Ingemar J. Cox
Matthew L. Miller
Jeffrey A. Bloom
Jessica Fridrich
Ton Kalker

Digital Watermarking *and* Steganography

MORGAN KAUFMANN

SECOND EDITION

Digital Watermarking and Steganography

The Morgan Kaufmann Series in Multimedia Information and Systems

Series Editor, Edward A. Fox, Virginia Poytechnic University

Digital Watermarking and Steganography, Second Edition
Ingemar J. Cox, Matthew L. Miller, Jeffrey A. Bloom, Jessica Fridrich, and Ton Kalker

Keeping Found Things Found: The Study and Practice of Personal Information Management
William P. Jones

Web Dragons: Inside the Myths of Search Engine Technology
Ian H. Witten, Marco Gori, and Teresa Numerico

Introduction to Data Compression, Third Edition
Khalid Sayood

Understanding Digital Libraries, Second Edition
Michael Lesk

Bioinformatics: Managing Scientific Data
Zoé Lacroix and Terence Critchlow

How to Build a Digital Library
Ian H. Witten and David Bainbridge

Readings in Multimedia Computing and Networking
Kevin Jeffay and Hong Jiang Zhang

Multimedia Servers: Applications, Environments, and Design
Dinkar Sitaram and Asit Dan

Visual Information Retrieval
Alberto del Bimbo

Managing Gigabytes: Compressing and Indexing Documents and Images, Second Edition
Ian H. Witten, Alistair Moffat, and Timothy C. Bell

Digital Compression for Multimedia: Principles & Standards
Jerry D. Gibson, Toby Berger, Tom Lookabaugh, Rich Baker, and David Lindbergh

Readings in Information Retrieval
Karen Sparck Jones, and Peter Willett

For further information on these books and for a list of forthcoming titles,
please visit our web site at http://www.mkp.com.

The Morgan Kaufmann Series in Computer Security

Digital Watermarking and Steganography, Second Edition
Ingemar J. Cox, Matthew L. Miller, Jeffrey A. Bloom, Jessica Fridrich, and Ton Kalker

Information Assurance: Dependability and Security in Networked Systems
Yi Qian, David Tipper, Prashant Krishnamurthy, and James Joshi

Network Recovery: Protection and Restoration of Optical, SONET-SDH, IP, and MPLS
Jean-Philippe Vasseur, Mario Pickavet, and Piet Demeester

For further information on these books and for a list of forthcoming titles,
please visit our Web site at http://www.mkp.com.

Digital Watermarking and Steganography

Second Edition

Ingemar J. Cox

Matthew L. Miller

Jeffrey A. Bloom

Jessica Fridrich

Ton Kalker

ELSEVIER

AMSTERDAM • BOSTON • HEIDELBERG • LONDON
NEW YORK • OXFORD • PARIS • SAN DIEGO
SAN FRANCISCO • SINGAPORE • SYDNEY • TOKYO

Morgan Kaufmann Publishers is an imprint of Elsevier

MORGAN KAUFMANN PUBLISHERS

Publishing Director Denise E. M. Penrose
Senior Acquisitions Editor Rick Adams
Publishing Services Manager George Morrison
Senior Project Manager Brandy Lilly
Editorial Assistant Gregory Chalson
Cover Design Dennis Schaefer
Text Design Elsevier, Inc.
Composition diacriTech
Copyeditor Janet Cocker
Proofreader Jodie Allen
Indexer Distributech Scientific Indexing
Interior printer The Maple-Vail Book Manufacturing Group
Cover printer Phoenix Color

Morgan Kaufmann Publishers is an imprint of Elsevier.
30 Corporate Drive, Suite 400, Burlington, MA 01803, USA

This book is printed on acid-free paper. ♾

Library of Congress Cataloging-in-Publication Data
Digital watermarking and steganography/Ingemar J. Cox ... [et al.].
p. cm.
Includes bibliographical references and index.
ISBN 978-0-12-372585-1 (casebound: alk. paper) 1. Computer security. 2. Digital watermarking. 3. Data protection. I. Cox, I. J. (Ingemar J.)
QA76.9.A25C68 2008
005.8–dc22
2007040595

ISBN 978-0-12-372585-1

For information on all Morgan Kaufmann publications,
visit our Web site at *www.mkp.com* or *www.books.elsevier.com*

Printed in the United States of America
07 08 09 10 11 5 4 3 2 1

This book is dedicated to the memory of

Ingy Cox

Age 12

May 23, 1986 to January 27, 1999

The light that burns twice as bright burns half as long—and you have burned so very very brightly.

—Eldon Tyrell to Roy Batty in *Blade Runner*.
Screenplay by Hampton Fancher and David Peoples.

This page intentionally left blank

Contents

Preface to the First Edition xv
Preface to the Second Edition xix
Example Watermarking Systems xxi

CHAPTER 1 Introduction **1**
1.1 Information Hiding, Steganography, and Watermarking 4
1.2 History of Watermarking 6
1.3 History of Steganography 9
1.4 Importance of Digital Watermarking 11
1.5 Importance of Steganography 12

CHAPTER 2 Applications and Properties **15**
2.1 Applications of Watermarking 16
2.1.1 Broadcast Monitoring 16
2.1.2 Owner Identification 19
2.1.3 Proof of Ownership 21
2.1.4 Transaction Tracking 23
2.1.5 Content Authentication 25
2.1.6 Copy Control 27
2.1.7 Device Control 31
2.1.8 Legacy Enhancement 32
2.2 Applications of Steganography 34
2.2.1 Steganography for Dissidents 34
2.2.2 Steganography for Criminals 35
2.3 Properties of Watermarking Systems 36
2.3.1 Embedding Effectiveness 37
2.3.2 Fidelity 37
2.3.3 Data Payload 38
2.3.4 Blind or Informed Detection 39
2.3.5 False Positive Rate 39
2.3.6 Robustness 40
2.3.7 Security 41
2.3.8 Cipher and Watermark Keys 43
2.3.9 Modification and Multiple Watermarks 45
2.3.10 Cost 46
2.4 Evaluating Watermarking Systems 46
2.4.1 The Notion of "Best" 47
2.4.2 Benchmarking 47
2.4.3 Scope of Testing 48

