	0.6490
	K_Y	0.9984	0.7922	0.3336	0.6518	0.5886
divergence K-L	$\mathbf{f}^1$	0.1632	0.0990	0.1148	0.0547	0.0484
	$\mathbf{f}^2$	0.1611	0.0949	0.1338	0.0563	0.0449
	$\mathbf{f}^3$	0.0775	0.0525	0.0619	0.0383	0.0272
	$\mathbf{f}^4$	0.0753	0.0495	0.0689	0.0389	0.0238

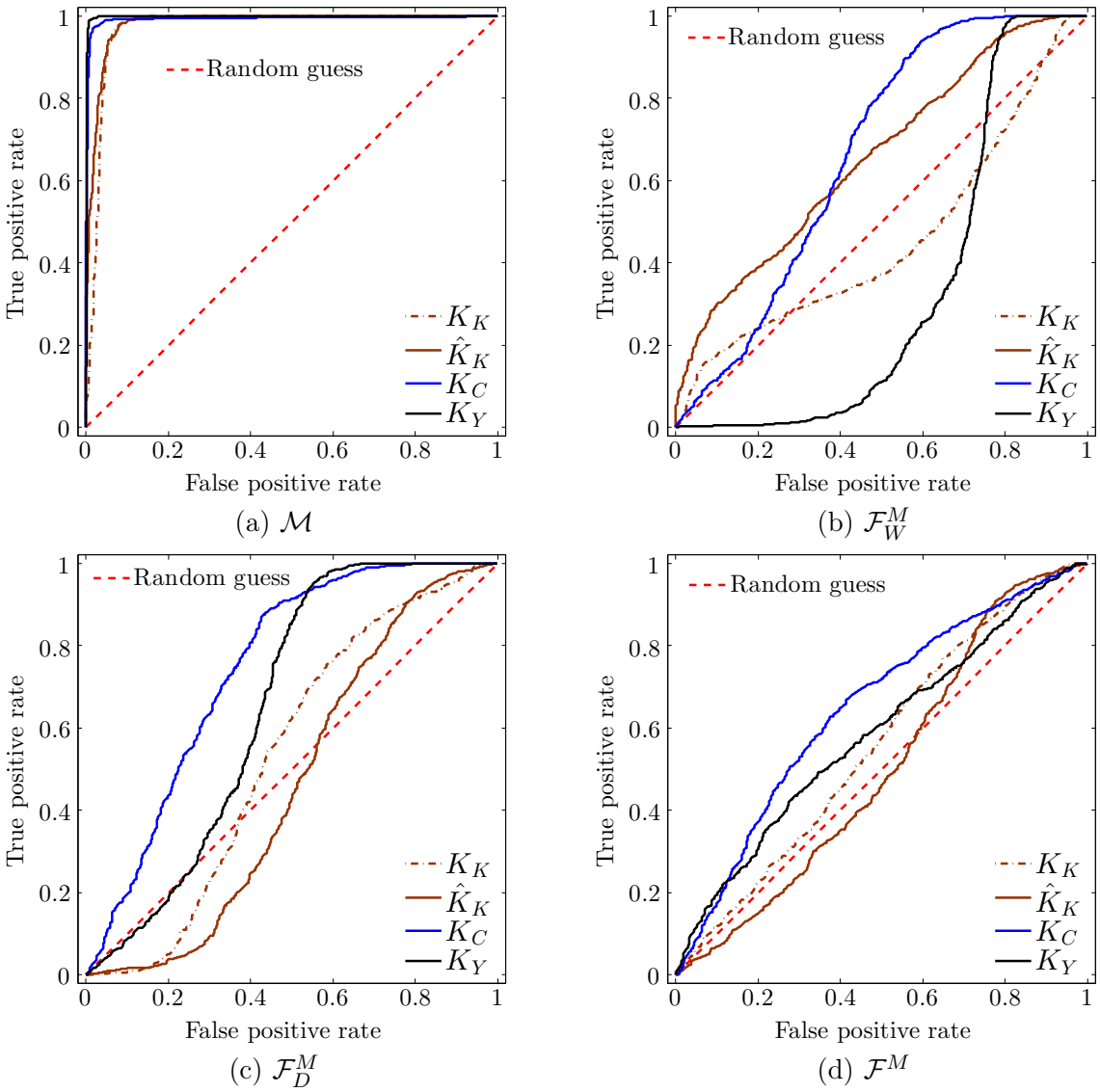
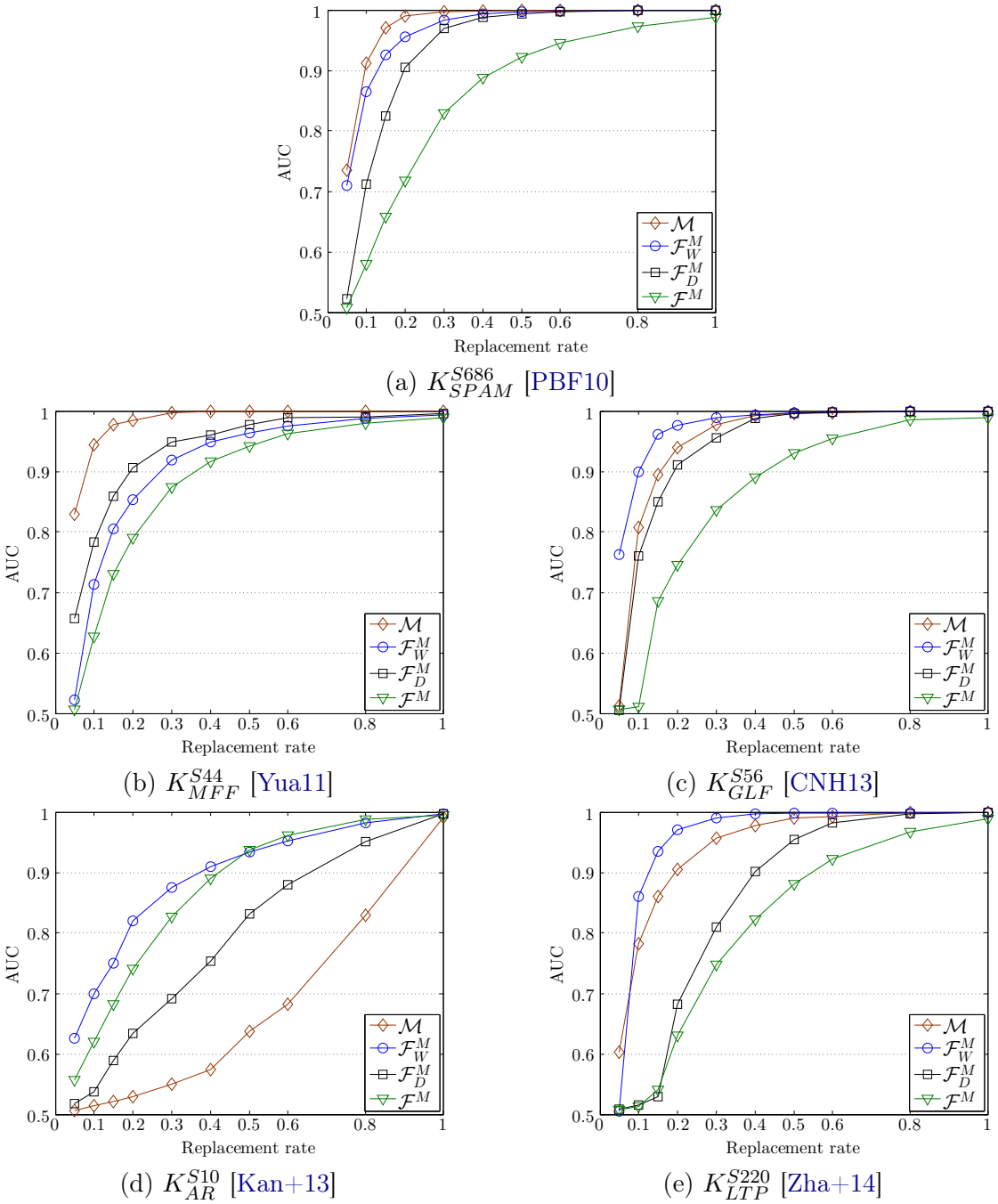


Figure A.16: Courbes ROC obtenues pour : l'image MF $\mathcal{M}$, l'image produite par Wu *et al.* $\mathcal{F}_W^M$ [WSL13], Dang-Nguyen *et al.* $\mathcal{F}_D^M$ [DN+13], et la méthode proposée $\mathcal{F}^M$, contre les quatre détecteurs scalaires K_K [KF10], $\hat{K}_K$ [KF10], K_C [Cao+10], et K_Y [Yua11]. Les résultats sont obtenus à partir de l'ensemble MFTE.

