Robust Digital Image Hashing Algorithms for Image Identification

by

Xudong Lv

B.Sc., The University of Science and Technology of China, 2007

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENTS FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

in

The Faculty of Graduate Studies

(Electrical and Computer Engineering)

THE UNIVERSITY OF BRITISH COLUMBIA

(Vancouver)

April 2013

Abstract

Image hashing has been a popular alternative of digital watermarking for copyright protection and content authentication of digital images, due to its two critical properties – robustness and security. Also, its uniqueness and compactness make image hashing attractive for efficient image indexing and retrieval applications. In this thesis, novel image hashing algorithms are proposed to improve the robustness of digital image hashing against various perceptually insignificant manipulations and distortions on image content. Furthermore, image hashing concept is extended to the content-based fingerprinting concept, which combines various hashing schemes efficiently to achieve superior robustness and higher identification accuracy.

The first contribution of the thesis is the novel FJLT image hashing, which applies a recently proposed low-distortion dimension reduction technique, referred as Fast Johnson-Lindenstrauss Transform (FJLT), into image hashing generation. FJLT shares the low distortion characteristics of random projections, but requires less computational cost, which are desirable properties to generate robust and secure image hashes. The Fourier-Mellin transform can also be incorporated into FJLT hashing to improve its performances under rotation attacks. Further, the content-based fingerprinting concept is proposed, which combines different FJLT-based hashes to achieve better overall robustness and identification capability.

The second contribution of the thesis is the novel shape contexts based image hashing (SCH) using robust local feature points. The robust SIFT-Harris detector is proposed to select the most stable feature points under various content-preserving distortions, and compact and robust image hashes are generated by embedding the detected feature points into the shape contexts based descriptors. The proposed SCH approach yields better identi-

fication performances under geometric attacks and brightness changes, and provides comparable performances under classical distortions.

The third contribution of this thesis addresses an important issue of compressing the real-valued image hashes into robust short binary image hashes. By exploring prior information from the virtual prior attacked hash space (VPAHS), the proposed semi-supervised spectral embedding approach could compress real-valued hashes into compact binary signatures, while the robustness against different attacks and distortions are preserved. Moreover, the proposed SSE framework could be easily generalized to combine different types of image hashes to generate a robust, fixed-length binary signature.

Preface

This thesis is written based on a collection of manuscripts, which are from the collaboration of several researchers. The majority of the research, including literature reviews, algorithm designs and implementations, simulation tests and results analysis, are conducted by the author, with suggestions from Prof. Z. Jane Wang. The manuscripts are primarily drafted by the author, with helpful revisions and comments from Prof. Z. Jane Wang.

Chapter 2 is partially based on the following manuscripts:

- X. Lv, M. Fatourechi, and Z. J. Wang, "A Survey of Image Hashing and Content-based Fingerprinting Literatures", *Technical Report*, 2010.
- M. Fatourechi, X. Lv, M. M. Esmaeili, Z. J. Wang, and R. K. Ward, "Image and Video Copy Detection using Content-Based Fingerprinting", *Book Chapter, Multimedia Image and Video Processing*, Second Edition, 2012.

Chapter 3 is based on the following manuscripts:

- X. Lv, and Z. J. Wang, "An Extended Image Hashing Concept: Content-based Fingerprinting using FJLT", *EURASIP Journal on Information Security*, 2009:1-17, 2009.
- X. Lv, and Z. J. Wang, "Fast Johnson-Lindenstrauss Transform for Robust and Secure Image Hashing", *Proc. of the IEEE 10th Workshop on Multimedia Signal Processing (MMSP)*, pp: 725-729, 2008.

Chapter 4 is based on the following manuscripts:

- X. Lv, and Z. J. Wang, "Perceptual Image Hashing Based on Shape Contexts and Local Feature Points", *IEEE Transactions on Information Forensics and Security*, 7(3):1081-1093, June 2012.
- X. Lv, and Z. J. Wang, "Shape Contexts Based Image Hashing using Local Feature Points", *Proc. of the IEEE International Conference on Image Processing (ICIP)*, pp:2 2541-2544, 2011.

Chapter 5 is based on the following manuscripts:

- X. Lv, and Z. J. Wang, "Compressed Binary Image Hashes Based on Semi-supervised Spectral Embedding", submitted, 2012.

Table of Contents

Abstract ii

Preface iv

Table of Contents vi

List of Tables x

List of Figures xii

List of Acronyms xiv

Acknowledgements xvi

1 Introduction 1
- 1.1 Background 1
- 1.2 Challenges and Motivation 2
- 1.3 Image Hashing and Content-Based Fingerprinting 3
  - 1.3.1 Concept and Properties 3
  - 1.3.2 Generic Framework 5
- 1.4 Contributions and Thesis Outline 7
  - 1.4.1 Thesis Contributions 7
  - 1.4.2 Thesis Outline 8

2 Image Hashing and Content-Based Fingerprinting Review 11
- 2.1 Review Framework 11
- 2.2 Pre-Processing 13
- 2.3 Robust Feature Extraction 15

2.4 Feature Compression and Post-Processing 17
2.5 Security Incorporation 19
2.6 Comparison and Decision Making 20
2.6.1 Distance Metrics 21
2.6.2 Classifiers 21
2.7 Evaluation Criteria 22
2.7.1 Perceptual Robustness 22
2.7.2 Security Analysis 25
2.8 Conclusion 26

3 Image Hashing and Content-Based Fingerprinting Based on Fast Johnson-Lindenstrauss Transform (FJLT) 28
3.1 Introduction 28
3.2 Theoretical Background 30
3.2.1 Fast Johnson-Lindenstrauss Transform 30
3.2.2 The Fast Johnson-Lindenstrauss Lemma 32
3.3 Image Hashing via FJLT 32
3.3.1 Random Sampling 33
3.3.2 Dimension Reduction by FJLT 34
3.3.3 Ordered Random Weighting 35
3.3.4 Identification and Evaluation 37
3.4 Rotation Invariant FJLT Hashing 40
3.4.1 Fourier-Mellin Transform 40
3.4.2 RI-FJLT Hashing 42
3.5 Content-Based Fingerprinting 44
3.5.1 Concept and Framework 44
3.5.2 A Simple Content-Based Fingerprinting Approach . 45
3.6 Experimental Results and Analysis 47
3.6.1 Database and Content-Preserving Manipulations . . 47
3.6.2 Identification Results and ROC Analysis 47
3.6.3 Unpredictability Analysis 55
3.6.4 Computational Complexity 57
3.7 Conclusion 58

4 Perceptual Image Hashing Based on Shape Contexts and Local Feature Points 62
4.1 Introduction 62
4.2 Robust Local Feature Points 63
4.2.1 Scale Invariant Feature Transform Review 64
4.2.2 Robust Keypoints Detection Using Harris Criterion . 66
4.2.3 Detection Evaluation 69
4.3 Image Hashing Based on Shape Contexts 72
4.3.1 Shape Contexts 73
4.3.2 Shape Contexts-Based Image Hashing 74
4.4 Experimental Results and Analysis 79
4.4.1 Evaluation of Perceptual Robustness 79
4.4.2 Evaluation of Tampering Detection 86
4.4.3 Unpredictability Analysis 89
4.4.4 CPU Time Cost 91
4.5 Conclusion 91

5 Compressed Binary Image Hashes Based on Semi-Supervised Spectral Embedding 93
5.1 Introduction 93
5.2 Motivation 95
5.2.1 Related Literature Review 95
5.2.2 VPAHS and Motivation 96
5.3 Proposed Binary Image Hash Generation 99
5.3.1 Spectral Embedding Ideas Review 99
5.3.2 Proposed Semi-Supervised Spectral Embedding (SSE) 102
5.3.3 Out-of-Sample Extension 104
5.3.4 Proposed Method for Binary Image Hash Construction 106
5.4 Proposed Framework for Combining Multiple Image Hashes 107
5.5 Experimental Results and Analysis 109
5.5.1 Database and Content-Preserving Manipulations . . . 109
5.5.2 Identification and Evaluation Measures 109

5.5.3 Intermediate Hashes and Baseline Methods 111
5.5.4 Embedding Training 113
5.5.5 Experimental Results 115
5.6 Conclusion 120

6 Conclusions and Future Works 122
6.1 Conclusions 122
6.2 Future Works 125
6.2.1 Learning Optimal Fusion on Hash Spaces Based on Semi-Supervised Spectral Embedding 125
6.2.2 Measurable Robustness and Sensitivity Toward Image Quality Changes 126
6.2.3 Hashing at Semantic Feature Levels 126
6.2.4 Universal Security Measurements 127

Bibliography 128

List of Tables

2.1 Literature references regarding pre-processing 14
2.2 Literature references regarding robust feature extraction . . 17
2.3 Literature references regarding feature compression and post-processing 19
2.4 Literature references regarding security incorporation 20
2.5 Literature references regarding content-preserving attacks part one 23
2.6 Literature references regarding content-preserving attacks part two 24
2.7 Literature references regarding on malicious attacks 25

3.1 Content-preserving manipulations and parameter settings . . 48
3.2 Identification accuracy for manipulated images by NMF-NMF-SQ (NMF) hashing, FJLT hashing and content-based fingerprinting (CBF) based on FJLT & RI-FJLT hashing) 49
3.3 Parameter setting in the FJLT hashing algorithm 50
3.4 Identification accuracy under rotation attacks by FJLT and RI-FJLT hashing 53
3.5 Computational time costs for lena with 256×256 by FJLT, RI-FJLT and NMF-NMF-SQ hashing algorithms 58

4.1 Average Hausdorff distances between the coordinates of the top 20 keypoints detected in the original image and manipulated copies using the proposed SIFT-Harris and end-stopped [73] detectors. 72
4.2 Content-preserving manipulations and parameters setting . . 80

4.3 Identification accuracy performances by RSCH, ASCH, R&A SCH, NMF, FJLT, RI-FJLT hashing algorithms under different attacks 82
4.4 The average CPU times required by the proposed SCH, FJLT, and NMF hashing approaches. 91

5.1 Content-preserving manipulations and parameters setting . . 110
5.2 Identification accuracy performances of different hashing algorithms under various attacks 115
5.3 The comparison of average identification accuracy of binary image hashes based on the proposed SSE and conventional-quantization methods 119

List of Figures

1.1 Examples of distorted image copies under different content-preserving attacks. 6
1.2 The generic framework of image hashing 7

2.1 The framework of digital image hashing and content-based fingerprinting review 12

3.1 An example of random sampling 33
3.2 An example of the correlations between the final hash distance and the intermediate hash distance under salt & pepper noise attacks when employing ordered random weighting and unordered random weighting 38
3.3 Illustrative examples to demonstrate the effect of ordering on the identification performance. 39
3.4 An example of conversion from Cartesian coordinates to log-polar coordinates. 43
3.5 The conceptual framework of the content-based fingerprinting 44
3.6 The overall ROC curves of NMF-NMF-SQ hashing, FJLT hashing, and content-based fingerprinting under all types of tested manipulations 50
3.7 The histogram of a typical FJLT hash vector component and the covariance matrix of the FJLT hash vectors for image lena from 3000 different secret keys. 56

4.1 A performance comparison of keypoints detection between SIFT and the proposed SIFT-Harris on the original image, the images under Gaussian noise, blurring and JPEG compression attacks 67
4.2 The average F values of the end-stopped wavelet, the SIFT detector and the proposed SIFT-Harris detector over 20 images under 9 types of content-preserving distortions. 71
4.3 The diagram of the original shape contexts and the proposed shape context hashing: RSCH and ASCH. 73
4.4 The radon transform $R(p, \theta)$ of a 2D function $f(x, y)$ 78
4.5 The ROC curves of the proposed shape contexts based image hashing approaches when compared with the state-of-art NMF, FJLT, and RI-FJLT hashing approaches. 85
4.6 An example of image tampering detection using the proposed shape contexts based image hashing approaches. 88
4.7 The ROC curves for tampering detection 89

5.1 The examples of VPAHS and hash clusters based on the FJLT image hashing scheme 97
5.2 The proposed binary image hash generation using the semi-supervised spectral embedding learning 106
5.3 The ROC curves of the conventional quantization, the SSE, and the joint SSE using FJLTH and SCH. 118

List of Acronyms

FJLT	Fast Johnson-Lindenstrauss Transform
NMF	Non-negative Matrix Factorization
SVD	Singular Value Decomposition
FMT	Fourier-Mellin Transform
SCH	Shape Contexts based Image Hashing
VPAHS	Virtual Prior Attacked Hash Space
SSE	Semi-supervised Spectral Embedding
CBF	Content-based Fingerprinting
RI-FJLTH	Rotation-Invariant FJLT
CBIR	Content-based Image Retrieval
HVS	Human Visual System
SIFT	Scale Invariant Feature Transform
RSCH	Radial Shape Context Hashing
ASCH	Angular Shape Context Hashing
DCT	Discrete Cosine Transform
DWT	Discrete Wavelet Transform
RASH	Radon Soft Hash

ECC	Error-Correcting Coding
PCA	Principal Component Analysis
SVM	Support Vector Machine
GMM	Gaussian Mixture Model
ANN	Approximate Nearest Neighbors
ROC	Receiver Operating Characteristics
MSER	Maximally Stable Extremal Regions
DOG	Difference-of-Gaussian
KNN	K-Nearest-Neighbour
LSH	Local Sensitive Hashing
KDE	Kernel Density Estimation
LDA	Linear Discriminant Analysis

Acknowledgements

The pursuit for PhD degree in the past 5 years at UBC has become a memorable period of time in my life. First and foremost, I would like to thank my supervisor Prof. Z. Jane Wang, for supporting me over the years and giving me the freedom to dig into interesting research topics. With her insightful suggestions and valuable guidance and discussions, I could finally grow up as an independent researcher and finish this thesis.

I would also like to thank Prof. Rabab Ward and Prof. Panos Nasiopoulos for their care and support for my research. I also would like to thank Prof. David Lowe and Prof. Jim Little in the Dept. of Computer Science of UBC for their kind instructions and help on my research.

I would like to thank my labmates, friends and colleagues at UBC, with whom I have had the pleasure of studying and working together over the years. These include Qiang Tang, Junning Li, Haoming Li, Di Xu, Joyce Chiang, Amir Valizadeh, Xiaohui Chen, Chen He, Xun Chen, Aiping Liu, Zhengyu Guo, and all other members in ICICS. Also special thanks to Dr. Rui Liao and Shun Miao for their help during my internship at Siemens USA.