2.5 Properties of Steganographic and Steganalysis Systems 49
 2.5.1 Embedding Effectiveness 49
 2.5.2 Fidelity 50
 2.5.3 Steganographic Capacity, Embedding Capacity, Embedding Efficiency, and Data Payload 50
 2.5.4 Blind or Informed Extraction 51
 2.5.5 Blind or Targeted Steganalysis 51
 2.5.6 Statistical Undetectability 52
 2.5.7 False Alarm Rate 53
 2.5.8 Robustness 53
 2.5.9 Security 54
 2.5.10 Stego Key 54
2.6 Evaluating and Testing Steganographic Systems 55
2.7 Summary 56

CHAPTER 3 Models of Watermarking **61**
3.1 Notation 62
3.2 Communications 63
 3.2.1 Components of Communications Systems 63
 3.2.2 Classes of Transmission Channels 64
 3.2.3 Secure Transmission 65
3.3 Communication-Based Models of Watermarking 67
 3.3.1 Basic Model 67
 3.3.2 Watermarking as Communications with Side Information at the Transmitter 75
 3.3.3 Watermarking as Multiplexed Communications 78
3.4 Geometric Models of Watermarking 80
 3.4.1 Distributions and Regions in Media Space 81
 3.4.2 Marking Spaces 87
3.5 Modeling Watermark Detection by Correlation 95
 3.5.1 Linear Correlation 96
 3.5.2 Normalized Correlation 97
 3.5.3 Correlation Coefficient 100
3.6 Summary 102

CHAPTER 4 Basic Message Coding **105**
4.1 Mapping Messages into Message Vectors 106
 4.1.1 Direct Message Coding 106
 4.1.2 Multisymbol Message Coding 110
4.2 Error Correction Coding 117
 4.2.1 The Problem with Simple Multisymbol Messages 117
 4.2.2 The Idea of Error Correction Codes 118
 4.2.3 Example: Trellis Codes and Viterbi Decoding 119

4.3 Detecting Multisymbol Watermarks 124
4.3.1 Detection by Looking for Valid Messages 125
4.3.2 Detection by Detecting Individual Symbols 126
4.3.3 Detection by Comparing against Quantized Vectors . . . 128
4.4 Summary 134

CHAPTER 5 Watermarking with Side Information **137**
5.1 Informed Embedding 139
5.1.1 Embedding as an Optimization Problem 140
5.1.2 Optimizing with Respect to a Detection Statistic 141
5.1.3 Optimizing with Respect to an Estimate of Robustness 147
5.2 Watermarking Using Side Information 153
5.2.1 Formal Definition of the Problem 153
5.2.2 Signal and Channel Models 155
5.2.3 Optimal Watermarking for a Single Cover Work 156
5.2.4 Optimal Coding for Multiple Cover Works 157
5.2.5 A Geometrical Interpretation of White Gaussian Signals 158
5.2.6 Understanding Shannon's Theorem 159
5.2.7 Correlated Gaussian Signals 161
5.3 Dirty-Paper Codes 164
5.3.1 Watermarking of Gaussian Signals: First Approach 164
5.3.2 Costa's Insight: Writing on Dirty Paper 170
5.3.3 Scalar Watermarking 175
5.3.4 Lattice Codes 179
5.4 Summary 181

CHAPTER 6 Practical Dirty-Paper Codes **183**
6.1 Practical Considerations for Dirty-Paper Codes 183
6.1.1 Efficient Encoding Algorithms 184
6.1.2 Efficient Decoding Algorithms 185
6.1.3 Tradeoff between Robustness and Encoding Cost 186
6.2 Broad Approaches to Dirty-Paper Code Design 188
6.2.1 Direct Binning 188
6.2.2 Quantization Index Modulation 188
6.2.3 Dither Modulation 189
6.3 Implementing DM with a Simple Lattice Code 189
6.4 Typical Tricks in Implementing Lattice Codes 194
6.4.1 Choice of Lattice 194
6.4.2 Distortion Compensation 194
6.4.3 Spreading Functions 195
6.4.4 Dither 195

6.5 Coding with Better Lattices 197
6.5.1 Using Nonorthogonal Lattices 197
6.5.2 Important Properties of Lattices 199
6.5.3 Constructing a Dirty-Paper Code from E_8 201
6.6 Making Lattice Codes Survive Valumetric Scaling 204
6.6.1 Scale-Invariant Marking Spaces 205
6.6.2 Rational Dither Modulation 207
6.6.3 Inverting Valumetric Scaling 208
6.7 Dirty-Paper Trellis Codes 208
6.8 Summary 212

CHAPTER 7 Analyzing Errors 213
7.1 Message Errors 214
7.2 False Positive Errors 218
7.2.1 Random-Watermark False Positive 219
7.2.2 Random-Work False Positive 221
7.3 False Negative Errors 225
7.4 ROC Curves 228
7.4.1 Hypothetical ROC 228
7.4.2 Histogram of a Real System 230
7.4.3 Interpolation Along One or Both Axes 231
7.5 The Effect of Whitening on Error Rates 232
7.6 Analysis of Normalized Correlation 239
7.6.1 False Positive Analysis 240
7.6.2 False Negative Analysis 250
7.7 Summary 252

CHAPTER 8 Using Perceptual Models 255
8.1 Evaluating Perceptual Impact of Watermarks 255
8.1.1 Fidelity and Quality 256
8.1.2 Human Evaluation Measurement Techniques 257
8.1.3 Automated Evaluation 260
8.2 General Form of a Perceptual Model 263
8.2.1 Sensitivity 263
8.2.2 Masking 266
8.2.3 Pooling 267
8.3 Two Examples of Perceptual Models 269
8.3.1 Watson's DCT-Based Visual Model 269
8.3.2 A Perceptual Model for Audio 273
8.4 Perceptually Adaptive Watermarking 277
8.4.1 Perceptual Shaping 280
8.4.2 Optimal Use of Perceptual Models 287
8.5 Summary 295

CHAPTER 9 Robust Watermarking 297

9.1 Approaches . . . 298
 9.1.1 Redundant Embedding . . . 299
 9.1.2 Spread Spectrum Coding . . . 300
 9.1.3 Embedding in Perceptually Significant Coefficients . . . 301
 9.1.4 Embedding in Coefficients of Known Robustness . . . 302
 9.1.5 Inverting Distortions in the Detector . . . 303
 9.1.6 Preinverting Distortions in the Embedder . . . 304
9.2 Robustness to Valumetric Distortions . . . 308
 9.2.1 Additive Noise . . . 308
 9.2.2 Amplitude Changes . . . 312
 9.2.3 Linear Filtering . . . 314
 9.2.4 Lossy Compression . . . 319
 9.2.5 Quantization . . . 320
9.3 Robustness to Temporal and Geometric Distortions . . . 325
 9.3.1 Temporal and Geometric Distortions . . . 326
 9.3.2 Exhaustive Search . . . 327
 9.3.3 Synchronization/Registration in Blind Detectors . . . 328
 9.3.4 Autocorrelation . . . 329
 9.3.5 Invariant Watermarks . . . 330
 9.3.6 Implicit Synchronization . . . 331
9.4 Summary . . . 332