A.7.4 Anti-criminalistique de filtrage médian

En résolvant le problème d'optimisation de l'équation (A.10) sur l'image MF, il est possible de créer une image MF anti-criminalistique dotée d'une bonne immunité à la détection anti-criminalistique, mais sa qualité visuelle laissera à désirer. Afin d'obtenir une image MF anti-criminalistique présentant une bonne immunité sans beaucoup de diminution de la qualité d'image, cette section propose une stratégie de perturbation légère de la valeur des pixels de l'image MF, avant qu'elle soit utilisée dans l'équation (A.10).



(a) K_{SPAM}^{S686} [PBF10]

(b) K_{MFF}^{S44} [Yua11]

(c) K_{GLF}^{S56} [CNH13]

(d) K_{AR}^{S10} [Kan+13]

(e) K_{LTP}^{S220} [Zha+14]

Figure A.17: La valeur AUC en fonction du taux de remplacement d'image pour différents types d'images, lorsqu'elles sont testées en utilisant les détecteurs à base de SVM. Les résultats sont obtenus à partir de MFTR pour l'entraînement et à partir de MFTE pour les tests.

Les sorties de K_K [KF10], K_C [Cao+10] et K_Y [Yua11] sont toutes implicitement ou explicitement reliées à une différence de valeurs de pixels, elles réagissent d'autant plus fortement dans les régions texturées de l'image. Le but ici est de réduire la fréquence d'occurrence de

la différence 0 par une modification légère de l'image MF. Si trois pixels adjacents vérifient $\mathbf{y}_{i+1} = \mathbf{y}_{i+2} = \mathbf{y}_{i+3}$, la modification de $\mathbf{y}_{i+2}$ à $\mathbf{y}_{i+2} \pm 1$ changera deux différences de valeurs de pixel de 0 à ± 1. De façon similaire, si deux pixels adjacents ont la même valeur, le pixel qui a la variance locale plus élevée que l'autre est modifié. De cette manière, la fréquence d'occurrence de la différence 0 est efficacement réduite, avec un impact très limité sur la qualité de l'image.

De même qu'à la section A.7.3, pour $\mathbf{k}$ nous utilisons à nouveau le noyau estimé par [KTF11], avec le noyau moyenne comme estimation initiale. Afin de créer l'image MF anti-criminalistique en résolvant le problème d'optimisation de l'équation (A.13), le paramétrage suivant est utilisé : $\omega = 0.1$, $\lambda = 1500$ et $\gamma = 500$. Ces paramètres sont obtenus depuis l'ensemble MFTE100.

Pour des raisons de concision, $\mathcal{F}^M$ représente l'image MF anti-criminalistique créée à partir de l'image MF par la méthode proposée. La table A.12 rapporte la qualité d'image (les valeurs de PSNR et de SSIM sont calculées en utilisant l'image originale comme référence) et la performance anti-criminalistique. Les courbes ROC obtenues pour différents types d'images et pour les détecteurs K_K [KF10], K_C [Cao+10] ou K_Y [Yua11] sont tracées dans la figure A.16. Les images MF anti-criminalistiques $\mathcal{F}^M$ créées par la méthode proposée induisent, pour les détecteurs criminalistiques, des courbes ROC qui sont au plus proche d'une décision aléatoire. Ce fait est également reflété par des valeurs AUC proches de 0.5 dans la table A.12 pour $\mathcal{F}^M$. De plus, la méthode proposée produit une meilleure qualité d'image que les autres images MF anti-criminalistiques de l'état de l'art [WSL13, DN+13]. Enfin, la table A.12 rapporte la divergence K-L obtenue par $\mathcal{F}^M$ et celles obtenues par les images anti-criminalistiques de l'état de l'art. Les valeurs de la divergence K-L obtenues par $\mathcal{F}^M$ sont plus petites, indiquant une meilleure performance de la méthode proposée afin de restaurer l'histogramme de différence de valeurs de pixels que l'état de l'art. Les courbes AUC dans la figure A.17 montrent que notre image anti-criminalistique a une meilleure indétectabilité criminalistique globale contre les détecteurs à base de SVM que l'état de l'art, sauf contre K_{AR}^{S10} [Kan+13].

A.8 Conclusions et perspectives

A.8.1 Résumé des contributions

Dans cette thèse, nous avons présenté nos travaux de recherche sur l'anti-criminalistique d'image sur la compression JPEG et le filtrage médian. Au cours de notre étude de l'anti-criminalistique d'image concernant le codage ou le traitement d'images, on constate des similitudes entre l'anti-criminalistique d'image et la restauration d'images. Toutes les deux visent à récupérer au mieux les informations perdues lors de la dégradation de l'image. Cependant, en plus de la qualité d'image, l'anti-criminalistique d'image a un autre objectif indispensable, *i.e.* : une bonne indétectabilité contre les détecteurs criminalistiques. À cette fin, nous introduisons des concepts/méthodes avancés de la restauration d'image pour concevoir de nouvelles méthodes anti-criminalistiques, en intégrant certains termes/stratégies anti-criminalistiques. Les résultats expérimentaux montrent que les méthodes proposées surpassent les méthodes

de l'état de l'art pour créer des images anti-criminalistiques avec une meilleure indétectabilité criminalistique contre les détecteurs existants, ainsi qu'une meilleure qualité d'image. Bien que la restauration d'image elle-même ne soit pas notre sujet principal de recherche dans cette thèse, nous proposons également deux méthodes d'amélioration de la qualité d'image pour la compression JPEG et le filtrage médian, respectivement. Elles servent d'étape préliminaire dans les méthodes anti-criminalistiques proposées, mais sont également avérés avoir de bonnes performances en termes de gain de PSNR/SSIM par rapport à l'image JPEG ou l'image MF.

Les contributions de cette thèse sont résumées comme suit.

Proposer une nouvelle ligne de recherche pour concevoir des méthodes anti-criminalistiques d'image via la restauration d'image : Après l'examen de la littérature, nous trouvons courant d'utiliser le traitement d'image simple pour cacher les traces laissées par une opération d'image ciblée aux fins d'anti-criminalistique. Par exemple, le filtrage médian est utilisé pour l'anti-criminalistique JPEG afin d'éliminer les artefacts de blocage. L'injection de bruit est utilisée pour cacher les traces de filtrage médian. Ces méthodes peuvent être efficaces pour attaquer détecteurs criminalistiques ciblés, mais peuvent par contre être détectées par plusieurs détecteurs avancés. En outre, l'image anti-criminalistique résultante souffre d'une faible qualité visuelle, ce qui peut susciter spontanément le doute sur son authenticité. Dans cette thèse, nous proposons une nouvelle ligne de recherche en s'appuyant sur les concepts/méthodes avancés de la restauration d'image afin de concevoir des méthodes d'anti-criminalistique d'image. Étant donnée une image dégradée par une opération irréversible de traitement d'image, par exemple, la compression JPEG ou le filtrage médian, l'objectif de l'anti-criminalistique d'image est de cacher les traces de traitement d'image tout en conservant une bonne qualité d'image. En se fondant sur les similitudes entre l'anti-criminalistique d'image et la restauration d'image, les cadres de travail et les termes de l'optimisation introduits en restauration d'image aident à supprimer en partie les traces, tout en gardant une bonne qualité d'image (même meilleur parfois) par rapport à l'image dégradée donnée. La bonne performance anti-criminalistique est obtenue en ajoutant des termes anti-criminalistiques au cadre de l'optimisation ou en utilisant des stratégies anti-criminalistiques supplémentaires. Nous suivons cette ligne de recherche tout au long de cette thèse pour l'anti-criminalistique de compression JPEG ainsi que l'anti-criminalistique de filtrage médian. Sa supériorité sur les méthodes anti-criminalistiques de l'état de l'art est bien prouvée par des résultats expérimentaux, avec une meilleure indétectabilité criminalistique et une meilleure qualité de l'image traitée. À notre connaissance, nous sommes les premiers à utiliser systématiquement la restauration aux fins d'anti-criminalistique. Nous espérons également que cette nouvelle ligne de recherche serait utile pour plusieurs problèmes d'anti-criminalistique d'image relatifs à d'autres traitements, par exemple le rééchantillonnage, l'amélioration du contraste, *etc.*