At last, I would like to thank my dear parents for their endless love and support. I couldn't finish this thesis without your love and care. Thanks for all of you.

Chapter 1

Introduction

1.1 Background

During the last two decades, digital media has profoundly changed our daily life. The advanced development of digital cameras and camcorders as well as storage techniques facilitate the massive proliferation of digital media such as images and videos. Due to the advancement of modern networking techniques and the easy-to-copy nature of digital media, millions of digital images and videos are distributed and shared daily over Internet with the help of popular social media network services, such as YouTube and Flicker etc. However, such convenience of the easy distribution of digital media data also raises several critical issues as follows:

- *Copyright Protection*: Due to the nature of Internet, once users upload their images into public websites, everyone could download the digital images without any authorization. The goal of copyright protections is to identify perceptually identical images or videos, even if they suffer from some distortions induced by the imperfect transmission channel or small malicious tampering, and to prevent possible illegal usage of these digital media data.

- *Content Authentication*: For the sake of easy-to-manipulate nature of digital images, content tampering such as object insertion or removal could be easily conducted using certain image processing softwares. Therefore, how to authenticate the integrity of digital image data and identify malicious tampering has become one of the most important issues in digital media security.

- *Efficient Indexing and Retrieval*: The efficient management of large image and video databases could offer users satisfactory query-retrieval services. However, traditional indexing methods based on manual annotations are time consuming and have become the bottleneck of efficient retrieval in large-scale media databases. Hence, automatic indexing schemes for content-based digital data annotation and fast searching algorithms for retrieving query data are desired.

Therefore, how to efficiently manage the large-scale media databases and effectively protect the copyright of digital media data are critical issues to be resolved.

1.2 Challenges and Motivation

Traditionally, digital watermarking is proposed as a promising technique to authenticate the integrity of media data and protect digital copyrights. The fundamental idea of digital watermarking is to embed some authorized signatures, referred as watermark signals, into host digital images or videos invisibly or visibly depending on the application scenarios. At the receiver side, the watermark signals can be detected and extracted as identification information to indicate the ownerships. However, such an active embedding process would inevitably cause some slight or imperceptible modifications on media content, especially when the embedded watermark signals are required to be robust against standard signal processing attacks such as additive noise and compression. Therefore there is always a tradeoff between the strength of the embedded watermark signals and the content quality of the host media data. Also, since the embedded watermark signals are usually independent of the host media, digital watermarking is usually incapable of serving content-based media data retrieval tasks.

To efficiently manage large-scale media databases, especially for digital images, content-based image retrieval (CBIR) [89] has been proposed and studied, which aims to extract features from a low level (e.g., color and texture etc) to a high level (e.g., salient points, objects, and structures etc) and

automatically retrieve the query image by feature matching. However robust features to be used for accurate retrieval usually lie in high dimensional spaces, which require a large storage space for saving and are inappropriate for fast indexing. Also these features generally lack of security protection and are vulnerable to unauthorized adversaries. Therefore the conventional CBIR schemes are not feasible for digital copyright protection.

In this sense, image hashing and the extended content-based fingerprinting concept are proposed and shown to be efficient tools to address issues of efficient image database management and copyright protection.

1.3 Image Hashing and Content-Based Fingerprinting

1.3.1 Concept and Properties

As an alternative way for efficient image database management and copyright protection, perceptual image hashing or content-based image fingerprinting has been proposed to generate an unique, compact, robust and secure signature for each image [96, 101]. Without embedding any additional watermark signal into host images, the generated image hash depends on the image content or characteristics itself.

Given images I and I' and their perceptually similar copies with minor distortion I_d and I'_d, and an image hashing function $H_k(.)$ depending on a secret key k, we can summarize the desired properties of $H_k(.)$ as follows:

- ***Uniqueness***: Perceptually distinct images should have unique signatures

$$Pr(H_k(I) \neq H_k(I')) \geq 1 - \tau,\ 0 \leq \tau < 1. \tag{1.1}$$

The uniqueness of signatures (image hashes) guarantees the applications of image hashing on content-based image identification and retrieval.

- ***Compactness***: The size of the hash signature should be much smaller

than that of the original image I

$$Size(H_k(I)) \ll Size(I). \tag{1.2}$$

Since for large-scale image databases, a critical issue is how to efficiently save and search the memory space for thousands of image hashes. The compactness of image hash is desired, because a short enough signature would facilitate the efficiency of searching and retrieval and require less storage space.

• ***Perceptual Robustness***: Perceptually identical images should have similar signatures

$$Pr(H_k(I) \approx H_k(I_d)) \geq 1 - \epsilon,\ 0 \leq \epsilon < 1. \tag{1.3}$$

Conventional hashing algorithms such as MD-5 and SHA-1 [68] in cryptography are sensitive to even slight changes in messages. While for digital image data, perceptually insignificant distortions introduced to original images due to lossy compression or noisy transmission channels etc. are inevitable, when images are distributed via Internet. Therefore, it is required to guarantee that perceptually similar images have similar image hashes, and image hashing should be robust to such content-preserved un-malicious distortions and attacks for image identification and retrieval purpose [101]. An example is illustrated in Figure 1.1, which includes the original image and its distorted copies under distortions, such as Gaussian blurring, Gaussian noise, motion blurring, JPEG compression, rotation, cropping, and shearing. Perceptually, these images are identical in human visual system (HVS), while they indeed undergo some content-preserving distortions and attacks. The perceptual robustness of image hashing guarantees that these images have very close hashes, if the algorithms are robust enough against these attacks.

However, for malicious manipulations on the image content such as object insertion and removal, images hashes should be sensitive to these perceptually significant attacks for the image authentication purpose, which is related to the research field of image tampering detection and localization

[93].

• ***One-way Function***: Ideally, the hash generation should be non-invertible,

$$I \mapsto H_k(I). \tag{1.4}$$

• ***Unpredictability***: The signature is intractable without the secret key,

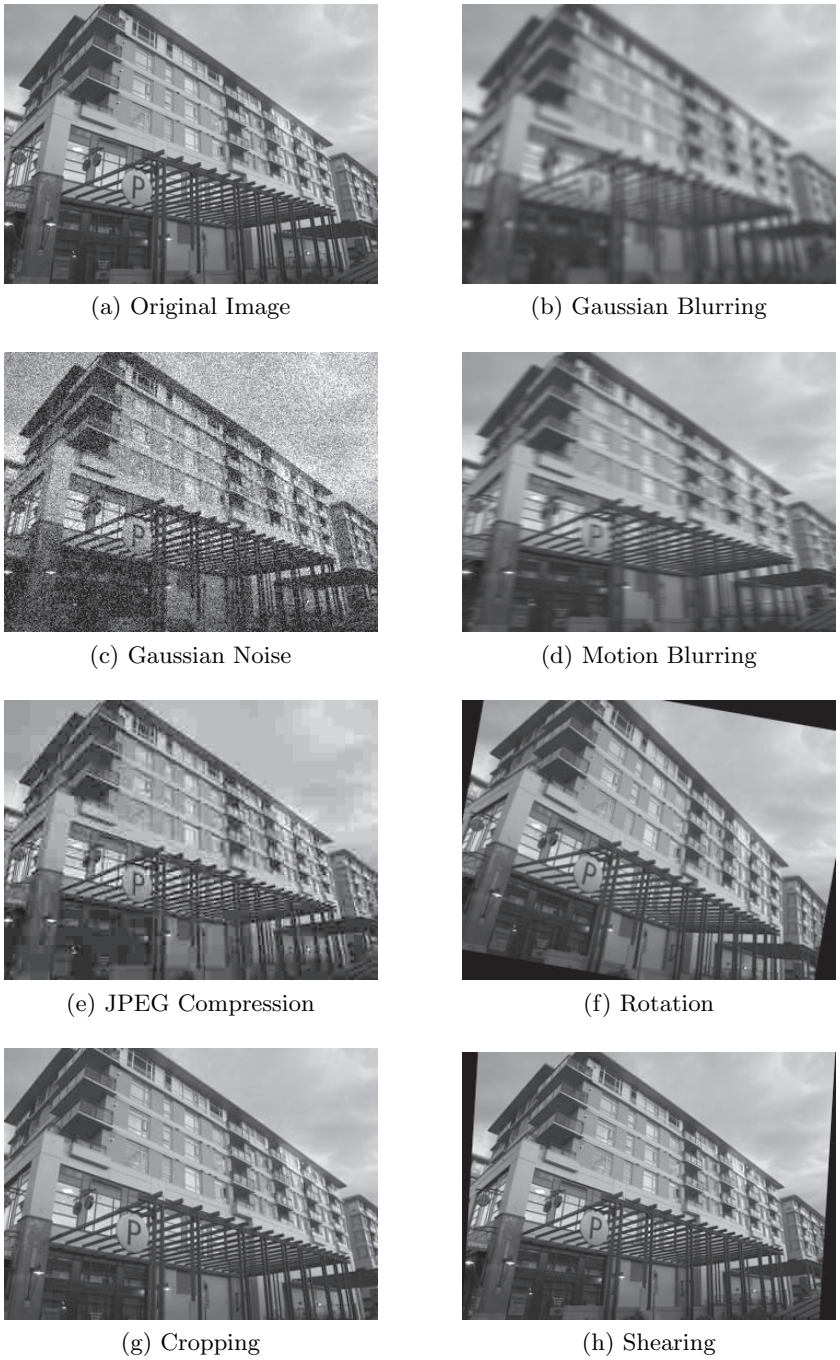
$$Pr(H_k(I) \neq H_{k'}(I)) \geq 1 - \delta,\ 0 \leq \delta < 1. \tag{1.5}$$

Similar to the traditional hashing, security is an important concern for image hashing. The property of one-way function guarantees the original image data is not accessible according to the corresponding hashes. On the other hand, pseudorandomization techniques are generally incorporated into image hash generation process to enhance the security of image hashes using secret keys and to prevent the unauthorized usage of digital images.

Ideally, all the above parameters ϵ, τ, and δ should be close to zero for a proper designed hashing scheme to generate unique and compact image hashes, which are robust enough against perceptually insignificant distortions and secure enough to prevent unauthorized access.

1.3.2 Generic Framework

A generic framework of image hashing/fingerprinting generation is shown in Figure 1.2. Generally, the robustness of image hashing arises from robust feature extraction and the compression component mainly contributes to the compactness of the final hash. To inherit the security of traditional hash functions and prevent unauthorized access, a secret key is incorporated into either feature extraction or compression or both to make image hashes unpredictable. To the best of our knowledge, most hashing algorithms incorporate the pseudorandomization relying on secret keys into the compression step, but some state-of-art hashing schemes [60, 74] also introduce pseudorandomizations into the feature extraction stage using random sampling or random projection to further enhance the security (as indicated by the dash line in Figure 1.2). Since secret keys are owned by owners, hash generation



(a) Original Image (b) Gaussian Blurring

(c) Gaussian Noise (d) Motion Blurring

(e) JPEG Compression (f) Rotation

(g) Cropping (h) Shearing

Figure 1.1: Examples of distorted image copies under different content-preserving attacks.

is a pseudorandom process rather than a completely random one. The incoming query hash corresponding to a specific query image will be compared with image hashes in the database for content identification, authentication and other applications. Since image hashes are compact signatures, the comparison could be performed efficiently.

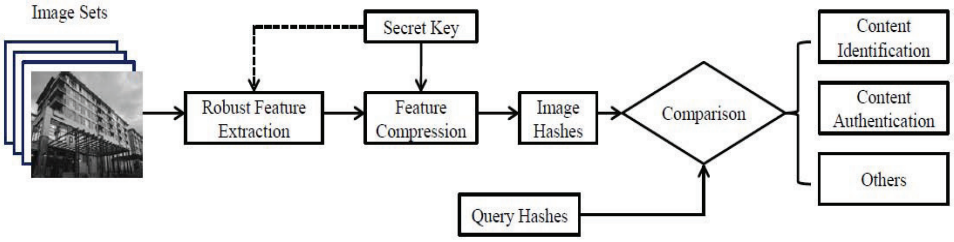


Figure 1.2: The generic framework of image hashing

1.4 Contributions and Thesis Outline

1.4.1 Thesis Contributions

Based on the generic framework of image hashing and the four desired properties: uniqueness, compactness, robustness, and security, it is obvious that robust feature extraction, feature compression and the incorporation of pseudorandomization are the key topics in designing desirable digital image hashing algorithms. In this thesis, we mainly focus on the robust feature extraction and feature compression issues and have the following contributions:

• I conduct a complete literature review on digital image hashing/content-based fingerprinting, analyze and compare various robust feature extraction and feature compression methods, and discuss the open issues and research directions of image hashing and content-based fingerprinting.

• I introduce a recently proposed low-distortion dimension reduction technique, referred as Fast Johnson-Lindenstrauss Transform (FJLT), and propose the use of FJLT for generating robust digital image hashes. To further improve the robustness of FJLT hashing (FJLTH) against rotation

attacks, I propose the rotation-invariant FJLT hashing (RI-FJLTH) by incorporating the Fourier-Mellin transform idea into FJLT hashing. Furthermore, the image hashing concept is extended to the content-based fingerprinting concept, which combines FJLTH and RI-FJLTH schemes to achieve superior identification performances under various distortions and attacks.

• I propose using the popular scale invariant feature transform (SIFT)[55] to detect robust feature points and incorporating the Harris criterion to select the most stable ones which are less vulnerable to image processing attacks. Then the shape contexts [9] are introduced into hash generation to represent the geometric distribution of the detected feature points. Experimental results show that the proposed image hashing scheme is robust to a wide range of distortions, especially against geometric attacks and illumination changes. Also, since the spatial structure of image content has been embedded into the hash, the proposed shape contexts based image hashes could be applied to detect and localize content tampering.

• I propose a binary image hashing compression scheme, which takes advantages of the extended hash feature space from virtual distortions and attacks and generates binary image hashes based on semi-supervised spectral embedding (SSE). The proposed scheme could compress real-valued intermediate hashes into binary image hashes, while preserving their robustness. Furthermore, it can be generalized to combine different types of real-valued image hashes and generate a fixed-length binary signature. The proposed binary image hashing shares the robustness of incorporated hashes, is more robust against various image distortions and is computationally efficient for image similarity comparison.