CHAPTER 10 Watermark Security 335

10.1 Security Requirements . . . 335
 10.1.1 Restricting Watermark Operations . . . 336
 10.1.2 Public and Private Watermarking . . . 338
 10.1.3 Categories of Attack . . . 340
 10.1.4 Assumptions about the Adversary . . . 345
10.2 Watermark Security and Cryptography . . . 348
 10.2.1 The Analogy between Watermarking and Cryptography . . . 348
 10.2.2 Preventing Unauthorized Detection . . . 349
 10.2.3 Preventing Unauthorized Embedding . . . 351
 10.2.4 Preventing Unauthorized Removal . . . 355
10.3 Some Significant Known Attacks . . . 358
 10.3.1 Scrambling Attacks . . . 359
 10.3.2 Pathological Distortions . . . 359
 10.3.3 Copy Attacks . . . 361
 10.3.4 Ambiguity Attacks . . . 362
 10.3.5 Sensitivity Analysis Attacks . . . 367
 10.3.6 Gradient Descent Attacks . . . 372
10.4 Summary . . . 373

CHAPTER 11 Content Authentication 375

11.1 Exact Authentication . . . 377
11.1.1 Fragile Watermarks . . . 377
11.1.2 Embedded Signatures . . . 378
11.1.3 Erasable Watermarks . . . 379
11.2 Selective Authentication . . . 395
11.2.1 Legitimate versus Illegitimate Distortions . . . 395
11.2.2 Semi-Fragile Watermarks . . . 399
11.2.3 Embedded, Semi-Fragile Signatures . . . 404
11.2.4 Telltale Watermarks . . . 409
11.3 Localization . . . 410
11.3.1 Block-Wise Content Authentication . . . 411
11.3.2 Sample-Wise Content Authentication . . . 412
11.3.3 Security Risks with Localization . . . 415
11.4 Restoration . . . 419
11.4.1 Embedded Redundancy . . . 419
11.4.2 Self-Embedding . . . 420
11.4.3 Blind Restoration . . . 421
11.5 Summary . . . 422

CHAPTER 12 Steganography 425

12.1 Steganographic Communication . . . 427
12.1.1 The Channel . . . 428
12.1.2 The Building Blocks . . . 429
12.2 Notation and Terminology . . . 433
12.3 Information-Theoretic Foundations of Steganography . . . 433
12.3.1 Cachin's Definition of Steganographic Security . . . 434
12.4 Practical Steganographic Methods . . . 439
12.4.1 Statistics Preserving Steganography . . . 439
12.4.2 Model-Based Steganography . . . 441
12.4.3 Masking Embedding as Natural Processing . . . 445
12.5 Minimizing the Embedding Impact . . . 449
12.5.1 Matrix Embedding . . . 450
12.5.2 Nonshared Selection Rule . . . 457
12.6 Summary . . . 467

CHAPTER 13 Steganalysis 469

13.1 Steganalysis Scenarios . . . 469
13.1.1 Detection . . . 470
13.1.2 Forensic Steganalysis . . . 475
13.1.3 The Influence of the Cover Work on Steganalysis . . . 476
13.2 Some Significant Steganalysis Algorithms . . . 477
13.2.1 LSB Embedding and the Histogram Attack . . . 478

13.2.2 Sample Pairs Analysis 480
13.2.3 Blind Steganalysis of JPEG Images Using Calibration . . . 486
13.2.4 Blind Steganalysis in the Spatial Domain 489
13.3 Summary 494

APPENDIX A Background Concepts **497**
A.1 Information Theory 497
A.1.1 Entropy 497
A.1.2 Mutual Information 498
A.1.3 Communication Rates 499
A.1.4 Channel Capacity 500
A.2 Coding Theory 503
A.2.1 Hamming Distance 503
A.2.2 Covering Radius 503
A.2.3 Linear Codes 504
A.3 Cryptography 505
A.3.1 Symmetric-Key Cryptography 505
A.3.2 Asymmetric-Key Cryptography 506
A.3.3 One-Way Hash Functions 508
A.3.4 Cryptographic Signatures 510

APPENDIX B Selected Theoretical Results **511**
B.1 Information-Theoretic Analysis of Secure Watermarking (Moulin and O'Sullivan) 511
B.1.1 Watermarking as a Game 511
B.1.2 General Capacity of Watermarking 513
B.1.3 Capacity with MSE Fidelity Constraint 514
B.2 Error Probabilities Using Normalized Correlation Detectors (Miller and Bloom) 517
B.3 Effect of Quantization Noise on Watermarks (Eggers and Girod) . 522
B.3.1 Background 524
B.3.2 Basic Approach 524
B.3.3 Finding the Probability Density Function 524
B.3.4 Finding the Moment-Generating Function 525
B.3.5 Determining the Expected Correlation for a Gaussian Watermark and Laplacian Content 527

APPENDIX C Notation and Common Variables **529**
C.1 Variable Naming Conventions 529
C.2 Operators 530
C.3 Common Variable Names 530
C.4 Common Functions 532

Glossary 533

References 549

Index 575

About the Authors 591

Preface to the First Edition

Watermarking, as we define it, is the practice of hiding a message about an image, audio clip, video clip, or other work of media within that work itself. Although such practices have existed for quite a long time—at least several centuries, if not millennia—the field of *digital* watermarking only gained widespread popularity as a research topic in the latter half of the 1990s. A few earlier books have devoted substantial space to the subject of digital watermarking [171, 207, 219]. However, to our knowledge, this is the first book dealing exclusively with this field.

PURPOSE

Our goal with this book is to provide a framework in which to conduct research and development of watermarking technology. This book is not intended as a comprehensive survey of the field of watermarking. Rather, it represents our own point of view on the subject. Although we analyze specific examples from the literature, we do so only to the extent that they highlight particular concepts being discussed. (Thus, omissions from the Bibliography should not be considered as reflections on the quality of the omitted works.)