L'anti-criminalistique JPEG en utilisant le déblocage basé sur la TV : Nous avons proposé une méthode anti-criminalistique de compression JPEG en utilisant la méthode de déblocage basé sur la TV. Cela est mis en œuvre par l'optimisation d'un problème de minimisation contraint basé sur la TV avec un terme de TV et un terme de mesure du déblocage par la TV. Pendant ce temps, la qualité d'image traitée est contrôlée par une projection QCS

modifiée. En outre, un détecteur puissant à base de calibrage est trompé par la minimisation d'une fonction de coût proche de la fonction de calculant la caractéristique basée sur le calibrage. Cette méthode parvient à bien éliminer les artefacts de blocage dans le domaine spatial. En outre, les artefacts de quantification dans le domaine DCT sont aussi largement atténués, dans la mesure où l'image anti-criminalistique créée est capable de passer comme n'ayant jamais été compressée face à des détecteurs scrutant les artefacts de quantification.

L'anti-criminalistique améliorée de compression JPEG avec une lissage perceptuel d'histogramme DCT : En effet, la méthode de déblocage JPEG basée sur la TV précédemment proposée est capable de réaliser une bonne indétectabilité criminalistique contre tous les détecteurs criminalistiques actuels de compression JPEG existants, y compris ceux qui examinant les artefacts de quantification. Cependant, l'histogramme DCT n'est pas toujours bien lissé, en particulier dans les sous-bandes de moyennes fréquences. Cette faiblesse n'en est pas une face aux détecteurs criminalistiques actuels de compression JPEG, mais peut être détectée par des algorithmes criminalistiques potentiellement plus avancés. Afin de remédier à ce problème, nous avons proposé une méthode anti-criminalistique améliorée en quatre étapes avec un lissage perceptuel de l'histogramme DCT. À l'aide des informations DCT estimées en partie après le déblocage JPEG basé sur la TV, une méthode de tramage adaptatif local est proposée en combinant la loi de Laplace et la loi uniforme. Les coefficients DCT sont modifiés par la résolution d'un problème d'affectation simplifié qui minimise la perte de qualité SSIM. On effectue un dernier déblocage léger basé sur la TV et le décalibrage pour améliorer encore l'indétectabilité criminalistique avec une perte de qualité d'image très mineure.

L'amélioration de qualité et l'anti-criminalistique de l'image JPEG basées sur un modèle d'image avancé : Nous avons proposé une autre méthode anti-criminalistique JPEG basé sur un modèle a priori d'image sophistiqué, qui est décrite par les étapes suivantes :

- *L'amélioration de qualité d'image JPEG en utilisant un modèle a priori d'image sophistiqué* : L'efficacité de l'utilisation de la TV dans l'anti-criminalistique JPEG nous motive à utiliser les modèles a priori d'image plus sophistiqués, avec l'espoir d'améliorer les performances d'anti-criminalistique JPEG. À cette fin, nous utilisons le cadre EPLL avec le GMM comme a priori pour les *patches* d'image. Le bruit de compression dans le domaine spatial est modélisé en utilisant la loi gaussienne multivariée de moyenne nulle, dont 64 types de matrices de covariance sont apprises pour chaque facteur de qualité JPEG. Par conséquent, un procédé d'amélioration de qualité d'image JPEG a été proposé en minimisant la fonction de coût en tenant compte des deux termes ci-dessus. L'image JPEG de qualité améliorée peut être obtenue en résolvant un problème d'optimisation à l'aide d'une étape d'estimation approximative du MAP. Cette méthode nous permet d'obtenir un bon gain de qualité d'image pour l'image JPEG, en particulier pour la compression JPEG très forte.

- *Le lissage non paramétrique d'histogramme DCT basé sur le calibrage* : En plus de notre méthode de lissage perceptuel d'histogramme DCT précédemment proposé, nous proposons un autre lissage d'histogramme DCT basé cette fois sur le calibrage, sans utiliser aucun modèle statistique. Basé sur les statistiques d'image et l'efficacité de

l'astuce du calibrage, le bruit de quantification dans le domaine DCT peut être estimé en découpant légèrement l'image JPEG de qualité améliorée et en recompressant l'image calibrée. L'histogramme DCT est lissé en ajoutant le bruit de quantification DCT estimé à partir de l'image JPEG de qualité améliorée. Cette nouvelle méthode de lissage non-paramétrique d'histogramme DCT ne permet pas d'obtenir une meilleure performance globale, mais a de meilleures performances dans les sous-bande DCT de basses fréquences que la méthode de lissage perceptuel précédemment proposée. En outre, la méthode basée sur le calibrage a un coût de calcul beaucoup plus faible que la perceptuelle.

- L'indétectabilité criminalistique est considérée au travers de la minimisation d'une fonction de coût avec un terme d'image de fidélité, une terme d'a priori d'image à base d'EPLL, et plusieurs termes anti-criminalistiques inspirés des algorithmes criminalistiques existants.

Pour récapituler, la méthode proposée pour l'amélioration de la qualité JPEG est illustrée par des expériences sur 4 images classiques de test et sur l'ensemble UCIDTest pour améliorer la qualité de l'image JPEG. En effet, la méthode anti-criminalistique proposée doit être améliorée, mais surpasse encore les méthodes de l'état de l'art proposées par Stamm *et al.* [Sta+10a, Sta+10b, SL11] en termes d'indétectabilité criminalistique ainsi que de qualité de l'image traitée.

L'amélioration de qualité et l'anti-criminalistique de l'image MF via une déconvolution variationnelle d'image : Nous avons proposé une déconvolution variationnelle d'image pour à la fois l'amélioration de qualité de l'image MF et l'anti-criminalistique de filtrage médian. Un noyau de convolution spatialement homogène est utilisé pour approcher le processus de filtrage médian. En ce qui concerne l'a priori d'image, la loi gaussienne généralisée de moyenne nulle est utilisée pour modéliser la différence de valeurs de pixels dont les statistiques changement considérablement après filtrage médian. Basé sur l'analyse ci-dessus, la fonction de coût de minimisation proposée est composée d'un terme de convolution, d'un terme de fidélité d'image par rapport à l'image MF, et d'un terme d'a priori. L'image MF de qualité améliorée peut être obtenue en résolvant ce problème de minimisation, avec une qualité d'image meilleure que l'image MF. Avec un autre paramétrage et une perturbation de valeur de pixel supplémentaire, l'image anti-criminalistique est également générée depuis l'image MF. La méthode proposée est capable d'atteindre une indétectabilité criminalistique contre les détecteurs criminalistiques actuels de filtrage médian et une qualité d'image meilleures par rapport aux méthodes anti-criminalistiques de filtrage médian de l'état de l'art.