1.4.2 Thesis Outline

The thesis outline is summarized as follows:

Chapter 2 presents a literature survey of digital image hashing and content-based fingerprinting. Based on the proposed framework, the previous algorithms are analyzed according to four basic modules, which include pre-processing on images, robust feature extraction, feature compression,

and post-processing. Also, security of image hashing based on pseudo-randomization is also discussed. From the comprehensive analysis on the state-of-the art approaches, some open research issues and directions are discussed.

Chapter 3 presents a digital image hashing algorithm based on a recent dimension reduction technique, the Fast Johnson-Lindenstrauss Transform (FJLT). The popular Fourier-Mellin transform is further incorporated into the proposed FJLTH to improve its performance under rotation attacks. By combining FJLTH and rotation-invariant FJLTH (RI-FJLTH), content-based fingerprinting idea is proposed and demonstrated to yield superior robustness against various distortions and attacks, when compared with the state-of-art NMF image hashing.

Chapter 4 presents a novel digital image hashing algorithm based on robust SIFT-Harris feature point detection and shape context descriptors. The state-of-art SIFT for feature point detection is investigated under various image distortions and attacks. Based on the investigation, Harris criterion is incorporated to select the most stable SIFT key points under various distortions. When compared with another local feature point detection approach based on end-stopped wavelets, the proposed scheme is shown to be more robust against various distortions. Radial shape context hashing (RSCH) and angular shape context hashing (ASCH) schemes are proposed by embedding the detected SIFT-Harris feature points into shape context descriptors in radial and angular directions respectively. The proposed SCH is shown to be more robust than FJLTH, RI-FJLTH, and NMFH under rotation attacks and illumination changes. Also, by combing both RSCH and ASCH, more robustness can be achieved and its application on image tampering detection is demonstrated for image authentication purpose.

Chapter 5 presents a novel binary image hashing compression algorithm using semi-supervised spectral embedding (SSE). With the availability of real-valued intermediate image hashes, the extended hash feature space under virtual prior attacks is generated and a training is introduced to learn the spectral embedding based on a given cost function, which is specifically designed to both preserve local similarity between image hashes from distorted

images and distinguish hashes from distinct images. Based on the learned embedding, real-valued intermediate image hashes could be projected into binary image hashes using out-of-sample extensions. The generated binary image hashes are more robust when compared with the ones using traditional quantization-based compression methods. Furthermore, the proposed SSE scheme is extended to combine multiple real-valued intermediate image hashes and embed them into fixed-length binary hashes, which are demonstrated to be more robust and more computationally efficient for similarity measures using Hamming metrics.

Chapter 6 concludes the dissertation and discusses the future work.

Chapter 2

Image Hashing and Content-Based Fingerprinting Review

2.1 Review Framework

In this chapter, previous works on digital image hashing and content-based fingerprinting approaches in last decades are reviewed, following the key issues of the framework shown in Figure 2.1, which includes pre-processing, robust feature extraction, feature compression, post processing, and security incorporation. These issues summarize the major components that are critical to design a robust and secure digital image hashing algorithm.

After the review on the algorithm design, the evaluation criteria are discussed based on different application scenarios. For instance, if the image hashing algorithm is designed for content identification, its robustness against content-preserving attacks that do not introduce obviously perceptual manipulations on image content is usually the major concern to be evaluated. Other desired evaluation criteria that people usually adopt are also shown in Figure 2.1 for corresponding application scenarios. Note that there is one dash lines in Figure 2.1, indicating that the measure is desired but not necessary. For instance, some digital image hashing algorithms are mainly designed for image copies detection, and thus focus mainly on the robustness analysis without taking the security into consideration. However, for the content authentication purpose, how to protect the integrity of images is the critical goal, which means that image hashes or fingerprints

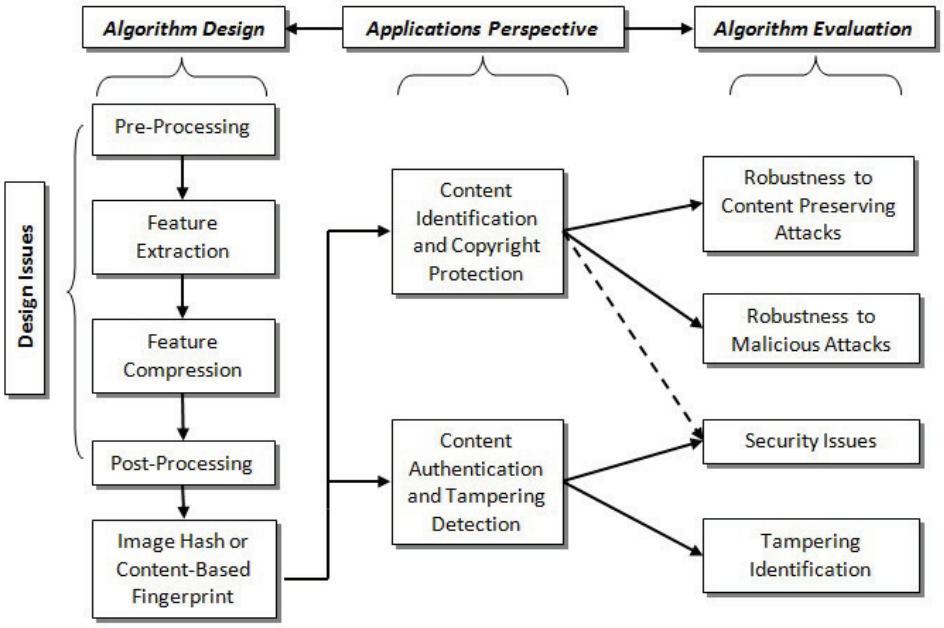


Figure 2.1: The framework of digital image hashing and content-based fingerprinting review

should be secure enough to prevent forgery attacks. Therefore the security issues have to be taken into consideration.

The papers cited in this review mainly include the following cases: First, the paper presents a complete hashing or fingerprinting design, which means that robust and compact hashes are generated to represent digital images, and security issues for image hash protection are also considered. Secondly, the paper focuses only on robust feature extraction and fast matching without considering security issues, but it still generates compact signatures to represent images and aims to retrieve distorted copies under content-preserving attacks. The reason we include this type of papers is that robust feature extraction is one of the most important modules in the design of image hashing. The robust feature descriptors could be treated as intermediate hashes and further encrypted in the post processing stage to generate secure signatures later. Thirdly, the paper mainly analyzes the security issues of image hash functions, including the unpredictability arising from pseudo-

randomization or fragility to some sophisticate attacks. Finally, the papers that evaluate or compare the performances of state-of-art image hashing or content-based fingerprinting schemes are also covered.

In order to exclusively focus on the topics we choose and do a comprehensive review on the related literatures, we mainly follow two ways to search the papers. First, we use some key words to search papers related to the specific topics from Google Scholar, including: "image hashing", "content-based image fingerprinting", "image digital signature" and so on. This is a rough way to obtain some popular papers with high citations. Secondly, we find other related papers following the references and citations. Within the literature survey, we also found some authors extended their valuable ideas from conference papers to journal papers. To avoid the redundancy, we only cite their journal papers when available, because journal papers usually illustrate more details about the proposed ideas, provide more experimental results with detailed analysis, and could better facilitate other people's work in the future.

2.2 Pre-Processing

The pre-processing step is a general way to "filter" the image content before the robust feature extraction step. Its major purpose is to enhance the robustness of features by preventing the effects of some distortions, such as additive noise. Some works take advantages of the pre-processing step to normalize images into a standardized format, which could facilitate the robust feature extraction step. The common pre-processing operations applied on digital images are illustrated as follows and the related references are listed in Table 2.1.

• *Colour Space Dimension Reduction*: This is a common operation applied in most digital image hashing algorithms. Colour images are first converted to grayscale images to reduce the computational cost for feature extraction (e.g. 3D to 2D). Another way is based on the colour space transform that converts RGB space to HSI space. Then the 2D digital image data in the illumination channel is further used for feature extraction.

Table 2.1: Literature references regarding pre-processing

Methods	**References**
Colour Space Dimension Reduction	[4, 14, 27, 40, 43, 47, 48, 52, 54, 56, 57, 60, 67, 70, 73, 74, 78, 81, 84, 88, 92, 96, 100, 104]
Resizing	[16, 31, 60, 70, 73, 84, 90, 92]
Filtering	[80, 84, 92, 103]
Illumination Normalization	[27, 84, 92]

• *Resizing*: Images are resized to a predefined size (usually very small, e.g. 256×384) as a default format. The advantages are twofold: First, the computational cost of feature extraction is much lower. In this sense, it improves the efficiency of hash generation and facilitates fast indexing and retrieval applications. Also, features extracted from an image with standardized size are more robust against the aspect ratio change, which is one of the geometric attacks.

• *Filtering*: It is an efficient way to improve the robustness of the extracted features against noise. Some popular filters, such as median filter and Gaussian filter, could be applied on digital images for noise reduction. However, these low-pass filters would also eliminate some details of image contents and generate blurred images, and thus require that the image hashing scheme is robust against blurring distortions.

• *Illumination Normalization*: Brightness change or Gamma correction is a common image processing attack. Illumination normalization processes such as histogram equalization could effectively render the extracted features invariant to illumination changes.

In the literatures, there are other pre-processing operations which can serve as assistant ways for extracting robust features, such as image registration [93], ellipse block partition [102] to achieve rotation invariance, and so on.

2.3 Robust Feature Extraction

Robust feature extraction is one of the fundamental modules in digital image hashing and fingerprinting algorithms. Image hashes are unique, since the extracted features are based on the characteristics of digital images, which are distinctive enough for content identification. On the other hand, as long as two digital images are perceptually identical, the extracted features and thus the image hashes should be as similar as possible even the images are under additive noise, blurring, geometric attacks and other content-preserving attaches. By reviewing the related literature in last decades, we note that most previous works focus on seeking robust features to resist certain distortions and attacks, which are summarized as follows and listed in Table 2.2.

• *Image Pixels and Statistics*: Image pixel values are the raw features that could be directly used for hash generation. However, an $N \times N$ image will have a feature vector with length N^2, which can be quite high dimensional. Therefore a dimension reduction technique that could preserve the local similarity is desired for feature compression in image hashing. Furthermore, the statistics of pixel values can also be applied as robust features for image hash generation, such as mean, variance, and other higher moments such as skewness and kurtosis. The statistic features are usually more robust than the raw pixel values against noising, blurring, and compression distortions, but with less distinctiveness, which is significantly important for the uniqueness of image hashes. Thus tradeoffs between different desired properties have to be taken into consideration in image hashing approaches.

• *Invariant Feature Transform*: Coefficients in a transformed domain can be critical features and robust enough against a large class of image processing attacks and distortions. The state-of-art transforms to extract robust features include Discrete Cosine Transform (DCT), Discrete Wavelet Transform (DWT), Fourier-Mellin Transform, Radon Transform etc. Kim proposed using ordinal measures of DCT coefficients to generate robust descriptors against histogram equalization, noise addition etc. for the copy detection purpose [43]. This method was further improved for rotation ma-

nipulations in [102]. The Radon Soft Hash algorithm (RASH) [47] shows robustness against small geometric transformation and common image processing distortions such as JPEG compression by employing radon transform to extract geometrically invariant features. Swaminathan's hashing scheme [92] applies Fourier-Mellin transform to extract invariant features to achieve better robustness against geometric attacks. The major advantages of the features in a transformed domain lie in their strong robustness against certain distortions and attacks. However, generally such robustness is at the cost of vulnerability to other types of attacks. For instance, the features from the Fourier-Mellin transform domain are robust against geometric transforms, but they are sensitive to noise addition and blurring attacks.

• *Local Feature Patterns*: Local feature patterns are one kind of important robust features for generating image hashes. Local feature patterns usually include edges, corners, blobs, salient regions and so on. Monga *et.al* [73] proposed an image hashing method using local feature points detected by end-stopped wavelet, which are robust enough under various geometric attacks, including rotation, cropping, shearing etc. Roy *et.al* [80] proposed a robust local preserving hashing algorithm based on scale invariant feature transform (SIFT) [55] to identify image tampering for the authentication purpose. Obviously, the benefit of applying local feature patterns mainly lies in their robustness against geometric attacks, which is often the bottleneck of other classical features. However, the sensitivity to noise addition, blurring, and compression limits their practical developments in image hashing, and it remains an open research issue.

• *Matrix Decomposition*: Since images are inherently 2D matrix data (e.g. grayscale images), matrix decomposition methods are also explored for extracting invariant features. For instance, using low-rank matrix approximations obtained via singular value decomposition (SVD) for hashing is explored in [45], and non-negative matrix factorization (NMF) with its non-negativity constraints is also applied for generating robust image hashes in [74]. The invariant features based on matrix decomposition show good robustness against noise addition, blurring and compressing attacks. However,

Table 2.2: Literature references regarding robust feature extraction

Robust Features	Sub-categories	References
Image Pixels and Statistics	*Pixels*	[27, 52, 60, 74, 83, 93, 103, 104]
	Mean	[18, 39, 43, 44, 54, 78, 96, 102–104, 110]
	Variance	[11, 39, 96, 102]
	Others	Color [28, 69], Cumulant [109]
Invariant Feature Transform	*DCT*	[34, 40, 43, 45, 49, 56, 109]
	DWT	[11, 16, 29, 31, 32, 45, 56, 57, 67, 69, 70, 75, 77, 96, 100, 108]
	Fourier Transform	[24, 27, 60, 78, 80, 90, 92]
	Radon Transform	[31, 47, 48, 84, 100]
	Others	Trace Transform [14], Gabor Filter [69]
Local Feature Pattern		[11, 24, 27, 56, 73, 78, 80, 81, 106, 110]
Matrix Decomposition		[45, 74, 94]

their performances under large geometric attacks are still limited.

Robust feature extraction is a key step in image hashing due to the critical robustness concern. Based on the literature review and our own study, we feel that seeking certain features to resist all types of image distortions and attacks are highly unlikely, while it is relatively easier to find a specific kind of features to be robust against certain attacks.

2.4 Feature Compression and Post-Processing

The compactness is another critical property of image hashing. Hence, robust features need to be further compressed into a short real-valued or even binary sequence, which can essentially be considered as a dimension reduction process. However, to obtain a very short image hash, it inevitably

becomes a lossy compression process, and how to preserve the local similarity from a higher dimensional space to a lower dimensional space is always a critical research challenge. Some typical methods are summarized as follows and related references are listed in Table 2.3.