Most of the literature on digital watermarking deals with its application to images, audio, and video, and these application areas have developed somewhat independently. This is in part because each medium has unique characteristics, and researchers seldom have expertise in all three. We are no exception, our own backgrounds being predominantly in images and video. Nevertheless, the fundamental principles behind still image, audio, and video watermarking are the same, so we have made an effort to keep our discussion of these principles generic.

The principles of watermarking we discuss are illustrated with several example algorithms and experiments (the C source code is provided in Appendix C). All of these examples are implemented for image watermarking only. We decided to use only image-based examples because, unlike audio or video, images can be easily presented in a book.

The example algorithms are very simple. In general, they are not themselves useful for real watermarking applications. Rather, each algorithm is intended to provide a clear illustration of a specific idea, and the experiments are intended to examine the idea's effect on performance.

The book contains a certain amount of repetition. This was a conscious decision, because we assume that many, if not most, readers will not read the book from cover to cover. Rather, we anticipate that readers will look up topics of interest and read only individual sections or chapters. Thus, if a point is relevant in a number of places, we may briefly repeat it several times. It is hoped that this will not make the book too tedious to read straight through, yet will make it more useful to those who read technical books the way we do.

CONTENT AND ORGANIZATION

Chapters 1 and 2 of this book provide introductory material. Chapter 1 provides a history of watermarking, as well as a discussion of the characteristics that distinguish watermarking from the related fields of data hiding and steganography. Chapter 2 describes a wide variety of applications of digital watermarking and serves as motivation. The applications highlight a variety of sometimes conflicting requirements for watermarking, which are discussed in more detail in the second half of the chapter.

The technical content of this book begins with Chapter 3, which presents several frameworks for modeling watermarking systems. Along the way, we describe, test, and analyze some simple image watermarking algorithms that illustrate the concepts being discussed. In Chapter 4, these algorithms are extended to carry larger data payloads by means of conventional message-coding techniques. Although these techniques are commonly used in watermarking systems, some recent research suggests that substantially better performance can be achieved by exploiting side information in the encoding process. This is discussed in Chapter 5.

Chapter 7 analyzes message errors, false positives, and false negatives that may occur in watermarking systems. It also introduces whitening.

The next three chapters explore a number of general problems related to fidelity, robustness, and security that arise in designing watermarking systems, and present techniques that can be used to overcome them. Chapter 8 examines the problems of modeling human perception, and of using those models in watermarking systems. Although simple perceptual models for audio and still images are described, perceptual modeling is not the focus of this chapter. Rather, we focus on how any perceptual model can be used to improve the fidelity of the watermarked content.

Chapter 9 covers techniques for making watermarks survive several types of common degradations, such as filtering, geometric or temporal transformations, and lossy compression.

Chapter 10 describes a framework for analyzing security issues in watermarking systems. It then presents a few types of malicious attacks to which watermarks might be subjected, along with possible countermeasures.

Finally, Chapter 11 covers techniques for using watermarks to verify the integrity of the content in which they are embedded. This includes the area of fragile watermarks, which disappear or become invalid if the watermarked Work is degraded in any way.

ACKNOWLEDGMENTS

First, we must thank several people who have directly helped us in making this book. Thanks to Karyn Johnson, Jennifer Mann, and Marnie Boyd of Morgan Kaufmann for their enthusiasm and help with this book. As reviewers, Ton Kalker, Rade Petrovic, Steve Decker, Adnan Alattar, Aaron Birenboim, and Gary Hartwick provided valuable feedback. Harold Stone and Steve Weinstein of NEC also gave us many hours of valuable discussion. And much of our thinking about authentication (Chapter 11) was shaped by a conversation with Dr. Richard Green of the Metropolitan Police Service, Scotland Yard. We also thank M. Gwenael Doerr for his review.

Special thanks, too, to Valerie Tucci, our librarian at NEC, who was invaluable in obtaining many, sometimes obscure, publications. And Karen Hahn for secretarial support. Finally, thanks to Dave Waltz, Mitsuhito Sakaguchi, and NEC Research Institute for providing the resources needed to write this book. It could not have been written otherwise.

We are also grateful to many researchers and engineers who have helped develop our understanding of this field over the last several years. Our work on watermarking began in 1995 thanks to a talk Larry O'Gorman presented at NECI. Joe Kilian, Tom Leighton, and Talal Shamoon were early collaborators. Joe has continued to provide valuable insights and support. Warren Smith has taught us much about high-dimensional geometry. Jont Allen, Jim Flanagan, and Jim Johnston helped us understand auditory perceptual modeling. Thanks also to those at NEC Central Research Labs who worked with us on several watermarking projects: Ryoma Oami, Takahiro Kimoto, Atsushi Murashima, and Naoki Shibata.

Each summer we had the good fortune to have excellent summer students who helped solve some difficult problems. Thanks to Andy McKellips and Min Wu of Princeton University and Ching-Yung Lin of Columbia University. We also had the good fortune to collaborate with professors Mike Orchard and Stu Schwartz of Princeton University.

We probably learned more about watermarking during our involvment in the request for proposals for watermarking technologies for DVD disks than at any other time. We are therefore grateful to our competitors for pushing us to our limits, especially Jean-Paul Linnartz, Ton Kalker (again), and Maurice Maes of Philips; Jeffrey Rhoads of Digimarc; John Ryan and Patrice Capitant of Macrovision; and Akio Koide, N. Morimoto, Shu Shimizu, Kohichi Kamijoh, and Tadashi Mizutani of IBM (with whom we later collaborated). We are also grateful to the engineers of NEC's PC&C division who worked on hardware implementations for this competition, especially Kazuyoshi Tanaka, Junya Watanabe, Yutaka Wakasu, and Shigeyuki Kurahashi.

Much of our work was conducted while we were employed at Signafy, and we are grateful to several Signafy personnel who helped with the technical challenges: Peter Blicher, Yui Man Lui, Doug Rayner, Jan Edler, and Alan Stein (whose real-time video library is amazing).