En résumé, la caractéristique/nouveauté principale de cette thèse est d'introduire des concepts/méthodes avancés de la restauration d'image pour concevoir des méthodes d'anti-criminalistique d'image. Plus précisément, nous avons conçu des méthodes anti-criminalistiques pour la compression JPEG et le filtrage médian, misant sur les éléments suivants de la restauration d'image : la TV, le cadre de EPLL avec le GMM comme modèle d'a priori pour les *patches* d'image, et le cadre de déconvolution d'images fondé sur le MAP. Pour un problème anti-criminalistique d'image donné, certains termes/stratégies anti-criminalistiques sont intégrés pour tromper les détecteurs existants. Pour la compression

JPEG, un terme de mesure de blocage JPEG basé sur la TV est utilisé pour le déblocage JPEG. En outre, deux méthodes de lissage d'histogramme DCT sont proposées pour supprimer les artefacts de quantification dans le domaine DCT : une procédure de lissage perceptuel de l'histogramme DCT, et une procédure non paramétrique de lissage de l'histogramme DCT basé sur le décalibrage. Certains termes anti-criminalistiques inspirés des algorithmes criminalistiques existants sont également intégrés dans le cadre de l'optimisation à base de MAP, pour tromper les détecteurs criminalistiques. Quant au filtrage médian, le modèle d'a priori d'image est choisi spécifiquement afin de régulariser les différences de valeurs de pixels d'image, qui sont explicitement ou implicitement utilisées dans les algorithmes criminalistiques existants. En outre, une procédure de perturbation de valeur de pixel est également proposée afin d'améliorer encore l'indétectabilité criminalistique, avec perte très mineure de qualité d'image.

A.8.2 Perspectives

À court terme, les perspectives de cette thèse comprennent l'amélioration de la performance de l'anti-criminalistique d'image, et la poursuite de la ligne de recherche proposée pour concevoir des méthodes anti-criminalistiques concernant d'autres traitements.

L'amélioration de l'anti-criminalistique d'image JPEG pour générer l'image anti-criminalistique avec une qualité plus haute que l'image JPEG : Dans les sections A.4-A.5, la TV est employée depuis la restauration d'image aux fins d'anti-criminalistique JPEG. La TV peut être considéré comme un a priori d'image simple mais efficace. La section A.6 présente la suite logique sur l'anti-criminalistique de compression JPEG, où un a priori plus sophistiqué que la TV est utilisé. Cependant, la nouvelle méthode anti-criminalistique JPEG ne surpasse pas celles basées sur la TV. L'analyse et les possibles raisons en sont fournies à la fin de la section A.6.3. Un problème anti-criminalistique difficile mais très intéressant pour la compression JPEG est de savoir si on peut créer l'image anti-criminalistique avec une qualité encore plus élevé que l'image JPEG, ne serait-ce que pour montrer la relativité de ces mesures. Une direction de recherche possible pourrait être la combinaison de l'amélioration de la qualité de l'image JPEG, le déblocage basé sur la TV, et le lissage perceptuel d'histogramme DCT. En fait, un objectif similaire a déjà été réalisé dans la section A.7.4 pour le filtrage médian : l'image anti-criminalistique MF a une qualité encore plus élevée que l'image MF dans le test sur l'ensemble MFTE.

Le développement d'autres méthodes anti-criminalistiques en misant sur la restauration d'image : Dans cette thèse, nous avons choisi de travailler sur la compression JPEG et le filtrage médian pour l'anti-criminalistique d'image. En traitement d'image, il existe beaucoup d'autres opérations dont le problème d'anti-criminalistique peut également être formulé comme un problème inverse mal posé. Par exemple, l'anti-criminalistique du rééchantillonnage peut partager certaines similitudes avec la super-résolution [PPK03]. L'anti-criminalistique d'amélioration du contraste peut être liée à l'estimation de la transformation de la luminosité des pixels [ZL14]. Idéalement, une bonne performance anti-criminalistique ne peut être atteinte que si des concepts/méthodes de la restauration d'image sont bien combinés avec

certains termes/stratégies anti-criminalistiques.

À long terme, l'un des buts ultimes de la criminalistique d'image et de l'anti-criminalistique est de développer des méthodes universelles sans viser des méthodes (anti-)criminalistiques spécifiques. Cela afin d'éviter au maximum un éternel " jeu du chat et de la souris " entre la criminalistique et l'anti-criminalistique. Bien que le travail de recherche de criminalistique (ou d'anti-criminalistique) d'image s'étale maintenant sur plus d'une décennie, il existe très peu de méthodes universelles. Actuellement, la seule méthode anti-criminalistique d'image est probablement celle proposée par Barni *et al.* [BFT12]. Toutefois, elle fait aussi l'hypothèse que les détecteurs criminalistiques examinent seulement les statistiques du premier ordre. Nous sommes conscients que les méthodes anti-criminalistiques proposées appartiennent tous à la catégorie " ciblées " selon la classification d'anti-criminalistique d'image proposée par Böhme et Kirchner [BK13]. Cependant, nous croyons que les méthodes anti-criminalistiques proposées ont le potentiel requis pour être généralisées en vue du développement d'une anti-criminalistique d'image universelle.

Vers une anti-criminalistique d'image universelle: Dans les sections A.6-A.7, les problèmes d'optimisation à base de MAP sont proposées pour calculer les images anti-criminalistiques de compression JPEG et de filtrage médian, respectivement. L'universalité du cadre de restauration d'image variationnelle peut être vu à travers ces deux problèmes. En matière de codage/traitement d'images, l'objectif de l'anti-criminalistique est de créer une image anti-criminalistique qui soit aussi " naturelle " que possible. À cette fin, le problème de la génération d'image " naturelle " peut être formulé comme un problème de restauration d'image, pour obtenir une image qui semble n'avoir jamais été traitée. Afin de résoudre ce type de problèmes inverses, on peut utiliser des méthodes statistiques comme l'estimateur du MAP. Dans ce cadre, un bon modèle d'a priori d'image est indispensable. Par ailleurs, le terme de vraisemblance, qui décrit le processus de dégradation de l'image causée par un certain traitement d'image, devrait varier en fonction de différents problèmes anti-criminalistiques. " L'universalité " ici est que le cadre anti-criminalistique est générique : il peut être utilisé pour cacher les traces laissées par diverses autres opérations de traitement d'image.

En outre, certains autres problèmes de recherche ouverts et étroitement liés à cette thèse sont répertoriés comme suit :

- Est-il possible de concevoir une attaque en une seule étape pour l'anti-criminalistique d'image JPEG, compte tenu que plusieurs détecteurs criminalistiques travaillent dans deux domaines différents ?
- Comment peut-on estimer le noyau de convolution spatialement hétérogène pour le filtrage médian ?
- Est-il possible de créer une image anti-criminalistique, de sorte que dans son ensemble elle puisse paraître comme jamais traitée face aux détecteurs à base de méthodes d'apprentissage automatique ?

Ces questions soulignent également quelques directions futures de recherche intéressantes dans

le vaste domaine de l'anti-criminalistique d'image. En outre, concernant le lien étroit entre l'image et la vidéo, les méthodes d'anti-criminalistique d'image proposées sur la compression JPEG et le filtrage médian peuvent également certainement contribuer à l'anti-criminalistique de vidéo.

Dans cette thèse, nous procédons à l'étude de l'anti-criminalistique d'image de compression JPEG et de filtrage médian, en s'inspirant de la restauration d'image. Ces deux problèmes spécifiques d'anti-criminalistique d'image constituent seulement une petite fraction de l'anti-criminalistique d'image. Nous sommes également conscients que cette ligne de recherche nouvellement formée peut être appliquée uniquement aux problèmes d'anti-criminalistique d'image, où le codage/traitement d'image est impliqué. Pour les autres problèmes d'anti-criminalistique d'image, par exemple, celui basé sur la physique ou la géométrie [Far09a], cette ligne de recherche peut être inadéquate. Par exemple, dans notre travail d'anti-criminalistique d'image basé sur l'estimation environnementale de lumière [Fan+12], c'est une tout autre ligne de recherche qui est suivie.

En conclusion, pour les problèmes d'anti-criminalistique d'image impliquant des codages/traitements d'image, nous croyons que les statistiques d'images naturelles sont essentielles dans la création d'une image anti-criminalistique avec une bonne indétectabilité criminalistique ainsi qu'une haute qualité d'image. À cette fin, certains concepts/méthodes avancés de la restauration d'image sont introduits et combinés avec des termes/stratégies anti-criminalistiques pour la compression JPEG et le filtrage médian. Les travaux de recherche présentés dans cette thèse ne représentent qu'une petite partie du potentiel énorme des statistiques d'images naturelles appliquées à l'anti-criminalistique d'image, la criminalistique d'image et la restauration d'image. Böhme et Kirchner [BK13] soulignent également l'importance du modèle statistique d'image naturelle dans la bataille entre la criminalistique et l'anti-criminalistique. Par conséquent, en dépit de beaucoup de points sujets à amélioration, notre travail peut servir de bon point de départ pour de futures recherches sur un large éventail de problèmes criminalistiques et anti-criminalistiques.