• *Quantization*: It is widely employed for converting continues feature space to finite discrete feature space and helpful for further signature encoding. Popular approaches include interval quantization, binary quantization using threshold, ordinal measures and so on for image hash generation.

• *Compression Coding*: The state-of-art coding techniques in communications can also be applied for compressing the robust features into short image hashes, including Distributed source coding (e.g. Wyner-Ziv, Slepian-Wlof), LDPC coding and error-correcting coding (ECC) etc.

• *Random Projection*: It is one of the state-of-art dimension reduction techniques to project data in a high dimensional space into a lower dimensional space, while preserving the local similarity of the data. The random projection approach can result in performances comparable to that of the conventional dimension reduction methods such as Principal Component Analysis (PCA), but be computationally more efficient. Another advantage of the random projection direction is that the projection is a pseudorandomization process that could enhance the security of the designed hashing scheme.

• *Clustering*: Clustering methods that divide the feature space into finite voxels given defined distance metrics and map similar features into the same centroids of clusters can also be employed in image hashing for feature compression. However, when new original images are registered in the database, clustering-based methods could only map them into existing clusters unless the model is retrained. Therefore the computational cost of clustering-based methods is generally higher than other types of methods.

• *Traditional Cryptography*: Conventional techniques in cryptography can sometimes be used for generating short image hashes based on strong robust features, since they are sensitive to minor distortions of digital images. Popular techniques include RSA, MD-5, DES, and so on.

Table 2.3: Literature references regarding feature compression and post-processing

Methods	References
Quantization	[3, 14, 18, 24, 27, 28, 31, 37, 39, 40, 43, 44, 49, 52, 54, 56, 57, 67, 69, 70, 73, 78, 80, 81, 84, 90, 92, 94, 96, 103, 108]
Compression Coding	[24, 29, 37, 52, 53, 66, 77, 92, 93, 96]
Random Projection	[32, 52, 53, 59, 60, 74, 80, 92, 93]
Clustering	[39, 43, 72, 81]
Traditional Cryptography	[11, 49, 54, 57, 104]

2.5 Security Incorporation

Security is one of the most important properties of image hashing or content-based fingerprinting due to its application on copyright protection. To make image hashes unpredictable, the basic idea is to incorporate a "secret key" into hash generation to make it as a pseudorandomization process. Ideally, for a secure image hashing scheme, users can't generate or even forge the right hash of an image unless with the help of the corresponding "secret key". Hence, the incorporation of the "secret key" is inherently an encryption process in cryptography. The typical way of security incorporation is summarized as follows and related references are listed in Table 2.4.

• *Randomized Tiling*: Images are randomly partitioned into overlapped subregions based on the selected "secret key". These subregions could be rectangle, circle, or even ellipse with randomly selected radii. Then, robust features could be extracted from these randomized areas to enhance the security of the final image hashes. Randomized tiling is usually applied in the pre-processing step and an effective way to make the final hashes unpredictable. The only bottleneck is its sensitivity to geometric attacks.

• *Randomized Transform*: After robust features are extracted, they are further transformed into another randomized domain determined by the selected "secret key". It is inherently a pseudo-encryption process applied

Table 2.4: Literature references regarding security incorporation

Methods	References
Randomized Tiling	[45, 60, 66, 67, 70, 73, 74, 94, 96]
Randomized Transform	[4, 27, 40, 52, 60, 66, 84, 94]
Random Projection	[24, 32, 44, 52, 54, 60, 70, 74, 78, 81, 83, 92, 93, 96, 103]
Traditional Cryptography	[11, 50, 57, 104]

in feature extraction step to make the features unpredictable.

• *Random Projection* : It is usually applied in feature compression and post-processing step to project robust features into a lower dimension based on the projection matrix, whose entries are random variables determined by the selected "secret key". One typical example is the Gaussian random projection [74].

• *Traditional Cryptography*: The conventional techniques in cryptography could also be employed for encrypting features after feature compression step. Although they are sensitive to the minor changes of encrypted data, it is still feasible as long as the features are robust enough.

With the help of "secret key", the encryption of image hashes could be controlled by the authorized users and prevent the unauthorized access, which facilitates the application on copyright protection.

2.6 Comparison and Decision Making

Following the framework of image hashing or content-based fingerprinting, a compact and secure hash is generated and associated with the corresponding original image in database as an index. When a query hash is received, it will be compared with the existing hashes based on the selected distance metrics and the corresponding image will be retrieved according to the classifiers. Hence, the distance metrics to measure the similarity between hashes and the classifiers to make decisions are also two important issues in hashing and fingerprinting schemes.

2.6.1 Distance Metrics

Given two hashes $H_1 = \{h_1(1), h_1(2), ..., h_1(k)\}$ and $H_2 = \{h_2(1), h_2(2), ..., h_2(k)\}$ of two images I_1 and I_2 with length k, the following distance metrics are usually employed:

$$\textit{Euclidean Distance}: \qquad Dist(H_1, H_2) = \sqrt{\sum_{i=1}^{k}(h_1(i) - h_2(i))^2} \tag{2.1}$$

$$\textit{L1 Norm}: \qquad Dist(H_1, H_2) = \sum_{i=1}^{k}|h_1(i) - h_2(i)| \tag{2.2}$$

$$\textit{Hamming Distance}: \qquad Dist(H_1, H_2) = \sum_{i=1}^{k}|h_1(i) \oplus h_2(i)| \tag{2.3}$$

The choice of distance metrics depends on the type of hashes. When the generated hashes are real-valued vectors, Euclidean distance or L1 norm is usually employed. Otherwise, Hamming distance should be used for binary hashes. For the comparison and retrieval in large database, binary hashes and Hamming metrics are preferable for the lower computational cost, while real-valued hashes and Euclidean distance or L1 norm provide higher identification accuracy with the cost of more computational burden.

2.6.2 Classifiers

After the similarity between hashes is measured by the selected distance metrics, classifiers are employed to make the decision for content identification. In most image hashing and fingerprinting algorithms, the simple nearest neighbour classifier or threshold based classifiers are usually used for making decision.

$$Dist(H_1, H_2) \leq \xi \tag{2.4}$$

where ξ is the selected threshold. Although there are a lot of advanced classification methods proposed in machine learning, they are seldom employed in image hashing area. The underlying reason is as follows: Image hashing is an infinite clustering problem, which takes each original image as a new cluster and all its perceptually identical copies are assumed to lie in the neighbourhood of the centroid (e.g. the original image). Hence, if advanced supervised classifiers, such as Support Vector Machine (SVM), are employed, they could only deal with the finite classification problems and have to be re-trained, whenever a new original image is registered in database. The re-training process may incur heavy computational cost when thousands of images are registered and training advanced classifiers to deal with classification for infinite classes is not feasible in practice. One attempt to explore the classification using SVM and Gaussian Mixture Model (GMM) for image copy detection is presented in [34]. However, the method could only deal with the finite classes and the model has to be retrained for new original images.

2.7 Evaluation Criteria

Since the most important properties of image hashing and content-based fingerprinting are the robustness and security, most existing works focus on evaluating these two issues to make the proposed schemes stand out.

2.7.1 Perceptual Robustness

Different from traditional hashing in cryptography, image hashing shouldn't suffer from the sensitivity to minor distortions of images due to the perceptual robustness. Hence, two images that are perceptually identical in human visual system (HVS) should have similar hashes. To evaluate the robustness of image hashes, a large class of distortions and attacks are designed, which could be roughly grouped as content-preserving attacks and malicious attacks.

- *Content-preserving Attacks*: The manipulations only introduce distor-

Table 2.5: Literature references regarding content-preserving attacks part one

Attacks&Distortions	**Sub-categories**	**References**
Noise Addition	*Gaussian Noise*	[14, 24, 25, 28, 31, 34, 43, 44, 56, 57, 60, 65, 67, 70, 73, 92, 94, 100, 102–104, 106, 108]
	Salt &Pepper Noise	[25, 28, 60]
	Speckle Noise	[25, 28, 60]
	Uniform Noise	[27, 31, 92, 103]
Filtering	*Gaussian Filter*	[4, 14, 25, 27, 31, 34, 37, 39, 47, 48, 56, 57, 60, 69, 73, 77, 84, 94, 103, 106, 108–110]
	Median Filter	[14, 25, 27, 28, 31, 34, 37, 39, 44, 56, 70, 73, 84, 92, 96, 100, 103, 106, 108–110]
	Average Filter	[47, 65, 92]
	Wiener Filter	[25, 44, 81, 92, 103]
	Sharpen Filter	[4, 27, 34, 37, 44, 48, 56, 57, 70, 77, 84, 92, 106, 109, 110]
	Motion Filter	[25, 43, 60, 102, 110]

tions on pixel level and the content of images perceptually remains the same in semantic level. The examples include noise addition, filtering, compression, geometric transforms, brightness changes etc. The details and related literatures are listed in Table 2.5 and Table 2.6.

• *Malicious Attacks*: These manipulations introduce small but significant visual changes in images, such as adding or removing small objects in image content etc. The details and related literatures are listed in Table 2.7.

Dealing with two major kinds of attacks, image hashing should be generally robust against content-preserving attacks for content identification and copyright protection purpose, but sensitive to malicious attacks if the

Table 2.6: Literature references regarding content-preserving attacks part two

Attacks&Distortions	**Sub-categories**	**References**
Geometric Attacks	*Rotation*	[14, 18, 25, 28, 31, 32, 34, 37, 39, 43, 45, 47, 56, 60, 69, 70, 73–75, 77, 81, 84, 90, 92, 93, 96, 100, 102, 103, 106, 108, 110]
	Cropping	[25, 28, 32, 34, 37, 45, 47, 56, 60, 69, 70, 73–75, 77, 81, 84, 90, 92, 93, 96, 102, 103, 106, 108–110]
	Scaling	[14, 24, 25, 28, 32, 34, 37, 47, 48, 56, 60, 65, 69, 70, 73, 75, 77, 84, 90, 92–94, 96, 100, 103, 106, 108–110]
	Shearing	[25, 70, 73, 74, 77, 81, 92, 103, 108, 109]
	Aspect Ratio	[18, 43, 56, 77, 102, 108, 110]
	Affine Transform	[34, 56, 74, 77, 96, 100, 102, 106]
	Others	Bending [56, 70, 73, 84, 103, 108, 109], Jittering [103]
Compression	*JPEG*	[3, 11, 18, 24, 25, 27–29, 31, 32, 34, 39, 44, 45, 47, 49, 52–54, 56, 60, 65, 67, 70, 74, 78, 81, 83, 84, 92–94, 96, 100, 102–104, 106, 108–110]
	JPEG2000	[29, 52, 53, 103]
Brightness Changes		[24, 25, 27, 28, 31, 34, 43, 44, 56, 60, 65, 67, 69, 77, 81, 90, 92, 93, 106, 108, 110]
Contrast Enhancement		[27, 43, 57, 69, 70, 73]

Table 2.7: Literature references regarding on malicious attacks

Methods	References
Adding Objects	[11, 24, 31, 43, 52, 54, 57, 66, 67, 73, 78, 90, 92, 93, 102]
Removing Objects or Lines	[24, 29, 34, 37, 39, 54, 56, 73, 81, 83, 84, 92, 96, 104, 105, 108, 109]
Manipulating Contents	[24, 29, 31, 54, 66, 73]

hashing is designed for content authentication purpose.

2.7.2 Security Analysis

Aside from the robustness analysis, security is also another important issue in image hashing design. a secure image hashing scheme means that the generated hashes are hardly predictable or forged without the knowledge about "secret key". The most works in last decades mainly evaluate security issue in two ways: one is the unpredictability and the other is to design certain sophisticate schemes to test the possibility of forgery.

- *Unpredictability*: It mainly focuses on evaluating the randomness of hash values, assuming adversaries knows the hashing algorithm without the knowledge of "secret key". The state-of-the art way is to use differential entropy [92] as a metric to evaluate the amount of hash randomness. However, some researchers stated that the secure hashes should have high differential entropy but not vice versa [74]. Later, the unicity distance concept in information theory [64] is employed to quantify the number of key reuses need for key estimation as a measure to evaluate the security. Other works, such as [4, 66], discuss key dependency of hash values. The change in the "secret key" should significantly change the hashes. Also, the size of key space is an indicator for evaluating the security of hashes. Recently, mutual information [37, 42] is adopted as a measure of information leakage. Mutual information could be used to measure the amount of uncertainty of pseudo randomness incorporated in hash generation, given the extracted features and final hashes.
- *Sophisticate Attacks*: It is an opposite way to evaluate the security of

hashing scheme compared with the unpredictability analysis. People design some sophisticate attacks to test whether an image hashing scheme is secure enough. The two popular attacks are collision attacks [4, 99] and forgery attacks [100]. Collision attack is trying to generate similar hashes for two images, which are partially similar but with noticeable difference. If the probability of hash collision is very high, it means the hash couldn't be used for authenticate image content, and attackers could easily get through the security check and modify the image content. Forgery attack is trying to generate a forged image with the similar hash as the target original image based on some optimization methods, such as hill-climbing. By minimizing the difference of the hashes from original and forged image, attackers adjust the forged image and generate forged hashes to approximate real hashes. Obviously, the sophisticate attacks are subjective ways to evaluate the security of image hashing and mainly designed for testing the possibility for authentication purpose.

However, both unpredictability measures and sophisticate attack analysis are only capable of revealing parts of the security of image hashing and couldn't be used as a universal measure to comprehensively evaluate the security, which is still a critical open research area in the future.

2.8 Conclusion

In this chapter, a comprehensive review on digital image hashing and content-based fingerprinting is presented and most of the closely related literature references are listed and briefly discussed. Base on the fundamental framework in Figure 2.1 shown in this chapter, the key issues, including pre-processing, robust feature extraction, feature compression, post-processing, security incorporation and evaluation criteria, are respectively discussed in details.