We wish also to thank the many others who have helped us out in a variety of ways. A special thanks to Phil Feig—our favorite patent attorney—for filing many of our patent applications with the minimum of overhead. Thanks to Takao Nishitani for supporting our cooperation with NEC's Central Research Labs. Thanks to Kasinath Anupindi, Kelly Feng, and Sanjay Palnitkar for system administration support. Thanks to Jim Philbin, Doug Bercow, Marc Triaureau, Gail Berreitter, and John Anello for making Signafy a fun and functioning place to work. Thanks to Alan Bell for making CPTWG possible. Thanks to Mitsuhito Sakaguchi (again), who first suggested that we become involved in the CPTWG meetings. Thanks to Shichiro Tsuruta for managing PC&C's effort during the CPTWG competition, and H. Morito of NEC's semiconductor division. Thanks to Dan Sullivan for the part he played in our collaboration with IBM. Thanks to the DHSG cochairs who organized the competition: Bob Finger, Jerry Pierce, and Paul Wehrenberg. Thanks also to the many people at the Hollywood studios who provided us with the content owners' perspective: Chris Cookson and Paul Klamer of Warner Brothers, Bob Lambert of Disney, Paul Heimbach and Gary Hartwick of Viacom, Jane Sunderland and David Grant of Fox, David Stebbings of the RIAA, and Paul Egge of the MPAA. Thanks to Christine Podilchuk for her support. It was much appreciated. Thanks to Bill Connolly for interesting discussions. Thanks to John Kulp, Rafael Alonso, the Sarnoff Corporation, and John Manville of Lehman Brothers for their support. And thanks to Vince Gentile, Tom Belton, Susan Kleiner, Ginger Mosier, Tom Nagle, and Cynthia Thorpe.

Finally, we thank our families for their patience and support during this project: Susan and Zoe Cox, Geidre Miller, and Pamela Bloom.

Preface to the Second Edition

It has been almost 7 years since the publication of *Digital Watermarking*. During this period there has been significant progress in digital watermarking; and the field of steganography has witnessed increasing interest since the terrorist events of September 11, 2001.

Digital watermarking and steganography are closely related. In the first edition of *Digital Watermarking* we made a decision to distinguish between watermarking and steganography and to focus exclusively on the former. For this second edition we decided to broaden the coverage to include steganography and to therefore change the title of the book to *Digital Watermarking and Steganography*.

Despite the new title, this is *not* a new book, but a revision of the original. We hope this is clear from the backcover material and apologize in advance to any reader who thought otherwise.

CONTENT AND ORGANIZATION

The organization of this book closely follows that of the original. The treatment of watermarking and steganography is, for the most part, kept separate. The reasons for this are twofold. First, we anticipate that readers might prefer not to read the book from cover to cover, but rather read specific chapters of interest. And second, an integrated revision would require considerably more work.

Chapters 1 and 2 include new material related to steganography and, where necessary, updated material related to watermarking. In particular, Chapter 2 highlights the similarities and differences between watermarking and steganography.

Chapters 3, 4, 7, 8, 9, and 10 remain untouched, except that bibliographic citations have been updated.

Chapter 5 of the first edition has now been expanded to two chapters, reflecting the research interest in modeling watermarking as communications with side information. Chapter 5 provides a more detailed theoretical discussion of the topic, especially with regard to dirty-paper coding. Chapter 6 then provides a description of a variety of common dirty-paper coding techniques for digital watermarking.

Section 11.1.3 in Chapter 11 has been revised to include material on a variety of erasable watermarking methods.

Finally, two new chapters, Chapters 12 and 13, have been added. These chapters discuss steganography and steganalysis, respectively.

ACKNOWLEDGMENTS

The authors would like to thank the following people: Alan Bell of Warner Brothers for discussions on HD-DVD digital rights management technology, John Choi for discussions relating to watermarking of MP3 files in Korea, David Soukal for creating graphics for the Stego chapter.

And of course we would like to thank our families and friends for their support in the endeavor: Rimante Okkels; Zoe, Geoff, and Astrid Cox; Pam Bloom and her watermarking team of Joshua, Madison, Emily Giedre, Fia, and Ada; Monika, Nicole, and Kathy Fridrich; Miroslav Goljan; Robin Redding; and all the animals.

Finally, to Matt, your coauthors send their strongest wishes—get well soon!

Example Watermarking Systems

In this book, we present a number of example watermarking systems to illustrate and test some of the main points. Discussions of test results provide additional insights and lead to subsequent sections.

Each investigation begins with a preamble. If a new watermarking system is being used, a description of the system is provided. Experimental procedures and results are then described.

The watermark embedders and watermark detectors that make up these systems are given names and are referred to many times throughout the book. The naming convention we use is as follows: All embedder and detector names are written in sans serif font to help set them apart from the other text. Embedder names all start with E_ and are followed by a word or acronym describing one of the main techniques illustrated by an algorithm. Similarly, detector names begin with D_ followed by a word or acronym. For example, the embedder in the first system is named E_BLIND (it is an implementation of blind embedding), and the detector is named D_LC (it is an implementation of linear correlation detection).

Each system used in an investigation consists of an embedder and a detector. In many cases, one or the other of these is shared with several other systems. For example, in Chapter 3, the D_LC detector is paired with the E_BLIND embedder in System 1 and with the E_FIXED_LC embedder in System 2. In subsequent chapters, this same detector appears again in a number of other systems. Each individual embedder and detector is described in detail in the first system in which it is used.

In the following, we list each of the 19 systems described in the text, along with the number of the page on which its description begins, as well as a brief review of the points it is meant to illustrate and how it works. The source code for these systems is provided in Appendix C.

System 1: E_BLIND/D_LC 70

Blind Embedding and Linear Correlation Detection: The blind embedder E_BLIND simply adds a pattern to an image. A reference pattern is scaled by a strength parameter, α, prior to being added to the image. Its sign is dictated by the message being encoded.

The D_LC linear correlation detector calculates the correlation between the received image and the reference pattern. If the magnitude of the correlation is higher than a threshold, the watermark is declared to be present. The message is encoded in the sign of the correlation.

System 2: E_FIXED_LC/D_LC 77

Fixed Linear Correlation Embedder and Linear Correlation Detection: This system uses the same D_LC linear correlation detector as System 1, but introduces a new embedding algorithm that implements a type of informed embedding. Interpreting the cover Work as channel noise that is known, the E_FIXED_LC embedder adjusts the strength of the watermark to compensate for this noise, to ensure that the watermarked Work has a specified linear correlation with the reference pattern.