Bibliography

[ADF05] F. Alter, S. Durand, and J. Froment. "Adapted total variation for artifact free decompression of JPEG images". *Journal of Mathematical Imaging and Vision* 23, 2 (2005), pp. 199–211 (Cited on pages xxii, 6, 26, 35, 48, 53, 55–59, 62, 70, 97, 166, 178).

[Aig] *Huaqi: the secret to self-innovation in the market.* `http://tech.sina.com.cn/it/2005-12-21/1432798682.shtml` (in Chinese) (Cited on pages 2, 161).

[BFP11] P. Bas, T. Filler, and T. Pevný. "Break our steganographic system — the ins and outs of organizing BOSS". In: *Proceedings of the International Conference on Information Hiding (IH).* Santa Barbara, California, USA, 2011, pp. 59–70 (Cited on pages 23, 24, 49, 50).

[BFT12] M. Barni, M. Fontani, and B. Tondi. "A universal technique to hide traces of histogram-based image manipulations". In: *Proceedings of the ACM Workshop on Multimedia and Security (MMSec).* Coventry, UK, 2012, pp. 97–104 (Cited on pages 77, 157, 202).

[BK13] R. Böhme and M. Kirchner. "Counter-forensics: attacking image forensics". In: *Digital Image Forensics.* Ed. by H. T. Sencar and N. Memon. New York: Springer, 2013, pp. 327–366 (Cited on pages 3, 12–14, 19, 20, 49, 53, 125, 157, 158, 163, 167–169, 202, 203).

[Bov87] A. C. Bovik. "Streaking in median filtered images". *IEEE Transactions on Acoustics, Speech and Signal Processing* 35, 4 (1987), pp. 493–503 (Cited on pages 40, 41, 176).

[BP12a] T. Bianchi and A. Piva. "Detection of nonaligned double JPEG compression based on integer periodicity maps". *IEEE Transactions on Information Forensics and Security* 7, 2 (2012), pp. 842–848 (Cited on pages 24, 89, 90, 92, 93, 96).

[BP12b] T. Bianchi and A. Piva. "Image forgery localization via block-grained analysis of JPEG artifacts". *IEEE Transactions on Information Forensics and Security* 7, 3 (2012), pp. 1003–1017 (Cited on pages 4, 24, 89, 90, 93–95, 164).

[BS99] R. W. Buccigrossi and E. P. Simonceli. "Image compression via joint statistical characterization in the wavelet domain". *IEEE Transactions on Image Processing* 8, 12 (1999), pp. 1688–1701 (Cited on pages 128, 192).

[BV04] S. Boyd and L. Vandenberghe. *Convex optimization.* New York: Cambridge University Press, 2004 (Cited on page 55).

[BXM03] S. Boyd, L. Xiao, and A. Mutapcic. *Subgradient methods.* Tech. rep. EE392o. Stanford University, 2003 (Cited on pages 25, 26, 173).

[Cao+10] G. Cao, Y. Zhao, R. Ni, L. Yu, and H. Tian. "Forensic detection of median filtering in digital images". In: *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME).* Singapore, 2010, pp. 89–94 (Cited on pages 42, 44, 45, 127, 137, 138, 140–143, 146, 147, 150, 173, 177, 191, 195–197).

[Cha09] R. Chartrand. "Fast algorithms for nonconvex compressive sensing: MRI reconstruction from very few data". In: *Proceedings of IEEE International Symposium on Biomedical Imaging: From Nano to Macro (ISBI)*. Boston, Massachusetts, USA, 2009, pp. 262–265 (Cited on pages 29, 131).

[Che+12] C. Chen, J. Ni, R. Huang, and J. Huang. "Blind median filtering detection using statistics in difference domain". In: *Proceedings of the International Conference on Information Hiding (IH)*. Berkeley, California, USA, 2012, pp. 1–15 (Cited on pages 43, 45, 127, 177).

[CL11] C.-C. Chang and C.-J. Lin. "LIBSVM: a Library for support vector machines". *ACM Transactions on Intelligent Systems and Technology* 2, (3 2011). Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`, 27:1–27:27 (Cited on pages 18, 19, 90, 144).

[CN11] C. Chen and J. Ni. "Median filtering detection using edge based prediction matrix". In: *Proceedings of the International Workshop on Digital Forensics and Watermarking (IWDW)*. Atlantic City, New Jersey, USA, 2011, pp. 361–375 (Cited on pages 43, 127).

[CNH13] C. Chen, J. Ni, and J. Huang. "Blind detection of median filtering in digital images: a difference domain based approach". *IEEE Transactions on Image Processing* 22, 12 (2013), pp. 4699–4710 (Cited on pages 43, 45, 127, 144, 145, 148, 151, 177, 196).

[Con11] V. Conotter. "Active and passive multimedia forensics". PhD thesis. University of Trento, 2011 (Cited on pages 2, 161, 162).

[CS08] C. Chen and Y. Q. Shi. "JPEG image steganalysis utilizing both intrablock and interblock correlations". In: *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS)*. Seattle, Washington, USA, 2008, pp. 3029–3032 (Cited on pages 38, 39, 64, 86, 87, 175, 187).

[DN+13] D. T. Dang-Nguyen, I. D. Gebru, V. Conotter, G. Boato, and F. G. B. De Natale. "Counter-forensics of median filtering". In: *Proceedings of the IEEE International Workshop on Multimedia Signal Processing (MMSP)*. Pula, Sardinia, Italy, 2013, pp. 260–265 (Cited on pages xxiii, 24, 40, 44–46, 138, 141–144, 146, 147, 149, 150, 176, 177, 195, 197).

[Fan+12] W. Fan, K. Wang, F. Cayre, and Z. Xiong. "3-D lighting-based image forgery detection using shape-from-shading". In: *Proceedings of the European Signal Processing Conference (EUSIPCO)*. Bucharest, Romania: IEEE, 2012, pp. 1777–1781 (Cited on pages 158, 203).

[Fan+13a] W. Fan, K. Wang, F. Cayre, and Z. Xiong. "A variational approach to JPEG anti-forensics". In: *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*. Vancouver, Canada, 2013, pp. 3058–3062 (Cited on pages 7, 47, 113, 149).

[Fan+13b] W. Fan, K. Wang, F. Cayre, and Z. Xiong. "JPEG anti-forensics using non-parametric DCT quantization noise estimation and natural image statistics". In: *Proceedings of ACM International Workshop on Information Hiding and Multimedia Security (IHMMSec)*. Montpellier, France, 2013, pp. 117–122 (Cited on pages 7, 105, 113, 116, 118, 191).

[Fan+14] W. Fan, K. Wang, F. Cayre, and Z. Xiong. "JPEG anti-forensics with improved tradeoff between forensic undetectability and image quality". *IEEE Transactions on Information Forensics and Security* 9, 8 (2014), pp. 1211–1226 (Cited on pages 7, 66, 113).

[Fan+15] W. Fan, K. Wang, F. Cayre, and Z. Xiong. "Median filtered image quality enhancement and anti-forensics via variational deconvolution". *IEEE Transactions on Information Forensics and Security* (2015) (Cited on page 124).

[Far06] H. Farid. *Digital image ballistics from JPEG quantization*. Tech. rep. TR2006-583. Department of Computer Science, Dartmouth College, 2006 (Cited on pages 10, 12, 168).

[Far09a] H. Farid. "A survey of image forgery detection". *IEEE Signal Processing Magazine* 2, 26 (2009), pp. 16–25 (Cited on pages 2, 10, 14, 34, 158, 162, 165, 167, 169, 203).

[Far09b] H. Farid. "Exposing digital forgeries from JPEG ghosts". *IEEE Transactions on Information Forensics and Security* 1, 4 (2009), pp. 154–160 (Cited on page 37).