It is clear that the robustness of image hashing mainly arises from the pre-processing and robust feature extraction steps. However, generating compact image hashes and preserving the robustness of features mainly rely on the effective feature compression and post-processing steps. There usu-

ally exists a tradeoff between the robustness and compactness, since the compression step embeds features from high dimensional spaces into very lower ones, which always incurs information loss. The security property makes digital image hashing schemes different from the conventional CBIR schemes. It is inherently a pesudorandomization process controlled by a "secret key" and could be incorporated into feature compression and post-processing steps or even the feature extraction step. Although there sill lacks of universal measures for evaluating the security of digital image hashing schemes, the high unpredictability of image hashing has made it an attractive, promising technique for copyright protection and image content authentication.

Chapter 3

Image Hashing and Content-Based Fingerprinting Based on Fast Johnson-Lindenstrauss Transform (FJLT)

3.1 Introduction

Recently, several image hashing schemes based on dimension reduction have been developed and reported to outperform previous techniques. For instance, using low-rank matrix approximations obtained via singular value decomposition (SVD) for hashing was explored in [45]. Its robustness against geometric attacks motivated other solutions in this direction. Monga introduced another dimension reduction technique, called non-negative matrix factorization (NMF) [85], into their new hashing algorithm [74]. The major benefit of NMF hashing is the structure of the basis resulting from its non-negative constraints, which leads to a parts-based representation. In contrast to the global representation obtained by SVD, the non-negativity constraints result in a basis of interesting local features [30]. Based on the results in [74], the NMF hashing possesses excellent robustness under a large class of perceptually insignificant attacks, while it significantly reduces misclassification for perceptually distinct images.

Inspired by the potential of dimension reduction techniques for image

hashing, we introduced Fast Johnson-Lindenstrauss transform (FJLT), a dimension reduction technique recently proposed in [5], into our new robust and secure image hashing algorithm [59]. FJLT shares the low-distortion characteristics of a random projection process but requires a lower computational complexity. It is also more suitable for practical implementation because of its high computational efficiency and security due to the random projection. Since we mainly focus on invariant feature extraction and are interested in image identification applications, the FJLT hashing seems promising because of its robustness to a large class of minor degradations and malicious attacks. Considering the fact that NMF hashing was reported to significantly outperform other existing hashing approaches [74], we use it as the comparison base for the proposed FJLT hashing. Our preliminary experimental results in [59] showed that FJLT hashing provides competitive or even better identification performance under various attacks such as additive noise, blurring, JPEG compression etc. Moreover, its lower computational cost also makes it attractive.

However, geometric attacks such as rotation, could essentially tamper the original images and thus prevent the accurate identification if we apply the hashing algorithms directly on the manipulated image. Even for the FJLT hashing, it still suffers from the rotation attacks with low identification accuracy. To address this concern, motivated by the work [92], [51], we plan to apply the Fourier-Mellin transform (FMT) on the original images first to make them invariant to geometric transform. Our later experimental results show that, under rotation attacks, the FJLT hashing combined with the proposed FMT preprocessing yields a better identification performance than that of the direct FJLT hashing.

Considering that a specific feature descriptor may be more robust against certain types of attacks, it is desirable to take advantage of different features together to enhance the overall robustness of hashing. Therefore we further propose an extended concept, namely content-based fingerprinting, to represent a combined, superior hashing approach based on different robust feature descriptors. Similar to the idea of having the unique fingerprint for each human being, we aim at combining invariant characteristics of each feature to

construct an exclusive (unique) identifier for each image. Under the framework of content-based fingerprinting, the inputs to the hashing algorithms are not restricted to the original images only, but can also be extendable to include various robust features extracted from the images, such as colour, texture, shape and so on. An efficient joint decision scheme is important for such a combinational framework and significantly affects the identification accuracy. Our experimental results demonstrate that the content-based fingerprinting using a simple joint decision scheme can provide a better performance than the traditional onefold hashing approach. More sophisticated joint decision-making schemes are worth further being investigated in the future.

3.2 Theoretical Background

The current task of image hashing is to extract more robust features to guarantee the identification accuracy under content-preserving manipulations (e.g. noising, blurring, compression etc.) and incorporate the pseudo-randomization techniques into the feature extraction to enhance the security of the hash generation. According to the information theory [21], if we consider the original image as a source signal, similar to a transmission channel in communication, the feature extraction process will make the loss of information inevitable. Therefore, how to efficiently extract the robust features as lossless as possible is a key issue that the hashing algorithms such as SVD [45], NMF [74] and our FJLT hashing want to tackle.

3.2.1 Fast Johnson-Lindenstrauss Transform

The Johnson-Lindenstrauss (JL) theorem has found numerous applications, including searching for approximate nearest neighbors (ANN) [5] and dimension reduction in database etc. By the JL lemma [22], n points in Euclidean space can be projected from the original d dimensions down to lower $k = \mathcal{O}(\varepsilon^{-2} \log n)$ dimensions while just incurring a distortion of at most $\pm\varepsilon$ in their pairwise distances, where $0 < \varepsilon < 1$. Based on the JL theorem,

Alion and Chazelle [5] proposed a new low-distortion embedding of l_p^d into l_p^k ($p = 1$ *or* 2), called Fast Johnson-Lindenstrauss transform (FJLT). FJLT is based on preconditioning of a sparse projection matrix with a randomized Fourier transform. Note that we will only consider the l_2 case ($p = 2$) because our hash is measured by the l_2 norm. For the l_1 case, interested readers please refer to [5].

Briefly speaking, FJLT is a random embedding, denoted as $\Phi = FJLT(n, d, \varepsilon)$, that can be obtained as a product of three real-valued matrices:

$$\Phi = P \cdot H \cdot D \tag{3.1}$$

where the matrices P and D are random and H is deterministic [5].

- P is a k-by-d matrix whose elements P_{ij} are drawn independently according to the following distribution, where $\mathcal{N}(0, q^{-1})$ means a Normal distribution with zero-mean and variance q^{-1},

$$\begin{cases} P_{ij} \sim \mathcal{N}(0, q^{-1}) & \text{with probability} \quad q, \\ P_{ij} = 0 & \text{with probability} \quad (1 - q), \end{cases}$$

  where

$$q = \min\left\{\frac{c \log^2 n}{d}, 1\right\},$$

  for a large enough constant c.

- H is a d-by-d normalized Hadamard matrix with the elements as:

$$H_{ij} = d^{-\frac{1}{2}}(-1)^{\langle i-1, j-1\rangle}, \tag{3.2}$$

  where $\langle i, j\rangle$ is the dot-product of the m-bit vectors of i, j expressed in binary.

- D is a d-by-d diagonal matrix, where each diagonal element D_{ii} is drawn independently from {-1,1} with probability 0.5.

Therefore, $\Phi = FJLT(n, d, \varepsilon)$ is a k-by-d matrix, where d is the original dimension number of the data and k is the lower dimension number, which is

set to be $c'\varepsilon^{-2}\log n$. Here, n is the number of data points, ε is the distortion rate, and c' is a constant. Given any data point X from a d-dimension space, it is intuitively mapped to the data point X' at a lower k-dimension space by the FJLT and the distortion of their pairwise distances could be illustrated by Johnson-Lindenstrauss lemma [5].

3.2.2 The Fast Johnson-Lindenstrauss Lemma

Lemma 1 *Fix any set X of n vectors in $\mathbb{R}^d$, $0 < \varepsilon < 1$, and let $\Phi = FJLT(n, d, \varepsilon)$. With probability at least $\frac{2}{3}$, the following two events occur:*

1. *For all $x \in X$,*

$$(1-\varepsilon)k\|x\|_2 \leq \|\Phi x\|_2 \leq (1+\varepsilon)k\|x\|_2. \tag{3.3}$$

2. *The mapping $\Phi : \mathbb{R}^d \to \mathbb{R}^k$ requires*

$$\mathcal{O}(d\log d + \min(d\varepsilon^{-2}\log n, \varepsilon^{-2}\log^3 n)) \tag{3.4}$$

operations.

Proofs of the above theorems can be found in [5]. Note that the probability of being successful (at least $\frac{2}{3}$) arises from the random projection and could be amplified to $(1-\delta)$ for any $\delta > 0$, if we repeat the construction $\mathcal{O}(\log \frac{1}{\delta})$ times [5]. Since the random projection is actually a pseudorandom process determined by a secret key in our case, most of the keys (at least $\frac{2}{3}$) are satisfied with the distortion bound described in FJLT lemma and could be used in our hashing algorithm. Hence, the FJLT will make our scheme widely applicable for most of the keys and suitable to be applied in practice.

3.3 Image Hashing via FJLT

Motivated by the hashing approaches based on SVD [45] and NMF [74], we believe that dimension reduction is a significantly important way to capture

the essential features that are invariant under many image processing attacks. For FJLT, three benefits facilitate its application in hashing. First, FJLT is a random projection, enhancing the security of the hashing scheme. Second, FJLT's low distortion guarantees its robustness to most routine degradations and malicious attacks. The last one is its low computation cost when implemented in practice. Hence, we propose to use FJLT for our new hashing algorithm. Given an image, the proposed hashing scheme consists of three steps: random sampling, dimension reduction by FJLT, and ordered random weighting. Due to our purpose, we are only interested in feature extraction and randomization. The hash generated by FJLT is just an intermediate hash. For readers who are interested in generating the final hash by compression step, as in the frameworks[92], [73], they are suggested to refer [37, 96] for details.

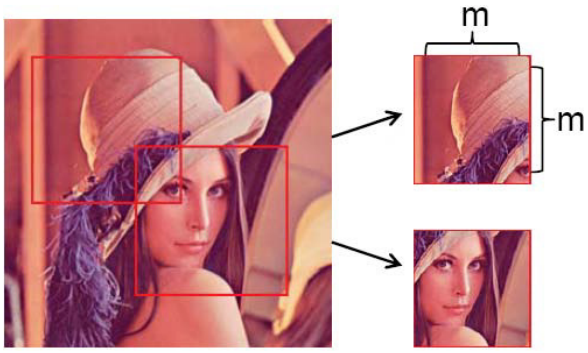


Figure 3.1: An example of random sampling. The subimages are selected by random sampling with size $m \times m$.
.

3.3.1 Random Sampling

The idea of selecting a few subimages as original feature by random sampling, as shown in Figure 3.1, is not novel [45], [74]. However, in our approach, we treat each subimage as a point in a high dimensional space rather than a two dimensional matrix as in SVD hashing [45] and NMF hashing [74]. For instance, the subimage in Figure 3.1, which is a m-by-m patch, is

actually a point in the m^2-dimensional space in our case, where we focus on gray images.

Given an original color image, we first convert it to a gray image and pseudorandomly select N subimages depending on the secret key and get $\{R_i\}$, for $1 \leq i \leq N$. Each R_i is a vector with length m^2 by concatenating the columns of the corresponding subimage. Then we construct our original feature as:

$$Feature = \{R_1, R_2, ..., R_N\} \quad \text{, with size} \quad m^2 \times N. \tag{3.5}$$

The advantage of forming such a feature is that we can capture the global information in the $Feature$ matrix and local information in each component R_i. Even if we lose some portions of the original image under geometric attacks such as cropping, it will only affect one or a few components in our $Feature$ matrix and have no significant influence on the global information. However, the $Feature$ matrix with the high dimension (e.g. m^2, when $m = 64$) is too large to store and match, which motivates us to employ dimension reduction techniques.

3.3.2 Dimension Reduction by FJLT

Based on the theorems in Section 3.2, FJLT is able to capture the essential features of the original data in a lower dimensional space with minor distortion, if the factor ε is close to 0. Recall the construction $\Phi = FJLT(n, d, \varepsilon)$, our work is to map the $Feature$ matrix from a high dimensional space to a lower dimensional space with minor distortion. We first get the three real-valued matrices P, H and D in our case, which is $\Phi = FJLT(N, m^2, \varepsilon)$, where H is deterministic but P and D are pseudorandomly dependent on the secret key. The lower dimension k is set to be $c'\varepsilon^{-2} \log N$ and c' is a constant. Then we can get our intermediate hash (IH) as

$$IH = \Phi(Feature) = P \cdot H \cdot D \cdot Feature \quad \text{, with size} \quad k \times N. \tag{3.6}$$

Here, the advantage of FJLT is that we can determine the lower dimension k by adjusting the number of data points, which is the number of image blocks by random sampling in our case, and the distortion rate ε. This provides us with a good chance to get a better identification performance. However, the smaller ε is, the larger k is. Hence we need to make a trade-off between ε and k in a real implementation.

3.3.3 Ordered Random Weighting

Although the original feature set has been mapped to a lower dimensional space with a small distortion, the size of intermediate hash can still be large. For instance, if we set $N = 20, \varepsilon = 0.1$ and $c' = 2$, the size of IH will be 600-by-20. To address this issue, similar to the NMF-NMF-SQ hashing in [74], we can introduce the pseudorandom weight vectors $\{w_i\}_{i=1}^{N}$ with $w_i \in R^k$ drawn from the uniform distribution $U(x|0,1)$ by the secret key, and we can calculate the final secure hash as

$$Hash = \{\langle IH_1, w_1\rangle, \langle IH_2, w_2\rangle, ..., \langle IH_N, w_N\rangle\}, \tag{3.7}$$

where IH_i is the ith column in IH, and $\langle IH_i, w_i\rangle$ is the inner product of the vectors IH_i and w_i. Hence, the final hash is obtained as a vector with length N for each image, which is compact and secure. However, the weight vector w_i drawn from $U(x|0,1)$ could diminish the distance between the hash components IH_i and IH_i' from two images and degrade the identification accuracy later. Here we describe a simple example to explain this effect. Suppose we have two vectors $A = \{10, 1\}$ and $A' = \{1, 1\}$, the Euclidean distance is 9. In the first case, if we assign the weight vector $w = \{0.1, 0.9\}$ to A and A', after the inner product (3.7), the hash values of A and A' will be 1.9 and 1 respectively. Obviously, the distance between A and A' is significantly shortened. However, if we assign the weight $w = \{0.9, 0.1\}$ to A and A' in the second case, After the inner product (3.7), the hash values of A and A' will be 9.1 and 1 respectively. The distance between A and A' is still 8.1. We would like to maintain the distinction of two vectors and avoid the effect of an inappropriate weight vector as the first case.