System 3: E_BLK_BLIND/D_BLK_CC 89

Block-Based, Blind Embedding, and Correlation Coefficient Detection: This system illustrates the division of watermarking into *media space* and *marking space* by use of an extraction function. It also introduces the use of the correlation coefficient as a detection measure.

The E_BLK_BLIND embedder performs three basic steps. First, a 64-dimensional vector, v_o, is extracted from the unwatermarked image by averaging 8 × 8 blocks. Second, a reference mark, w_r, is scaled and either added to or subtracted from v_o. This yields a marked vector, v_w. Finally, the difference between v_o and v_w is added to each block in the image, thus ensuring that the extraction process (block averaging), when applied to the resulting image, will yield v_w.

The D_BLK_CC detector extracts a vector from an image by averaging 8 × 8 pixel blocks. It then compares the resulting 64-dimensional vector, v, against a reference mark using the correlation coefficient.

System 4: E_SIMPLE_8/D_SIMPLE_8 116

8-Bit Blind Embedder, 8-Bit Detector: The E_SIMPLE_8 embedder is a version of the E_BLIND embedder modified to embed 8-bit messages. It first constructs a message pattern by adding or subtracting each of eight reference patterns. Each reference pattern denotes 1 bit, and the sign of the bit determines whether it is added or subtracted. It then multiplies the message pattern by a scaling factor and adds it to the image.

The D_SIMPLE_BITS detector correlates the received image against each of the eight reference patterns and uses the sign of each correlation to determine the most likely value for the corresponding bit. This yields the decoded message. The detector does not distinguish between marked and unwatermarked images.

System 5: E_TRELLIS_8/D_TRELLIS_8 123

Trellis-Coding Embedder, Viterbi Detector: This system embeds 8-bit messages using trellis-coded modulation. In the E_TRELLIS_8 embedder, the 8-bit

message is redundantly encoded as a sequence of symbols drawn from an alphabet of 16 symbols. A message pattern is then constructed by adding together reference patterns representing the symbols in the sequence. The pattern is then embedded with blind embedding.

The D_TRELLIS_8 detector uses a Viterbi decoder to determine the most likely 8-bit message. It does not distinguish between watermarked and unwatermarked images.

System 6: E_BLK_8/D_BLK_8 131

Block-Based Trellis-Coding Embedder and Block-Based Viterbi Detector That Detects by Reencoding: This system illustrates a method of testing for the presence of multibit watermarks using the correlation coefficient. The E_BLK_8 embedder is similar to the E_TRELLIS_8 embedder, in that it encodes an 8-bit message with trellis-coded modulation. However, it constructs an 8 × 8 message mark, which is embedded into the 8 × 8 average of blocks in the image, in the same way as the E_BLK_BLIND embedder.

The D_BLK_8 detector averages 8 × 8 blocks and uses a Viterbi decoder to identify the most likely 8-bit message. It then reencodes that 8-bit message to find the most likely message mark, and tests for that message mark using the correlation coefficient.

System 7: E_BLK_FIXED_CC/D_BLK_CC 144

Block-Based Watermarks with Fixed Normalized Correlation Embedding: This is a first attempt at informed embedding for normalized correlation detection. Like the E_FIXED_LC embedder, the E_BLK_FIXED_CC embedder aims to ensure a specified detection value. However, experiments with this system show that its robustness is not as high as might be hoped.

The E_BLK_FIXED_CC embedder is based on the E_BLK_BLIND embedder, performing the same basic three steps of extracting a vector from the unwatermarked image, modifying that vector to embed the mark, and then modifying the image so that it will yield the new extracted vector. However, rather than modify the extracted vector by blindly adding or subtracting a reference mark, the E_BLK_FIXED_CC embedder finds the closest point in 64 space that will yield a specified correlation coefficient with the reference mark. The D_BLK_CC detector used here is the same as in the E_BLK_BLIND/D_BLK_CC system.

System 8: E_BLK_FIXED_R/D_BLK_CC 149

Block-Based Watermarks with Fixed Robustness Embedding: This system fixes the difficulty with the E_BLK_FIXED_CC/D_BLK_CC system by trying to obtain a fixed estimate of robustness, rather than a fixed detection value.

After extracting a vector from the unwatermarked image, the E_BLK_FIXED_R embedder finds the closest point in 64 space that is likely to lie within the detection region even after a specified amount of noise has been added. The D_BLK_CC detector used here is the same as in the E_BLK_BLIND/D_BLK_CC system.

System 9: E_LATTICE/D_LATTICE 191
Lattice-Coded Watermarks: This illustrates a method of watermarking with dirty-paper codes that can yield much higher data payloads than are practical with the E_DIRTY_PAPER/D_DIRTY_PAPER system. Here, the set of code vectors is not random. Rather, each code vector is a point on a lattice. Each message is represented by all points on a sublattice.

The embedder takes a 345-bit message and applies an error correction code to obtain a sequence of 1,380 bits. It then identifies the sublattice that corresponds to this sequence of bits and quantizes the cover image to find the closest point in that sublattice. Finally, it modifies the image to obtain a watermarked image close to this lattice point.

The detector quantizes its input image to obtain the closest point on the entire lattice. It then identifies the sublattice that contains this point, which corresponds to a sequence of 1,380 bits. Finally, it decodes this bit sequence to obtain a 345-bit message. It makes no attempt to determine whether or not a watermark is present, but simply returns a random message when presented with an unwatermarked image.

System 10: E_E$_8$LATTICE/D_E$_8$LATTICE 202
E_8 *Lattice-Coded Watermarks:* This System illustrates the benefits of using an E_8 lattice over an orthogonal lattice, used in System 9. Experimental results compare the performance of System 10 and System 9 and demonstrate that the E_8 lattice has superior performance.

System 11: E_BLIND/D_WHITE 234
Blind Embedding and Whitened Linear Correlation Detection: This system explores the effects of applying a whitening filter in linear correlation detection. It uses the E_BLIND embedding algorithm introduced in System 1.

The D_WHITE detector applies a whitening filter to the image and the watermark reference pattern before computing the linear correlation between them. The whitening filter is an 11 × 11 kernel derived from a simple model of the distribution of unwatermarked images as an elliptical Gaussian.

System 12: E_BLK_BLIND/D_WHITE_BLK_CC 247

Block-Based Blind Embedding and Whitened Correlation Coefficient Detection: This system explores the effects of whitening on correlation coefficient detection. It uses the E_BLK_BLIND embedding algorithm introduced in System 3.