[FB12] M. Fontani and M. Barni. "Hiding traces of median filtering in digital images". In: *Proceedings of the European Signal Processing Conference (EUSIPCO)*. Bucharest, Romania: IEEE, 2012, pp. 1239–1243 (Cited on pages 24, 40, 44, 45, 138, 141, 176).

[FD00] Z. Fan and R. L. De Queiroz. "Maximum likelihood estimation of JPEG quantization table in the identification of bitmap compression history". In: *Proceedings of the IEEE International Conference on Image Processing (ICIP)*. Vancouver, Canada, 2000, pp. 948–951 (Cited on page 49).

[FD03] Z. Fan and R. L. De Queiroz. "Identification of bitmap compression history: JPEG detection and quantizer estimation". *IEEE Transactions on Image Processing* 12, 2 (2003), pp. 230–235 (Cited on pages 6, 23, 24, 36, 37, 39, 49–51, 53, 57, 60, 61, 64, 67, 69, 80, 116–118, 149, 150, 166, 174, 175).

[FGH02] J. Fridrich, M. Goljan, and D. Hogea. "Steganalysis of JPEG images: breaking the F5 algorithm". In: *Proceedings of the International Workshop on Information Hiding (IH)*. Noordwijkerhout, the Netherlands, 2002, pp. 310–323 (Cited on pages 7, 38, 106, 112, 116, 166, 189).

[Fou] *Fourandsix Technologies, Inc.* http://www.fourandsix.com/about-us/ (Cited on pages 2, 161).

[Fri93] G. L. Friedman. "The trustworthy digital camera: restoring credibility to the photographic image". *IEEE Transactions on Consumer Electronics* 39, 4 (1993), pp. 905–910 (Cited on pages 2, 161).

[GO09] T. Goldstein and S. Osher. "The split Bregman method for ℓ_1-regularized problems". *SIAM Journal on Imaging Sciences* 2, 2 (2009), pp. 323–343 (Cited on pages 28, 29, 131, 173, 193).

[GY95] D. Geman and C. Yang. "Nonlinear image recovery with half-quadratic regularization". *IEEE Transactions on Image Processing* 4, 7 (1995), pp. 932–946 (Cited on pages 27, 173).

[HF13] V. Holub and J. Fridrich. "Digital image steganography using universal distortion". In: *Proceedings of the ACM International Workshop on Information Hiding and Multimedia Security (IHMMSec)*. Montpellier, France, 2013, pp. 59–68 (Cited on page 19).

[Iee] *IEEE Xplore digital library*. `http://ieeexplore.ieee.org` (Cited on pages 3, 162).

[Ijg] *The independent JPEG group*. `http://www.ijg.org` (Cited on page 32).

[Kan+12] X. Kang, M. C. Stamm, A. Peng, and K. J. R. Liu. "Robust median filtering forensics based on the autoregressive model of median filtered residual". In: *Proceedings of the Asia-Pacific Signal Information Processing Association Annual Summit and Conference*. Hollywood, California, USA, 2012, pp. 1–9 (Cited on pages 43, 45, 127, 177).

[Kan+13] X. Kang, M. C. Stamm, A. Peng, and K. J. R. Liu. "Robust median filtering forensics using an autoregressive model". *IEEE Transactions on Information Forensics and Security* 8, 9 (2013), pp. 1456–1468 (Cited on pages 43, 45, 127, 144, 145, 148, 151, 177, 196, 197).

[KB07] M. Kirchner and R. Böhme. "Tamper hiding: defeating image forensics". In: *Proceedings of the International Conference on Information Hiding (IH)*. Saint Malo, France, 2007, pp. 326–341 (Cited on pages 3, 163).

[KF09] D. Krishnan and R. Fergus. "Fast image deconvolution using hyper-Laplacian priors". In: *Proceedings of the Neural Information Processing Systems (NIPS)*. Vancouver, Canada, 2009, pp. 1033–1041 (Cited on pages 7, 27, 125, 129–131, 166, 192, 193).

[KF10] M. Kirchner and J. Fridrich. "On detection of median filtering in digital images". In: *Proceedings of the SPIE: Media Forensics and Security II*. San Jose, California, USA, 2010, 754110:1–754110:12 (Cited on pages 41, 42, 44, 45, 127, 137, 138, 140, 142, 143, 146, 147, 150, 173, 177, 191, 195–197).

[KL51] S. Kullback and R. A. Leibler. "On information and sufficiency". *Annals of Mathematical Statistics* 22, 1 (1951), pp. 49–86 (Cited on pages 15, 22, 172).

[KR08] M. Kirchner and R. Röhme. "Hiding traces of resampling in digital images". *IEEE Transactions on Information Forensics and Security* 3, 4 (2008), pp. 582–592 (Cited on pages 4, 5, 121, 125, 146, 163, 165).

[KTF11] D. Krishnan, T. Tay, and R. Fergus. "Blind deconvolution using a normalized sparsity measure". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Colorado Springs, Colorado, USA, 2011, pp. 233–240 (Cited on pages 7, 29, 125, 131, 132, 166, 192, 193, 197).

[Kuh55] H. W. Kuhn. "The Hungarian method for the assignment problem". *Naval Research Logistics Quarterly* 2, 1-2 (1955), pp. 83–97 (Cited on pages 26, 77, 81, 173).

[LB11] S. Lai and R. Böhme. "Countering counter-forensics: the case of JPEG compression". In: *Proceedings of the International Conference on Information Hiding (IH)*. Prague, Czech Republic, 2011, pp. 285–298 (Cited on pages 37–39, 53, 56, 57, 60, 61, 80, 86, 112, 116–118, 175, 179, 189, 190).

[Lev+09] A. Levin, Y. Weiss, F. Durand, and W. T. Freeman. "Understanding and evaluating blind deconvolution algorithms". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Miami, Florida, USA, 2009, pp. 1964–1971 (Cited on page 125).

[LG00] E. Y. Lam and J. W. Goodman. "A mathematical analysis of the DCT coefficient distributions for images". *IEEE Transactions on Image Processing* 9, 10 (2000), pp. 1661–1666 (Cited on pages 36, 71).

[LHQ10] W. Luo, J. Huang, and G. Qiu. "JPEG error analysis and its applications to digital image forensics". *IEEE Transactions on Information Forensics and Security* 5, 3 (2010), pp. 480–491 (Cited on pages 36, 39, 53, 60, 64, 67, 69, 175).

[LLH12] H. Li, W. Luo, and J. Huang. "Countering anti-JPEG compression forensics". In: *Proceedings of the IEEE International Conference on Image Processing (ICIP)*. Orlando, Florida, USA, 2012, pp. 241–244 (Cited on pages 38, 39, 64, 86, 87, 175, 186, 187).

[LY04] A. W.-C. Liew and H. Yan. "Blocking artifacts suppression in block-coded images using overcomplete wavelet representation". *IEEE Transactions on Circuits and Systems for Video Technology* 14, 4 (2004), pp. 450–461 (Cited on pages 35, 48, 53).

[Mar] *Mary Meeker's state of the Internet*. `http://moonlighthk.com/mary-meekers-state-internet` (Cited on pages 1, 160).

[MB04] F. Melgani and L. Bruzzone. "Classification of hyperspectral remote sensing images with support vector machines". *IEEE Transactions on Geoscience and Remote Sensing* 42, 8 (2004), pp. 1778–1790 (Cited on page 19).

[PBF10] T. Pevný, P. Bas, and J. Fridrich. "Steganalysis by subtractive pixel adjacency matrix". *IEEE Transactions on Information Forensics and Security* 5, 2 (2010), pp. 215–224 (Cited on pages 18, 19, 38, 39, 42–45, 64, 86, 87, 127, 144, 145, 148, 151, 175, 177, 186, 187, 196).

[PF05] A.C. Popescu and H. Farid. "Exposing digital forgeries by detecting traces of resampling". *IEEE Transactions on Signal Processing* 53, 2 (2005), pp. 758–767 (Cited on pages 146, 147).