To maintain this distance-preserving property, a possible simple solution, referred as ordered random weighting, is to sort the elements of IH_i and w_i in a descending order before the inner product (3.7) and make sure that a larger weight value will be assigned to a larger component. In this way, the perceptual quality of the hash vector is retained by minimizing the influence of the weights. To demonstrate the effects of ordering, we investigate the correlation between the intermediate hash distances and the final hash distances when employing the unordered random weighting and ordered random weighting. Intuitively, for both the intermediate hash and the final hash, the distance between the hash generated from the original image (without distortion) and the hash from its distorted copy should increase when the attack/distortion is more severe. One example is illustrated in Figure 3.2, where we investigate 50 nature images and their 10 distorted copies with Salt & Pepper noise attacks (with variance level: $0 \sim 0.1$) from our database described in Section 3.6.1. We observe that the normalized intermediate hash distance and the final hash distance are highly correlated when using ordered random weighting, as shown in Figure 3.2 (a), while the distances are much less correlated under unordered random weighting, as shown in Figure 3.2 (b). In Figure 3.2, one example of distance correlation based on one of the 50 nature images is indicated by the solid purple lines, where a monotonically increasing relationship between the distances is clearly noticed when using ordered random weighting. Figure 3.2 suggests that the ordered random weighting in the proposed hashing approach maintains the property of low distortion in pairwise distances of the FJLT dimension reduction technique.

Furthermore, we also investigate the effect of ordering on the identification performance by comparing the ordered and unordered random weighting approaches. One illustrative example is shown in Figure 3.3, where the distances between different hashes are reported. Among 50 original images, we randomly pick out one as the target image and use its distorted copies as the query images to be identified. To compare the normalized Euclidean distances between the final hashes of the query images and the original 50 images, the final hash distances between the target image and its distorted

copies are indicated by red squares, and others are marked by blue crosses. For the Salt & Pepper noise attacks (with variance level: $0 \sim 0.1$) as shown in Figure 3.3 (a) & (b), we can see that, when using both ordered random weighting and unordered random weighting, the query images could be easily identified as the true target image based on the identification process described in Section 3.3.4. It is also clear that the ordered random weighting approach should provide a better identification performance statistically since the distance groups are better separated. For the Gaussian blurring attacks (with filter size: $3 \sim 21$) as shown in Figure 3.3 (c) & (d), it is clear that the correct classification/identification can only be achieved by using the ordered random weighting. Based on the two examples illustrated in Figure 3.3 and the tests on other attacks described in Section 3.6.1, we notice that the identification performance under the blurring attacks is significantly improved using the ordered random weighting when compared with the unordered approach. The improvement is less significant under noise and other attacks. In summary, we observe that ordered random weighting maintains better the distance-preserving property of FJLT compared with the unordered random weighting and thus yields a better identification performance.

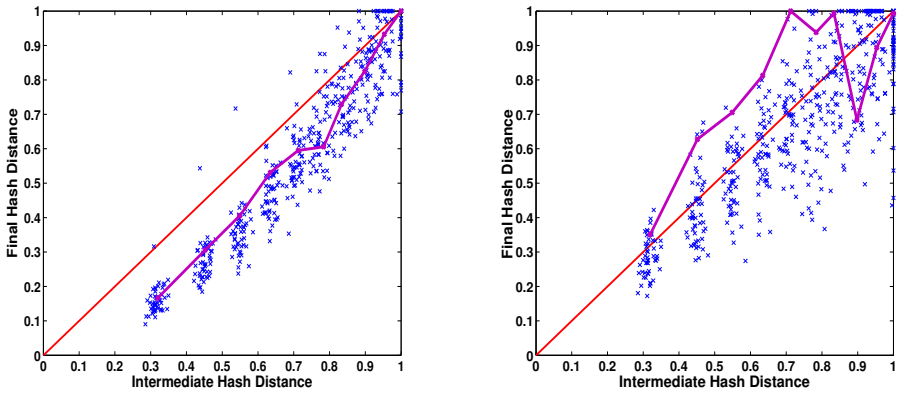
3.3.4 Identification and Evaluation

Identification Process

Let $S = \{s_i\}_{i=1}^N$ be the set of original images in the tested database and define a space $H(S) = \{H(s_i)\}_{i=1}^N$ as the set of corresponding hash vectors. We use Euclidean distance as the performance metric to measure the discriminating capability between two hash vectors, defined as

$$Distance = \|H(s_1) - H(s_2)\|_2 = \sqrt{\sum_{i=1}^{n} (h_i(s_1) - h_i(s_2))^2}, \tag{3.8}$$

where $H(s_i) = \{h_1(s_i), h_2(s_i), ..., h_n(s_i)\}$ means the corresponding hash vector with length n of the image s_i. Given a tested image D, we first calculate



(a) Ordered (b) Unordered

Figure 3.2: An example of the correlations between the final hash distance and the intermediate hash distance based on 50 images under salt & pepper noise attacks (with variance level: $0 \sim 0.1$) when employing ordered random weighting and unordered random weighting.

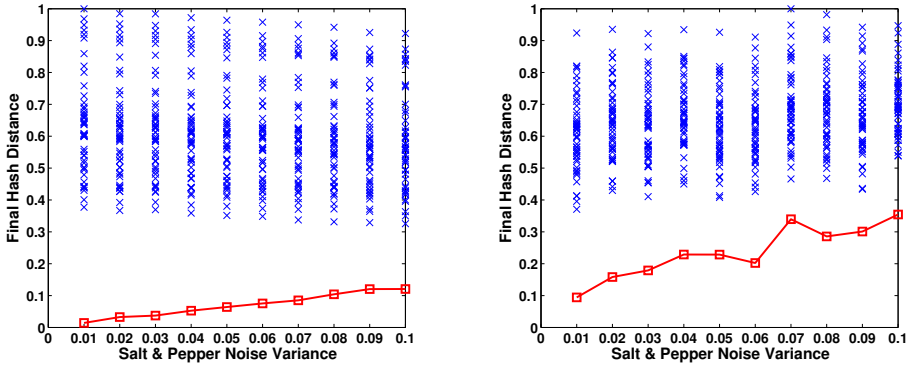
its hash $H(D)$ and then obtain its distances to each original image in the hash space $H(S)$. Intuitively, the query image D is identified as the $\hat{i}$th original images which yields the minimum corresponding distance, expressed as,

$$\hat{i} = \arg\min_{i} \{\|H(D) - H(s_i)\|_2\}, \quad i = 1, ..., N. \tag{3.9}$$

The simple identification process described above can be considered as a special case of the K-nearest-neighbor classification approach with $K = 1$. Here K is set as 1 since we only have one copy of each original image in the current database. For a more general case, if we have K multiple copies of each original image with no distortion or with only slight distortions, we could adopt the K-nearest neighbor (KNN) algorithm for image identification in our problem.

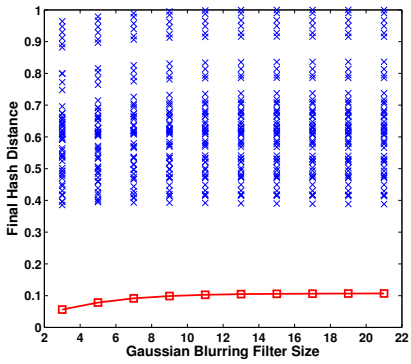
Receiver Operating Characteristics Analysis

Except investigating identification accuracy, we also study the receiver operating characteristics (ROC) curve [26] to visualize the performance of dif-

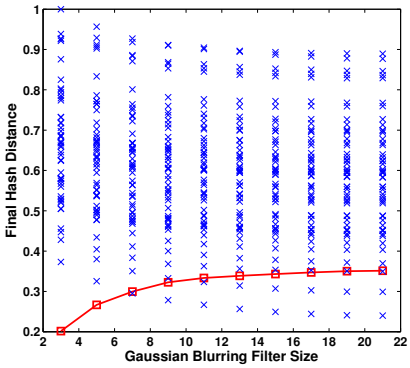


(a) Ordered

(b) Unordered



(c) Ordered



(d) Unordered

Figure 3.3: Illustrative examples to demonstrate the effect of ordering on the identification performance. The final hash distances between the query images and the original 50 images are shown for comparing the ordered random weighting and the unordered random weighting approaches. (a) & (b) The query images are under salt & pepper noise attacks. (c) & (d) The query images are under Gaussian blurring attacks.

ferent hashing approaches, including NMF-NMF-SQ hashing, FJLT hashing and Content-based fingerprinting proposed later. The ROC curve depicts the relative tradeoffs between benefits and cost of the identification and is an effective way to compare the performances of different hashing approaches.

To obtain ROC curves to analyze the hashing algorithms, we may define the probability of true identification $P_T(\xi)$ and probability of false alarm

$P_F(\xi)$ as

$$P_T(\xi) = Pr(\|H(I) - H(I_M)\|_2 < \xi), \tag{3.10}$$

$$P_F(\xi) = Pr(\|H(I) - H(I'_M)\|_2 < \xi), \tag{3.11}$$

where ξ is the identification threshold. The image I and I' are two distinct original images and the image I_M and I'_M are manipulated versions of the image I and I' respectively. Ideally, we hope that the hashes of the original image I and its manipulated version I_M should be similar and thus be identified accurately, while the distinct images I and I'_M should have different hashes. In other words, given a certain threshold ξ, an efficient hashing should provide a higher $P_T(\xi)$ with a lower $P_F(\xi)$ simultaneously. Consequently, when we obtain all the distances between manipulated images and original images, we could generate a ROC curve by sweeping the threshold ξ from the minimum value to the maximum value, and further compare the performances of different hashing approaches.

3.4 Rotation Invariant FJLT Hashing

Although the Fast Johnson-Lindenstrauss transform has been shown to be successful in the hashing in our previous preliminary work[59], the FJLT hashing can still be vulnerable to rotation attacks. Based on the hashing scheme described in Section 3.3, random sampling can be an effective approach to reduce the distortion introduced by cropping, and scaling attack can be efficiently tackled by upsampling and downsampling in the preprocessing. However, to successfully handle the rotation attacks, we need to introduce other geometrically invariant transform to improve the performance of the original FJLT hashing.

3.4.1 Fourier-Mellin Transform

The Fourier-Mellin transform (FMT) is a useful mathematical tool for image recognition and registration, because its resulting spectrum is invariant to

rotation, translation and scaling [92], [51]. Let f denote a gray-level image defined over a compact set of $\mathbb{R}^2$, the standard FMT of f in polar coordinates (log-polar coordinates) is given by:

$$M_f(k,v) = \frac{1}{2\pi}\int_0^{2\pi}\int_0^{\infty} f(r,\theta) r^{-iv} e^{-ik\theta} \mathrm{d}\theta \frac{\mathrm{d}r}{r}. \tag{3.12}$$

If we make $r = e^{\gamma}, \mathrm{d}r = e^{\gamma}\mathrm{d}\gamma$, the equation (3.12) is clearly a Fourier transform like:

$$M_f(k,v) = \frac{1}{2\pi}\int_0^{2\pi}\int_{-\infty}^{\infty} f(e^{\gamma},\theta) e^{-iv\gamma} e^{-ik\theta} \mathrm{d}\gamma \mathrm{d}\theta. \tag{3.13}$$

Therefore, the FMT could be divided into three steps, which result in the invariance to geometric attacks:

- *Fourier Transform*: It converts the translation of original image in spatial domain into the offset of angle in spectrum domain. The magnitude is translation invariant.
- *Cartesian to Log-Polar Coordinates*: It converts the scaling and rotation in Cartesian coordinates into the vertical and horizontal offsets in Log-Polar Coordinates.
- *Mellin Transform*: It is another Fourier transform in Log-Polar coordinates and converts the vertical and horizontal offsets into the offsets of angles in spectrum domain. The final magnitude is invariant to translation, rotation and scaling.

However, the inherent drawback of the Fourier transform makes FMT only robust to geometric transform, but vulnerable to many other classical signal processing distortions such as cropping and noising. As we know, when converting an image into the spectrum domain by 2D Fourier transform, each coefficient is contributed by all the pixels of the image. It means that the Fourier coefficients are dependent on the global information of the image in the spatial domain. Therefore, the features extracted by Fourier-Mellin transform are sensitive to certain attacks such as noising and cropping, because the global information is no longer maintained. To overcome

this problem, we have modified the FMT implementation in our proposed rotation-invariant FJLT (RI-FJLT) hashing.

3.4.2 RI-FJLT Hashing

The invariance of FMT to geometric attacks such as rotation and scaling has been widely applied in image hashing [92], [101] and watermarking [51], [6]. It also motivates us to address the deficiency of FJLT hashing by incorporating FMT. Here, we propose the rotation-invariant FJLT hashing by introducing FMT into the FJLT hashing. Specially, the proposed rotation-invariant FJLT hashing (RI-FJLT) consists of three steps:

- Step 1: Converting the image into the Log-Polar coordinates

$$I(x, y) \rightarrow G(\log \rho, \theta), \tag{3.14}$$

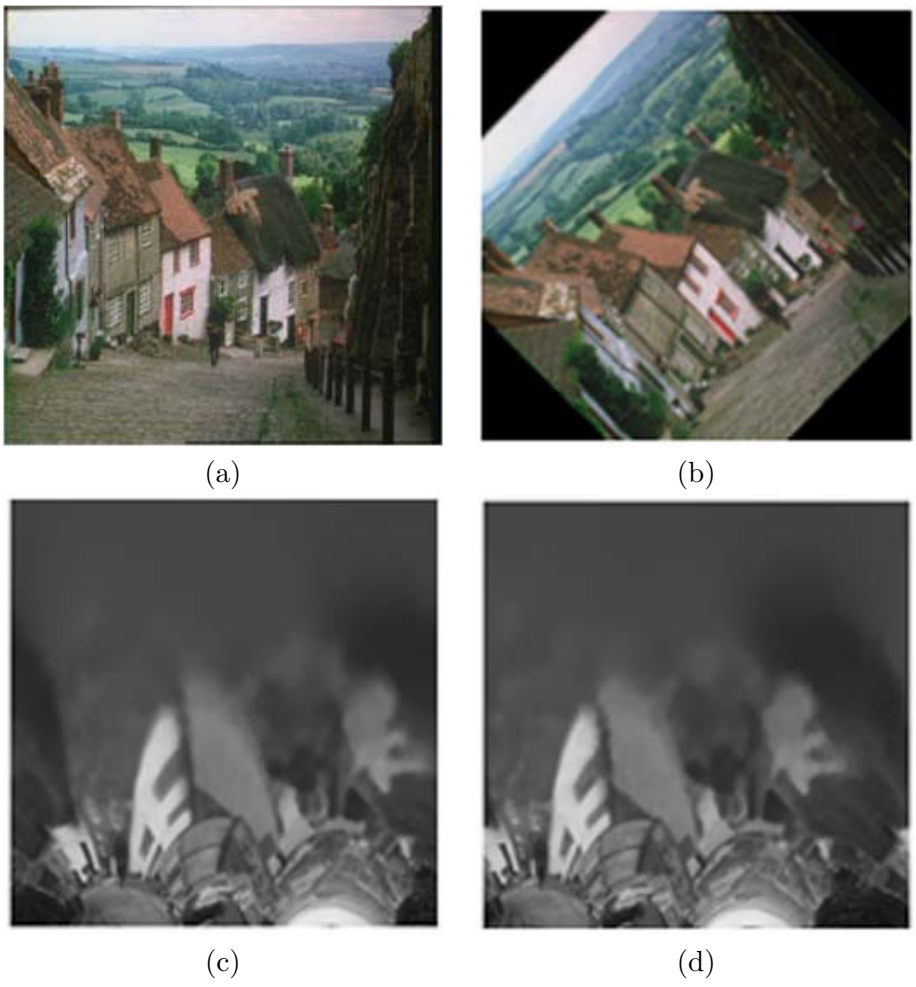
  where x and y are Cartesian coordinates and ρ and θ are Log-Polar coordinates. Any rotation and scaling will be considered as vertical and horizontal offsets in Log-Polar coordinates. An example is given in Figure 3.4.