The D_WHITE_BLK_CC detector first extracts a 64 vector from the image by averaging 8 × 8 blocks. It then filters the result with the same whitening filter used in D_WHITE. This is roughly equivalent to filtering the image before extracting the vector. Finally, it computes the correlation coefficient between the filtered, extracted vector and a filtered version of a reference mark.

System 13: E_PERC_GSCALE 277

Perceptually Limited Embedding and Linear Correlation Detection: This system begins an exploration of the use of perceptual models in watermark embedding. It uses the D_LC detector introduced in System 1.

The E_PERC_GSCALE embedder is similar to the E_BLIND embedder in that, ultimately, it scales the reference mark and adds it to the image. However, in E_PERC_GSCALE the scaling is automatically chosen to obtain a specified perceptual distance, as measured by Watson's perceptual model.

System 14: E_PERC_SHAPE 284

Perceptually Shaped Embedding and Linear Correlation Detection: This system is similar to System 11, but before computing the scaling factor for the entire reference pattern the E_PERC_SHAPE embedder first *perceptually shapes* the pattern.

The perceptual shaping is performed in three steps. First, the embedder converts the reference pattern into the block DCT domain (the domain in which Watson's model is defined). Next, it scales each term of the transformed reference pattern by a corresponding *slack* value obtained by applying Watson's model to the cover image. This amplifies the pattern in areas where the image can easily hide noise, and attenuates in areas where noise would be visible. Finally, the resultant shaped pattern is converted back into the spatial domain. The shaped pattern is then scaled and added to the image in the same manner as in E_PERC_GSCALE.

System 15: E_PERC_OPT 290

Optimally Scaled Embedding and Linear Correlation Detection: This system is essentially the same as System 12. The only difference is that perceptual shaping is performed using an "optimal" algorithm, instead of simply scaling each term of the reference pattern's block DCT. This shaping is optimal in the sense that the resulting pattern yields the highest possible correlation with the reference pattern for a given perceptual distance (as measured by Watson's model).

System 16: E_MOD/D_LC 381

Watermark Embedding Using Modulo Addition: This is a simple example of a system that produces erasable watermarks. It uses the D_LC detector introduced in System 1.

The E_MOD embedder is essentially the same as the E_BLIND embedder, in that it scales a reference pattern and adds it to the image. The difference is that the E_MOD embedder uses modulo 256 addition. This means that rather than being clipped to a range of 0 to 255, the pixel values wrap around. Therefore, for example, 253 + 4 becomes 1. Because of this wraparound, it is possible for someone who knows the watermark pattern and embedding strength to perfectly invert the embedding process, erasing the watermark and obtaining a bit-for-bit copy of the original.

System 17: E_DCTQ/D_DCTQ 400

Semi-fragile Watermarking: This system illustrates a carefully targeted semi-fragile watermark intended for authenticating images. The watermarks are designed to be robust against JPEG compression down to a specified quality factor, but fragile against most other processes (including more severe JPEG compression).

The E_DCTQ embedder first converts the image into the block DCT domain used by JPEG. It then quantizes several high-frequency coefficients in each block to either an even or odd multiple of a quantization step size. Each quantized coefficient encodes either a 0, if it is quantized to an even multiple, or a 1, if quantized to an odd multiple. The pattern of 1s and 0s embedded depends on a key that is shared with the detector. The quantization step sizes are chosen according to the expected effect of JPEG compression at the worst quality factor the watermark should survive.

The D_DCTQ detector converts the image into the block DCT domain and identifies the closest quantization multiples for each of the high-frequency coefficients used during embedding. From these, it obtains a pattern of bits, which it compares against the pattern embedded. If enough bits match, the detector declares that the watermark is present.

The D_DCTQ detector can be modified to yield localized information about where an image has been corrupted. This is done by checking the number of correct bits in each block independently. Any block with enough correctly embedded bits is deemed authentic.

System 18: E_SFSIG/D_SFSIG 406

Semi-fragile Signature: This extends the E_DCTQ/D_DCTQ system to provide detection of distortions that only effect the low-frequency terms of the block DCT. Here, the embedded bit pattern is a semi-fragile signature derived from the low-frequency terms of the block DCT.

The E_SFSIG embedder computes a bit pattern by comparing the magnitudes of corresponding low-frequency coefficients in randomly selected pairs of blocks. Because quantization usually does not affect the relative magnitudes of different values, most bits of this signature should be unaffected by JPEG (which quantizes images in the block DCT domain). The signature is embedded in the high-frequency coefficients of the blocks using the same method used in E_DCTQ.

The D_SFSIG detector computes a signature in the same way as E_SFSIG and compares it against the watermark found in the high-frequency coefficients. If enough bits match, the watermark is deemed present.

System 19: E_PXL/D_PXL 412

Pixel-by-Pixel Localized Authentication: This system illustrates a method of authenticating images with pixel-by-pixel localization. That is, the detector determines whether each individual pixel is authentic.

The E_PXL embedder embeds a predefined binary pattern, usually a tiled logo that can be easily recognized by human observers. Each bit is embedded in one pixel according to a secret mapping of pixel values into bit values (known to both embedder and detector). The pixel is moved to the closest value that maps to the desired bit value. Error diffusion is used to minimize the perceptual impact.

The D_PXL detector simply maps each pixel value to a bit value according to the secret mapping. Regions of the image modified since the watermark was embedded result in essentially random bit patterns, whereas unmodified regions result in the embedded pattern. By examining the detected bit pattern, it is easy to see where the image has been modified.

System 20: SE_LTSOLVER 463

Linear System Solver for Matrices Satisfying Robust Soliton Distribution: This system describes a method for solving a system of linear equations, $Ax = y$, when the Hamming weights of the matrix A columns follow a robust soliton distribution. It is intended to be used as part of a practical implementation of wet paper codes with non-shared selection rules.

The SE_LTSOLVER accepts on its input the linear system matrix, A, and the right hand side, y, and outputs the solution to the system if it exists, or a message that the solution cannot be found. The solution proceeds by repeatedly swapping the rows and columns of the matrix until an upper diagonal matrix is obtained (if the system has a solution). The solution is then found by backsubstitution as in classical Gaussian elimination and re-permuting the solution vector.

System 21: SD_SPA 484

Detector of LSB Embedding: This is a steganalysis system that detects images with messages embedded using LSB embedding. It uses sample pairs analysis to estimate the number of flipped LSBs in an image and thereby detect LSB steganography.