[PF08] T. Pevný and J. Fridrich. "Detection of double-compression in JPEG images for applications in steganography". *IEEE Transactions on Information Forensics and Security* 3, 2 (2008), pp. 247–258 (Cited on pages 89–91).

[Piv13] A. Piva. "An overview on image forensics". *ISRN Signal Processing* (2013), 22:1–22:22 (Cited on pages 11, 12, 14, 167).

[PK12] A. Peng and X. Kang. "Robust median filtering detection based on filtered residual". In: *Proceedings of the International Workshop on Digital Forensics and Watermarking (IWDW)*. Shanghai, China, 2012, pp. 344–357 (Cited on pages 43, 127).

[PM93] W. B. Pennebaker and J. L. Mitchell. *JPEG still image data compression standard*. New York: Van Nostrand Reinhold, 1993 (Cited on page 32).

[PPK03] S. C. Park, M. K. Park, and M. G. Kang. "Super-resolution image reconstruction: a technical overview". *IEEE Signal Processing Magazine* 20, 3 (2003), pp. 21–36 (Cited on pages 157, 201).

[PR99] J. R. Price and M. Rabbani. "Biased reconstruction for JPEG decoding". *IEEE Signal Processing Letters* 6, 12 (1999), pp. 297–299 (Cited on pages 71, 76, 111).

[PV92] I. Pitas and A. N. Venetsanopoulos. "Order statistics in digital image processing". *Proceedings of IEEE* 80, 12 (1992), pp. 1893–1921 (Cited on page 40).

[ROF92] L. I. Rudin, S. Osher, and E. Fatemi. "Nonlinear total variation based noise removal algorithms". *Physica D: Nonlinear Phenomena* 60, 1-4 (1992), pp. 259–268 (Cited on pages 37, 48, 178).

[RS05] M. A. Robertson and R. L. Stevenson. "DCT quantization noise in compressed images". *IEEE Transactions on Circuits and Systems for Video Technology* 15, 1 (2005), pp. 27–38 (Cited on pages 6, 7, 34–36, 57, 72, 73, 97, 106–109, 166, 174, 183).

[RTD11] J. A. Redi, W. Taktak, and J.-L. Dugelay. "Digital image forensics: a booklet for beginners". *Multimedia Tools and Applications* 51, 1 (2011), pp. 133–162 (Cited on pages 10, 11, 14, 167).

[Sal03] P. Sallee. *Matlab JPEG toolbox*. Available at `http://dde.binghamton.edu/download/feature_extractors/`. 2003 (Cited on pages 33, 55).

[SC07] D. Sun and W.-K. Cham. "Postprocessing of low bit-rate block DCT coded images based on a fields of experts prior". *IEEE Transactions on Image Processing* 16, 11 (2007), pp. 2743–2751 (Cited on pages 35, 57, 97, 106–110, 188).

[SL11] M. C. Stamm and K. J. R. Liu. "Anti-forensics of digital image compression". *IEEE Transactions on Information Forensics and Security* 6, 3 (2011), pp. 1050–1065 (Cited on pages xxii, 4–6, 36–40, 46, 48, 49, 52, 53, 58, 60, 63, 64, 67, 71, 76, 81–90, 92, 96, 106, 110–113, 116, 118, 119, 121, 125, 146, 149, 155, 163, 165, 175, 177, 180, 181, 185, 200).

[SS04] G. Schaefer and M. Stich. "UCID - an uncompressed colour image database". In: *Proceedings of the SPIE: Storage and Retrieval Methods and Applications for Multimedia*. San Jose, California, USA, 2004, pp. 472–480 (Cited on pages 22–24, 50, 51, 56, 60, 71, 72, 89, 172, 183, 184).

[SS11] P. Sutthiwan and Y. Q. Shi. "Anti-forensics of double JPEG compression detection". In: *Proceedings of the International Workshop on Digital Forensics and Watermarking (IWDW)*. Atlantic City, New Jersey, USA, 2011, pp. 411–424 (Cited on pages xxii, 22, 38–40, 52, 53, 58, 60, 62, 89–91, 93, 118, 121, 172, 175, 180, 181).

[Sta+10a] M. Stamm, S. Tjoa, W. S. Lin, and K. J. R. Liu. "Anti-forensics of JPEG compression". In: *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*. Dallas, Texas, USA, 2010, pp. 1694–1697 (Cited on pages xxii, 22, 36–40, 48, 49, 52, 53, 58, 60, 63, 64, 67, 71, 74, 76, 81–92, 96, 106, 110–113, 116, 118, 119, 121, 125, 149, 155, 172, 175, 180, 181, 185, 200).

[Sta+10b] M. Stamm, S. Tjoa, W. S. Lin, and K. J. R. Liu. "Undetectable image tampering through JPEG compression anti-forensics". In: *Proceedings of the IEEE International Conference on Image Processing (ICIP)*. Hongkong, China, 2010, pp. 2109–2112 (Cited on pages xxii, 22, 37–40, 46, 48, 52, 53, 58, 60, 63, 64, 67, 85–92, 96, 116, 118, 119, 121, 125, 149, 155, 172, 175, 177, 180, 181, 200).

[SWL13] M. C. Stamm, M. Wu, and K. J. R. Liu. "Information forensics: an overview of the first decade". *IEEE Access* 1, (2013), pp. 167–200 (Cited on pages 12–14, 167).

[TT10] X. Tan and B. Triggs. "Enhanced local texture feature sets for face recognition under difficult lighting conditions". *IEEE Transactions on Image Processing* 19, 6 (2010), pp. 1635–1650 (Cited on page 43).

[UW08] C. Ullerich and A. Westfeld. "Weaknesses of MB2". In: *Proceedings of the International Workshop on Digital Watermarking (IWDW)*. Guangzhou, China, 2008, pp. 127–142 (Cited on page 51).

[Val+11] G. Valenzise, V. Nobile, M. Tagliasacchi, and S. Tubaro. "Countering JPEG anti-forensics". In: *Proceedings of the IEEE International Conference on Image Processing (ICIP)*. Brussels, Belguim, 2011, pp. 1949–1952 (Cited on pages 37, 39, 53, 60, 86, 116, 118, 175).

[Vap98] V. N. Vapnik. *The nature of statistical learning theory*. Second. Springer, 1998 (Cited on page 19).

[VPPG11] D. Vázquez-Padín and F. Pérez-González. "Prefilter design for forensic resampling estimation". In: *Proceedings of the IEEE International Workshop on Information Forensics and Security (WIFS)*. Iguacu Falls, Brazil, 2011, pp. 1–6 (Cited on page 24).

[VPPG12] D. Vázquez-Padín and F. Pérez-González. "ML estimation of the resampling factor". In: *Proceedings of the IEEE International Workshop on Information Forensics and Security (WIFS)*. Tenerife, Spain, 2012, pp. 205–210 (Cited on page 24).

[VTT11] G. Valenzise, M. Tagliasacchi, and S. Tubaro. "The cost of JPEG compression anti-forensics". In: *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*. Prague, Czech Republic, 2011, pp. 1884–1887 (Cited on pages xxii, 22, 37, 39, 40, 52, 53, 58, 60, 62, 74, 81–83, 86, 89–91, 111, 118, 121, 172, 175, 180, 181).

[VTT13] G. Valenzise, M. Tagliasacchi, and S. Tubaro. "Revealing the traces of JPEG compression anti-forensics". *IEEE Transactions on Information Forensics and Security* 8, 2 (2013), pp. 335–349 (Cited on pages 37–39, 53, 60, 86, 118, 175).

[W3t] http://w3techs.com/technologies/overview/image_format/all (Cited on pages 4, 164).

[Wan+04] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. "Image quality assessment: from error visibility to structural similarity". *IEEE Transactions on Image Processing* 13, 4 (2004), pp. 600–612 (Cited on pages 21, 22, 77, 133, 171).

[WB06] Z. Wang and A. C. Bovik. *Modern image quality assessment*. Morgan & Claypool, 2006 (Cited on pages 21, 171).

[WN09] Z. Wei and K. N. Ngan. "Spatio-temporal just noticeable distortion profile for grey scale image/video in DCT domain". *IEEE Transactions on Circuits and Systems for Video Technology* 19, 3 (2009), pp. 337–346 (Cited on page 37).