- Step 2: Applying Mellin transform (Fourier transform under Log-Polar coordinates) to the converted image and return the magnitude feature image.

- Step 3: Applying FJLT hashing in Section 3.3 to the magnitude feature image derived in Step 2.

For the conversion in Step 1, since the pixels in Cartesian coordinates are not able to be one-to-one mapped to pixels in the Log-Polar coordinates space, some value interpolation approaches are needed. We have investigated three different interpolation approaches for the proposed RI-FJLT hashing, including nearest neighbor, bilinear and bicubic interpolations, and found that the bilinear is superior to others. Therefore we only report the results under bilinear interpolation here. Note that we abandon the first step of FMT in RI-FJLT hashing, because we only focus on rotation attacks

(other translations are considered as cropping) and it is helpful to reduce the influence of noising attacks by removing the Fourier transform step. The performance will be illustrated in Section 3.6. However, since Step 2 can inevitably be affected by attacks such as noising etc., some preprocessing such as median filtering can help improve the final identification performance.



(a) (b)

(c) (d)

Figure 3.4: An example of conversion from Cartesian coordinates to log-polar coordinates. (a) Original goldhill. (b) Goldhill rotated by 45^0. (c) Original goldhill in log-polar coordinates. (d) Rotated goldhill in log-polar coordinates.

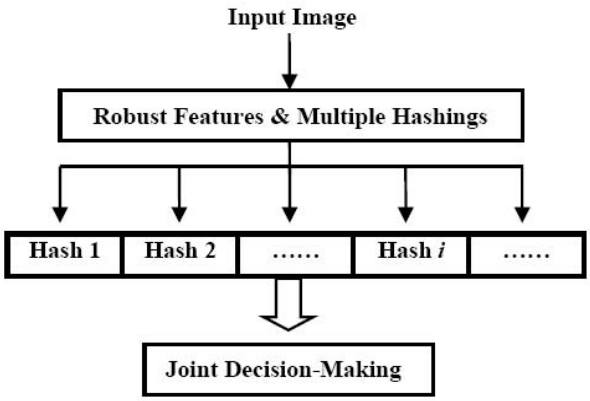


Figure 3.5: The conceptual framework of the content-based fingerprinting

3.5 Content-Based Fingerprinting

3.5.1 Concept and Framework

Considering that certain features can be more robust against certain attacks, to take advantage of different features, we plan to propose a new content-based fingerprinting concept. This concept combines benefits of conventional content-based indexing (used to extract discriminative content features) and multimedia hashing. Here we define content-based image fingerprinting as a combination of multiple robust feature descriptors and secure hashing algorithms. Similar to the concept of image hash, it is a digital signature based on the significant content of image itself and represents a compact and discriminative description for the corresponding image. Therefore, it has a wide range of applications in practice such as integrity verification, watermarking, content-based indexing, identification and retrieval etc. The framework is illustrated in Figure 3.5.

Specially, each vertical arrow in Figure 3.5 represents an independent hashing generation procedure, which consists of robust feature extraction and intermediate hash generation proposed by [72, 92]. Because it is the combination of various hash descriptors, the content-based fingerprinting can be considered as an extension and evolution of image hashing, and

thus offers much more freedom to accommodate different robust features (color, shape, texture, salient points etc. [89]) and design efficient hashing algorithms to successfully against different types of attacks and distortions. Similar to the idea of finding one-to-one relationships between the fingerprints and an individual human being, the goal of content-based fingerprinting is to generate an exclusive digital signature, which is able to uniquely identify the corresponding media data no matter which content-preserving manipulation or attack is taken on.

Compared with the traditional image hashing concept, the superiority of content-based fingerprint concept lies in its potential high discriminating capability, better robustness and multilayer security arising from the combination of various robust feature descriptors and a joint decision-making process. Same as in any information fusion processes, theoretically the discrimination capability of the content-based fingerprinting with effective joint decision-making scheme should outperform a single image hashing. Since the content-based fingerprint consists of several hash vectors, which are generated based on various robust features and different secret keys, it is argued that the framework of content-based fingerprinting results in a better robustness and multilayer security when an efficient joint decision-making is available. However, combining multiple image hashes approaches requires additional computation cost for the generation of content-based fingerprinting. The tradeoff between computation cost and performance is a concern with great importance in practice.

3.5.2 A Simple Content-Based Fingerprinting Approach

From the experimental results in Section 3.6, we note that FJLT hashing is robust to most types of the tested distortions and attacks except for rotation attacks and that RI-FJLT hashing provides a significantly better performance for rotation attacks at the cost of the degraded performances under other types of attacks. Recall an important fact that it's relatively easy to find a robust feature to resist one specific type of distortion, however it is very difficult, if not impossible, to find a feature which is uniformly robust to

against all types of distortions and attacks. Any desire to generate an exclusive signature for the image by a single image hashing approach is infeasible. Here we plan to demonstrate the advantages of the concept of content-based fingerprinting by combining the proposed FJLT hashing and RI-FJLT hashing. The major components of the content-based fingerprinting framework include hash generations and the joint decision-making process which should take advantage of the combinations of the hashes to achieve a superior identification decision-making. Regarding the joint decision-making, there are many approaches in machine learning [12] that can be useful. Here we only present a simple decision-making process in rank level [35] to demonstrate the superiority of content-based fingerprinting.

Given an image d with certain distortion, we respectively generate the hash vectors H_f^d and H_r^d by FJLT and RI-FJLT hashing. Suppose the hash values of original images s are H_f^s and H_r^s generated by FJLT and RI-FJLT hashing respectively. We denote $P_f(s|d)$ as the confidence measure that we identify image d as image s when applying the FJLT hashing. Similarly, $P_r(s|d)$ is denoted for that of the RI-FJLT hashing. Here, we simply define:

$$P_f(s|d) = W_f(1 - \frac{Norm(H_f^d - H_f^s)}{Norm(H_f^s)}), \tag{3.15}$$

$$P_r(s|d) = W_r(1 - \frac{Norm(H_r^d - H_r^s)}{Norm(H_r^s)}), \tag{3.16}$$

where W_f and W_r are pre-selected weights in the case of FJLT and RI-FJLT hashing respectively and $Norm$ means the Euclidean norm. Considering the poor performances of RI-FJLT hashing under many other types of attacks except for rotation ones, we intuitively introduce the weights W_f and W_r, where $0 < W_r < W_f \leq 1$, to the original confidence measures of FJLT and RI-FJLT hashing to decrease the possible negative influence of RI-FJLT hashing and maintain the advantages of both FJLT and RI-FJLT hashing in the proposed content-based fingerprinting under different attacks.

Regarding the identification decision making, given a tested image d, we calculate all the confidence measures $P_f(s_i|d)_{i=1}^N$ and $P_r(s_i|d)_{i=1}^N$ over the

image database of $S = \{s_i\}_{i=1}^{N}$ by using FJLT and RI-FJLT hashing, and make the identification decision correspondingly by selecting the highest one among $P_f(s_i|d)_{i=1}^{N}$ and $P_r(s_i|d)_{i=1}^{N}$. Note that if a confidence measure $P(s|d)$ is negative, it means that the image d is outside the confidence interval of the image s and the confidence measure is assigned to be zero.

3.6 Experimental Results and Analysis

3.6.1 Database and Content-Preserving Manipulations

In order to evaluate the performance of the proposed new hashing algorithms, we test FJLT hashing and RI-FJLT hashing on a database of 100000 images. In this database, there are 1000 original colour nature images, which are mainly selected from the ten sets of categories in the content-based image retrieval database of the University of Washington [2] as well as our own database. Therefore, some of the original images can be similar in content if they come from the same category, and some are distinct if they come from the different categories. For each original colour image with size 256×384, we generate 99 similar but distorted versions by manipulating the original image according to eleven classes of content-preserving operations, including additive noise, filtering operations, geometric attacks and so on, as listed in Table 3.1. All the operations are implemented using Matlab. Here we give some brief explanations of some ambiguous manipulations. For image rotation, a black frame around the image will be added by Matlab but some parts of image will be cut if we want to keep its size the same as the original image. An example is given in Figure 3.4(b). Here our cropping attacks refer to the removal of the outer parts (i.e. let the values of the pixels on each boundary equal to null and keep the significant content in the middle).

3.6.2 Identification Results and ROC Analysis

Our preliminary study [59] on a small database showed that FJLT hashing provides nearly perfect identification accuracy for the standard test images such as Baboon, Lena, and Peppers. Here we will measure the FJLT hashing

Table 3.1: Content-preserving manipulations and parameter settings

Manipulation	Parameters Setting	Number
Additive Noise		
Gaussian Noise	sigma: $0 \sim 0.1$	10
Salt&Pepper Noise	sigma: $0 \sim 0.1$	10
Speckle Noise	sigma: $0 \sim 0.1$	10
Blurring		
Gaussian Blurring	filter size: $3 \sim 21$, sigma=5	10
Circular Blurring	radius: $1 \sim 10$	10
Motion Blurring	len: $5 \sim 15$, $\theta : 0^0 \sim 90^0$	9
Geometric Attacks		
Rotation	degree $= 5^0 \sim 45^0$	9
Cropping	5%,10%,20%,25%,30%,35%	6
Scaling	25%, 50%, 75%, 150%, 200%	5
JPEG Compression	Quality Factor= $(5 \sim 50)$	10
Gamma Correction	$\gamma = (0.75 \sim 1.25)$	10

and the new proposed RI-FJLT hashing on the new database, which consists of 1000 nature images from ten categories. Ideally, to be robust to all routine degradations and malicious attacks, no matter what content-preserving manipulation is done, the image with any distortion should still be correctly classified into the corresponding original image.

It is worth mentioning that all the pseudorandomizations of NMF-NMF-SQ (statistics quantization) hashing, FJLT hashing, and content-based fingerprinting are dependent on the same secret key in our experiment. As discussed in [74], the secret keys, more precisely the key-based randomizations, play important roles on both increasing the security (i.e. making the hash unpredictable) and enhancing scalability (i.e. keeping the collision ability from distinct images low and thus yielding a better identification performance) of the hashing algorithm. Therefore, the identification accuracy of a hashing algorithm is determined simultaneously by both the dimension reduction techniques (e.g. FJLT and NMF) and the secret keys. As shown in NMF hashing in [74], if we generate hashes of different images with varied secret keys, the identification performance can be further improved signif-

Table 3.2: Identification accuracy for manipulated images by NMF-NMF-SQ (NMF) hashing, FJLT hashing and content-based fingerprinting (CBF) based on FJLT & RI-FJLT hashing)

Manipulations	NMF	FJLT	CBF
Additive Noise			
Gaussian Noise *	59.38%	69.5%	62.36%
Salt&Pepper Noise	81.87%	96.87%	97.71%
Speckle Noise	78.27%	99.83%	99.77%
Blurring			
Gaussian Blurring	98.31%	99.49%	99.04%
Circular Blurring	98.36%	99.51%	99.09%
Motion Blurring	98.88%	99.81%	99.66%
Geometric Attacks			
Rotation	16.43%	36.86%	86.54%
Cropping	16.75%	96.6%	96.14%
Scaling	98.47%	100%	100%
JPEG Compression	99.7%	100%	100%
Gamma Correction	5.22%	86.62%	74.26%

* With the help of median filter in preprocessing, the identification accuracy of NMF hashing under Gaussian noise could be improved to 90.61%, and 99.5% for FJLT hashing

icantly because the secret key boosts up the cardinality of the probability space and brings down the probability of false alarm. Since we mainly focus on examining the identification capacity of hashing schemes themselves rather than the effects of secret keys, to minimize the effects of the factor of the secret keys, we use the same key in generating hash vectors for different images.