It works by first dividing all pixels in the image into pairs and then assigns them to several categories. The cardinalities of the categories are used to form a quadratic equation for the unknown relative number of flipped LSBs. The input is a grayscale image, the output is the estimate of the relative message length in bits per pixel.

System 22: SD_DEN_FEATURES 491

Blind Steganalysis in Spatial Domain based on de-noising and a feature vector: This system extracts 27 features from a grayscale image for the purpose of blind steganlysis primarily in the spatial domain.

The SD_DEN_FEATURES system first applies a denoising filter to the image and then extracts the noise residual, which is subsequently transformed to the wavelet domain. Statistical moments of the coefficients from the three highest-frequency subbands are then calculated as features for steganalysis. Classification can be performed using a variety of machine learning tools.

CHAPTER 1
Introduction

Hold an American $20 bill up to the light. If you are looking at the side with the portrait of president Andrew Jackson, you will see that the portrait is echoed in a watermark on the right. This watermark is embedded directly into the paper during the papermaking process, and is therefore very difficult to forge. It also thwarts a common method of counterfeiting in which the counterfeiter washes the ink out of $20 bills and prints $100 bills on the same paper.

The watermark on the $20 bill (Figure 1.1), just like most paper watermarks today, has two properties that relate to the subject of the present book. First, the watermark is hidden from view during normal use, only becoming visible as a result of a special viewing process (in this case, holding the bill up to the light). Second, the watermark carries information about the object in which it is hidden (in this case, the watermark indicates the authenticity of the bill).

In addition to paper, watermarking can be applied to other physical objects and to electronic signals. Fabrics, garment labels, and product packaging are examples of physical objects that can be watermarked using special invisible dyes and inks [344, 348]. Electronic representations of music, photographs, and video are common types of signals that can be watermarked.

Consider another example that also involves imperceptible marking of paper but is fundamentally different on a philosophical level. Imagine a spy called Alice who needs to communicate a very important finding to her superiors. Alice begins by writing a letter describing her wonderful recent family vacation. After writing the letter, Alice replaces the ink in her pen with milk and writes a top secret message between the inked lines of her letter. When the milk dries, this secret message becomes imperceptible to the human eye. Heating up the paper above a candle will make the secret message visible. This is an example of steganography. In contrast to watermarking, the hidden message is unrelated to the content of the letter, which only serves as a decoy or cover to hide the very presence of sending the secret message.

FIGURE 1.1

American $20 bill.

This book deals with both watermarking and steganology[1] of electronic signals. We adopt the following terminology to describe these signals. We refer to a specific song, video, or picture—or to a specific copy of such—as a *Work,*[2] and to the set of all possible Works as *content*. Thus, audio music is an example of content, and the song "Satisfaction" by the Rolling Stones is an example of a Work. The original unaltered Work is sometimes referred to as the *cover Work,* in that it hides or "covers" the watermark or the secret message. We use the term *media* to refer to the means of representing, transmitting, and recording content. Thus, the audio CD on which "Satisfaction" is recorded is an example of a medium.

> *We define* watermarking *as the practice of imperceptibly altering a Work to embed a message about that Work.*[3]
> *We define* steganography *as the practice of undetectably altering a Work to embed a secret message.*

Even though the objectives of watermarking and steganography are quite different, both applications share certain high-level elements. Both systems

[1] We use the term *steganology* to refer to both steganography and steganalysis, just as cryptology refers to both cryptography and cryptanalysis. The term *steganology* is not commonly used but is more precise than using steganography. However, we will often use steganography and steganology interchangeably.

[2] This definition of the term *Work* is consistent with the language used in the United States copyright laws [416]. Other terms that have been used can be found in the disscussion of this term in the Glossary.

[3] Some researchers do not consider imperceptibility a defining characteristic of digital watermarking. This leads to the field of perceptible watermarking [52, 164, 286, 294, 295], which is outside the scope of this book.

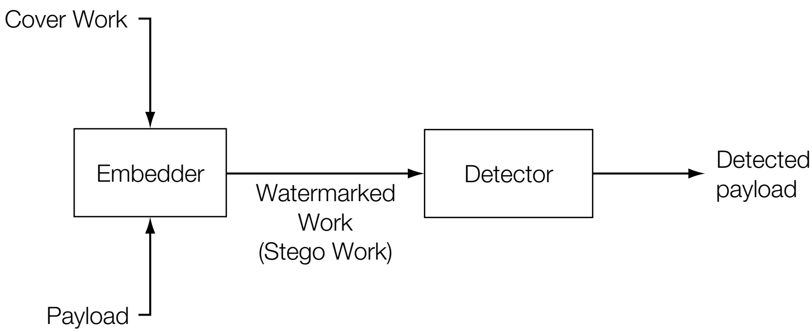

FIGURE 1.2

A generic watermarking (steganography) system.

consist of an *embedder* and a *detector,* as illustrated in Figure 1.2. The embedder takes two inputs. One is the payload we want to embed (e.g., either the watermark or the secret message), and the other is the cover Work in which we want to embed the payload. The output of the embedder is typically transmitted or recorded. Later, that Work (or some other Work that has not been through the embedder) is presented as an input to the detector. Most detectors try to determine whether a payload is present, and if so, output the message encoded by it.

In the late 1990s there was an explosion of interest in digital systems for the watermarking of various content. The main focus has been on photographs, audio, and video, but other content—such as binary images [453], text [49, 50, 271], line drawings [380], three-dimensional models [36, 312, 462], animation parameters [177], executable code [385], and integrated circuits [215, 249]—have also been marked. The proposed applications of these methods are many and varied, and include identification of the copyright owner, indication to recording equipment that the marked content should not be recorded, verification that content has not been modified since the mark was embedded, and the monitoring of broadcast channels looking for marked content.

Interest in steganology increased significantly after the terrorist attacks on September 11, 2001, when it became clear that means for concealing the communication itself are likely to be used for criminal activities.[4] The first steganalytic methods focused on the most common type of hiding called Least Significant Bit embedding [142, 444] in bitmap and GIF images. Later, substantial effort has been directed to the most common image format—JPEG—[132, 144] and audio files [443]. Accurate methods for detecting hidden messages prompted further research in steganography for multimedia files [147, 442].

[4] Interestingly, *USA Today* reported on this possibility several months before the September 11, 2001 attacks [1]. However, there has been little evidence to substantiate these claims.