[WSL13] Z.-H. Wu, M. C. Stamm, and K. J. R. Liu. "Anti-forensics of median filtering". In: *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*. Vancouver, Canada, 2013, pp. 3043–3047 (Cited on pages xxiii, 4, 6, 24, 25, 40, 44, 45, 128, 138, 141–144, 146, 147, 149, 150, 163, 165, 176, 195, 197).

[YGK95] Y. Yang, N. Galatsanos, and A. Katsaggelos. "Projection-based spatially adaptive reconstruction of block-transform compressed images". *IEEE Transactions on Image Processing* 4, 7 (1995), pp. 896–908 (Cited on pages 35, 57).

[Yua11] H.-D. Yuan. "Blind forensics of median filtering in digital images". *IEEE Transactions on Information Forensics and Security* 6, 4 (2011), pp. 1335–1345 (Cited on pages 42, 44, 45, 127, 132, 137, 138, 140–148, 150, 151, 173, 177, 191, 193, 195–197).

[Zha+08] G. Zhai, W. Zhang, X. Yang, W. Lin, and Y. Xu. "Efficient image deblocking based on postfiltering in shifted windows". *IEEE Transactions on Circuits and Systems for Video Technology* 18, 1 (2008), pp. 122–126 (Cited on pages 35, 48, 53).

[Zha+10] B. Zhang, Y. Gao, S. Zhao, and J. Liu. "Local derivative pattern versus local binary pattern: face recognition with high-order local pattern descriptor". *IEEE Transactions on Image Processing* 19, 2 (2010), pp. 533–544 (Cited on page 44).

[Zha+14] Y. Zhang, S. Li, S. Wang, and Y. Q. Shi. "Revealing the traces of median filtering using high-order local ternary patterns". *IEEE Signal Processing Letters* 21, 3 (2014), pp. 275–280 (Cited on pages 43, 45, 127, 144, 145, 148, 151, 177, 196).

[ZL14] X. Zhang and S. Lyu. "Blind estimation of pixel brightness transform". In: *Proceedings of the IEEE International Conference on Image Processing (ICIP)*. Paris, France, 2014, pp. 4472–4476 (Cited on pages 157, 201).

[ZW11] D. Zoran and Y. Weiss. "From learning models of natural image patches to whole image restoration". In: *Proceedings of the International Conference on Computer Vision (ICCV)*. Barcelona, Spain, 2011, pp. 479–486 (Cited on pages 7, 27, 29, 36, 97, 106, 108, 109, 116, 117, 131, 166, 188, 189).

Author's Publications

International Journals

- **Wei Fan**, Kai Wang, François Cayre, and Zhang Xiong, "JPEG anti-forensics with improved tradeoff between forensic undetectability and image quality", *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 8, pp. 1211-1226, 2014.
- **Wei Fan**, Kai Wang, François Cayre, and Zhang Xiong, "Median filtered image quality enhancement and anti-forensics via variational deconvolution", *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 5, pp. 1076-1091, 2015.

International Conferences

- **Wei Fan**, Kai Wang, François Cayre, and Zhang Xiong, "3-D lighting-based image forgery detection using shape-from-shading", In: *Proceedings of the European Signal Processing Conference (EUSIPCO)*, Bucharest, Romania: IEEE, 2012, pp. 1777-1781.
- **Wei Fan**, Kai Wang, François Cayre, and Zhang Xiong, "A variational approach to JPEG anti-forensics", In: *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Vancouver, Canada, 2013, pp. 3058-3062.
- **Wei Fan**, Kai Wang, François Cayre, and Zhang Xiong, "JPEG anti-forensics using non-parametric DCT quantization noise estimation and natural image statistics", In: *Proceedings of the ACM International Workshop on Information Hiding and Multimedia Security (IHMMSec)*, Montpellier, France, 2013, pp. 117-122. (**Best Paper Award**)
- Meijuan Wang, Zhenyong Chen, **Wei Fan**, and Zhang Xiong, "Countering anti-forensics to wavelet-based compression", In: *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, Paris, France, 2014, pp. 5382-5386.

Local Conference

- **Wei Fan**, Kai Wang, François Cayre, and Zhang Xiong, "Empêcher la détection criminalistique de filtrage médian en images numériques", In: *Proceedings of the Colloque Gretsi - Communauté Francophone du Traitement du Signal et des Images*, Lyon, France, 2015. (in French)

Title: Towards Digital Image Anti-Forensics via Image Restoration

Abstract: Image forensics enjoys its increasing popularity as a powerful image authentication tool, working in a blind passive way without the aid of any *a priori* embedded information compared to fragile image watermarking. On its opponent side, image anti-forensics attacks forensic algorithms for the future development of more trustworthy forensics. When image coding or processing is involved, we notice that image anti-forensics to some extent shares a similar goal with image restoration. Both of them aim to recover the information lost during the image degradation, yet image anti-forensics has one additional indispensable forensic undetectability requirement. In this thesis, we form a new research line for image anti-forensics, by leveraging on advanced concepts/methods from image restoration meanwhile with integrations of anti-forensic strategies/terms. Under this context, this thesis contributes on the following four aspects for JPEG compression and median filtering anti-forensics: (i) JPEG anti-forensics using Total Variation based deblocking, (ii) improved Total Variation based JPEG anti-forensics with assignment problem based perceptual DCT histogram smoothing, (iii) JPEG anti-forensics using JPEG image quality enhancement based on a sophisticated image prior model and non-parametric DCT histogram smoothing based on calibration, and (iv) median filtered image quality enhancement and anti-forensics via variational deconvolution. Experimental results demonstrate the effectiveness of the proposed anti-forensic methods with a better forensic undetectability against existing forensic detectors as well as a higher visual quality of the processed image, by comparisons with the state-of-the-art methods.

Keywords: Anti-forensics, digital image, image restoration, JPEG compression, median filtering

Titre : Vers l'anti-criminalistique en images numériques via la restauration d'images

Résumé : La criminalistique en images numériques se développe comme un outil puissant pour l'authentification d'image, en travaillant de manière passive et aveugle sans l'aide d'informations d'authentification pré-intégrées dans l'image (contrairement au tatouage fragile d'image). En parallèle, l'anti-criminalistique se propose d'attaquer les algorithmes de criminalistique afin de maintenir une saine émulation susceptible d'aider à leur amélioration. En images numériques, l'anti-criminalistique partage quelques similitudes avec la restauration d'image : dans les deux cas, l'on souhaite approcher au mieux les informations perdues pendant un processus de dégradation d'image. Cependant, l'anti-criminalistique se doit de remplir au mieux un objectif supplémentaire, *i.e.* : être non détectable par la criminalistique actuelle. Dans cette thèse, nous proposons une nouvelle piste de recherche pour la criminalistique en images numériques, en tirant profit des concepts/méthodes avancés de la restauration d'image mais en intégrant des stratégies/termes spécifiquement anti-criminalistiques. Dans ce contexte, cette thèse apporte des contributions sur quatre aspects concernant, en criminalistique JPEG, (i) l'introduction du déblocage basé sur la variation totale pour contrer les méthodes de criminalistique JPEG et (ii) l'amélioration apportée par l'adjonction d'un lissage perceptuel de l'histogramme DCT, (iii) l'utilisation d'un modèle d'image sophistiqué et d'un lissage non paramétrique de l'histogramme DCT visant l'amélioration de la qualité de l'image falsifiée; et, en criminalistique du filtrage médian, (iv) l'introduction d'une méthode fondée sur la déconvolution variationnelle. Les résultats expérimentaux démontrent l'efficacité des méthodes anti-criminalistiques proposées, avec notamment une meilleure indétectabilité face aux détecteurs criminalistiques actuels ainsi qu'une meilleure qualité visuelle de l'image falsifiée par rapport aux méthodes anti-criminalistiques de l'état de l'art.

Mots clés : Anti-criminalistique, image numérique, restauration d'image, compression JPEG, filtrage médian

GIPSA-lab, 11 rue des Mathématiques, Grenoble Campus BP 46,
F-38402 Saint Martin d'Hères CEDEX, France