Results of FJLT Hashing

Following the algorithms designed in Section 3.3, we test the FJLT hashing with the parameters chosen as $m = 64$, $N = 40$, $\varepsilon = 0.1$, $key = 5$, as summarized in Table. 3.3. Note that most of the keys could be used in FJLT

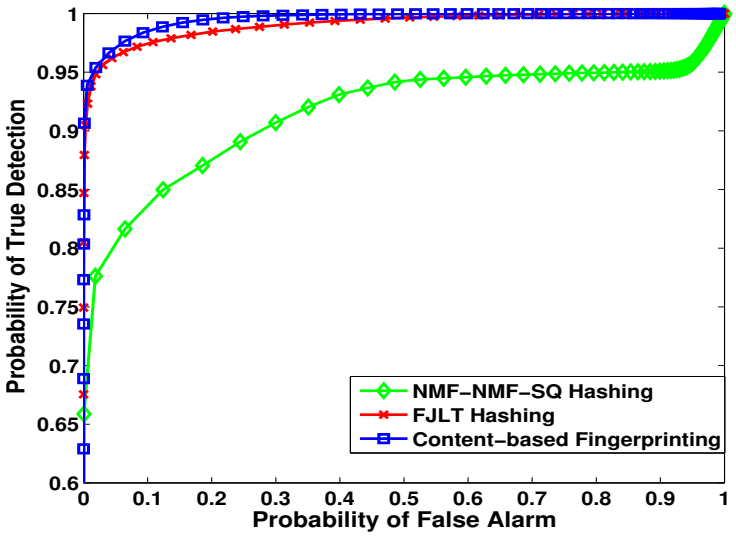


Figure 3.6: The overall ROC curves of NMF-NMF-SQ hashing, FJLT hashing, and content-based fingerprinting under all types of tested manipulations

hashing because of its robustness to secret keys, which has been illustrated in [59]. Since the NMF-NMF-SQ hashing has been shown to outperform the SVD-SVD and PR-SQ hashing algorithms having the best known robustness properties in the existing literature, we compare the performance of our proposed FJLT hashing algorithm with NMF-NMF-SQ hashing when testing on the new database. For the NMF approach, the parameters are set as $m = 64$, $p = 10$, $r_1 = 2$, $r_2 = 1$, and $M = 40$ according to [74]. It is worth

Table 3.3: Parameter setting in the FJLT hashing algorithm

Parameter	Value
Size of the subimage	$m = 64$
Length of the hash vector	$N = 40$
Parameters of FJLT	$\epsilon = 0.1$, $c = 250$, $c' = 1$
Secret key	$key = 5$

mentioning that, to be consistent with the FJLT approach, we chose the same size of subimages and length of hash vector in NMF hashing (denoted as m and M), which facilitate a fair comparison between them later. We also tried the setting $p = 40$ (with p represents the number of subimages in the NMF approach), but it was found that the choice of $p = 10$ yields a better performance. Consequently, NMF hash vector has the same length 40 as the FJLT hash vector. We first examine the identification accuracy of both hashing algorithms under different attacks, and the identification results are shown in Table 3.2. It is clearly noted that the proposed FJLT hashing consistently yields a higher identification accuracy than that of NMF hashing under different types of tested manipulations and attacks.

We then present a statistical comparison of the proposed FJLT and NMF hashing algorithms by studying the corresponding ROC curves. We first generate the overall ROC curves for all types of tested manipulations when applying different hashing schemes, and the resulting ROC curves are shown in Figure 3.6. From Figure 3.6, one major observation is that the proposed FJLT hashing outperforms NMF-NMF-SQ hashing. To test the robustness to each type of attacks, an ROC curve is also generated for a particular attack and hash algorithm. Since we note from Table 3.2 that the proposed FJLT hashing significantly outperforms NMF-NMF-SQ for additive noise, cropping and gamma correction attacks, we show the ROC curves corresponding to the six attacks (i.e. Gaussian noise, Salt&Pepper noise, Speckle noise, Rotation attacks, Cropping and Gamma correction) in Figure 3.8. Once again, the ROC curves in Figure 3.8 reinforce the observation that FJLT hashing significantly outperform the state-of-art NMF hashing. However, both of them are still a little sensitive to Gaussian noise as shown in Figure 3.8 (a). The underlying reason is that we didn't incorporate any preprocessing such as median filter into FJLT hashing or NMF hashing, because we would investigate the robustness of FJLT and NMF hashing themselves to additive noise. In practice, the preprocessing such as image denoising before applying image hashing could further improve the robustness to additive noise (referring to the annotation below Table 3.2), since both FJLT hashing and NMF hashing are strongly robust to blurring.

As for the attacks such as JPEG compression and Blurring, since we observe perfect identification performances and no false alarms in our own experiments, we do not report the ROC curves further, which are similar to the ROC results via NMF hashing shown in [74].

Here we try to give some intuitive explanations regarding the observed performances of the two hashing algorithms. In NMF hashing, the dimension reduction technique is based on the approximative non-negative matrix factorization, which factorizes the image matrix into two lower rank matrices. However, the problem of choosing a low rank r (e.g. r_1, r_2 in the NMF hashing) is of great importance, though it is observed to be sensitive to the data. While for FJLT hashing, the mapping is obtained by a coefficients matrix and a subimage is treated as a point in a high dimensional space (in our case, the dimension is $64 \times 64 = 4096$). One advantage of FJLT hashing is that minor modifications in the content will not affect the integrity of the global information, which results in a better performance. However, as illustrated in Table 3.2 and the ROC curve in Figure 3.8 (d), both FJLT hashing and NMF hashing provide poor performances under rotation attacks, and we shall investigate this problem further.

Results of RI-FJLT Hashing

In Table 3.2, we note that one drawback of FJLT hashing is its vulnerability to rotation attacks. Especially, as shown by an example in Figure 3.4, for a large rotation degree of 45, FJLT hashing failed to identify the image content. Here we apply the RI-FJLT hashing approach presented in Section 3.4 to overcome this drawback.

We generated 36 rotated versions for each test image in the database and the rotation degrees are varied from 5 to 180 with an interval of 5 degrees. Though not investigated further here, it is worth mentioning that, before the conversion from Cartesian coordinates to Log-Polar coordinates, some preprocessing operations such as median filtering can be helpful to enhance the identification performance [92], especially under additive noise distortions. We have employed median filter as preprocessing in RI-FJLT hashing. The

identification results under rotation attacks are shown in Table 3.4. We can see from the table that FJLT hashing is obviously sensitive to rotation attacks and thus its identification accuracy greatly degrades with the increase of rotation degree. It is also noted that RI-FJLT hashing still consistently achieves almost perfect identification accuracy under rotation attacks even with large rotation degrees.

Although the invariance of Fourier-Mellin transform benefits the FJLT hashing with the robustness to rotation attacks, such robustness to rotation comes at the cost of degraded identification accuracy for other types of manipulations and attacks. We have intuitively discussed the reasons for this observation in Section 3.4. We argue that it may not be feasible to be robustly against various attacks by only depending on single feature descriptor. This observation motivates us to look for an alternative solution that is the content-based fingerprinting we proposed in Section 3.5 to tackle this problem.

Results of Content-based Fingerprinting

Since FJLT hashing is demonstrated to be robust against a large class of distortions except for rotation attacks and RI-FJLT hashing achieves superior performance under rotation attacks at the cost of sensitivity to other manipulations, It accounts for the fact that it is very difficult to design a globally optimal hashing approach that could handle all of the distortions and manipulations. Hence, we combine FJLT hashing and RI-FJLT hashing following

Table 3.4: Identification accuracy under rotation attacks by FJLT and RI-FJLT hashing

Rotation Degree	FJLT	RI-FJLT
$5^0 \sim 45^0$	30.43%	94.57%
$50^0 \sim 90^0$	0.67%	96.03%
$95^0 \sim 135^0$	0.58%	94.62%
$140^0 \sim 180^0$	1.13%	96.06%
Overall	8.2%	95.32%

the framework of content-based fingerprinting proposed in Section 3.5 and test its performance on the database described in Section 3.6.1. Considering the poor performance of RI-FJLT hashing on other manipulations, we need to introduce an elaborate weight shown in Section 3.5.2 to the confidence measure of RI-FJLT hashing to get rid of its negative influence and try to maintain the advantages of both FJLT and RI-FJLT hashing in the proposed content-based fingerprinting. Based on our preliminary study, we set $W_f = 1$ to keep the advantages of FJLT hashing and find that a good weight W_r could be drawn from the interval range $\{0.85 \sim 0.9\}$. We set $W_r = 0.895$ in our implementation and exhibit the results in Table 3.2.

To have a fair comparison between different approaches, though we combine the FJLT hashing and the RI-FJLT hashing in the content-based fingerprinting, the length of the overall fingerprint vector is still chosen as 40 (with 20 components from the FJLT hashing and the left 20 from the RI-FJLT hashing), which is the same as that of the FJLT hashing and the NMF hashing. It is clear that the simple joint decision-making complements the drawback of FJLT hashing under rotation attacks by incorporating the RI-FJLT hashing into the proposed content-based fingerprinting. The ROC curves for FJLT hashing, NMF hashing and the proposed content-based fingerprinting under rotation attacks are shown in Figure 3.8 (d). Obviously, among the three approaches, the content-based fingerprinting yields the highest true positive rates when the same false positive rates are considered. The ROC curves of the content-based fingerprinting approach under other types of attacks are also illustrated in Figure 3.8. We note that the robustness of content-based fingerprinting to additive noise, cropping, and Gamma correction slightly degrades, as shown in Figure 3.8. One possible explanation could be that the current simple decision-making process is not the theoretically optimal one that could eliminate the negative effect of RI-FJLT hashing under these attacks. However, the overall performance of content-based fingerprinting as illustrated by the ROC curve in Figure 3.6 demonstrates that it is superior and more flexible than a single hashing approach, because the selection of features and secure hashes can be adapted to address different practical application concerns. Therefore, the proposed

content-based fingerprinting can be a promising extension and evolution of traditional image hashing.

3.6.3 Unpredictability Analysis

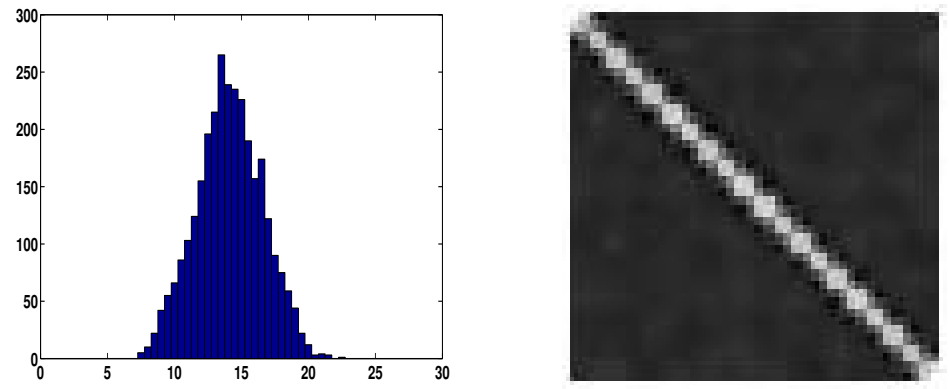
Except for the robustness against different types of attacks, the security in terms of unpredictability that arises from the key-dependent randomization is another important property of hashing and the proposed content-based fingerprinting. Here we mainly focus on the unpredictability analysis of FJLT hashing, because the unpredictability of the RI-FJLT hashing and the content-based fingerprinting proposed arise from the FJLT hashing. Higher amount of the randomness in the hash values makes it harder for the adversary to estimate and forge the hash without knowing the secret keys. Since it is believed that a high differential entropy is a necessary property of secure image hashes, we evaluate the security in terms of unpredictability of FJLT hashing by quantifying the differential entropy of the FJLT hash vector, as proposed in [92]. The differential entropy of a continuous random variable X is given by

$$H(X) = \int_{\Omega} f(x) \log \frac{1}{f(x)} \mathrm{d}x, \tag{3.17}$$

where $f(x)$ means the probability density function (pdf) of X and Ω means the support area of $f(x)$. Since the analytical model of the pdf of the FJLT hash vector component is generally not available, we carry out the practical pdf approximation using the histograms of the hash vector components. Figure 3.7(a) shows the histogram of a typical component from the FJLT hash vector of image Lena resulting from 3000 different keys. It is noted that it approximately follows a Gaussian distribution. Similarly, we can obtain the histograms of other components. Based on our observations, we state that the FJLT hash vector approximately follows a multivariate Gaussian distribution. Therefore, similar to the hash in [74], we have the differential entropy of the FJLT hash vector X as

$$H(X) = \frac{1}{2} \log(2\pi e)^N |Cov| \ bits, \tag{3.18}$$

where $|Cov|$ means the determinant of the covariance matrix of the hash vector, and N means the length of the FJLT hash vector.



(a) (b)

Figure 3.7: . (a) The histogram of a typical FJLT hash vector component for image lena from 3000 different secret keys. (b) The covariance matrix of the FJLT hash vectors for image lena from 3000 different secret keys.

From Figure 3.7(b) where an example of the covariance matrix of the FJLT hash vector is shown, we can see that the covariance matrix is approximately a diagonal matrix, meaning that the components are approximately statistically independent. Therefore, $|Cov|$ can be approximately estimated as

$$|Cov| = \prod_{i=1}^{N} \sigma_i^2, \tag{3.19}$$

where σ_i^2 means the variance of the component h_i in the FJLT hash vector. Since from information theory, the differential entropy of a random vector $X \in \mathbb{R}^n$ is maximized when X follows a multivariate normal distribution $\mathcal{N}(0, Cov)$ [21], we argue that the proposed FJLT hashing is highly secure (unpredictable) as it approximately follows Eqn. 3.18. We note that NMF-NMF-SQ hashing also was shown to approximately follow a joint Gaussian distribution and a similar statement in terms of differential entropy was given in [74]. Hence, we state that the proposed FJLT hash is comparably

as secure as NMF hashing, which was shown to be presumably more secure than previously proposed schemes that are based on random rectangles alone [74]).

However, the security of image hashing does not only lie on a higher differential entropy, which is only one aspect of a secure image hashing [74, 92], but also includes other factors such as key diversity and prior knowledge possessed by adversaries. Therefore, how to comprehensively evaluate the security of image hashing is still an open question. Interested readers could refer to the literatures [64, 92] regarding the security analysis issues.

3.6.4 Computational Complexity

We analyze the computational complexity of the proposed FJLT hashing and RI-FJLT algorithms (the computational cost of content-based fingerprinting is the sum of FJLT and RI-FJLT hashing) when compared with the NMF-NMF-SQ hashing algorithm.

- NMF: In [74], the computational complexity of NMF-NMF-SQ hashing has been given as follows: it does a rank r_1 NMF on n $m \times m$ matrices and then a rank r_2 approximation from the resulting $m \times 2pr_1$ matrix in [74]. At last, pseudorandom numbers are incorporated in the NMF-NMF vector of length $mr_2 + 2pr_1r_2$, and the total computation cost is

$$C_{NMF} = n \cdot \mathcal{O}(m^2 r_1) + \mathcal{O}(2mnr_1r_2) + \mathcal{O}(mr_2 + 2nr_1r_2). \quad (3.20)$$

- FJLT: Based on the analysis in [5], given a $x \in \mathbb{R}^d$, the computation cost of FJLT on x is calculated as follows. Computing $D(x)$ requires $\mathcal{O}(d)$ time and $H(Dx)$ requires $\mathcal{O}(d \log d)$. For computing $P(HDx)$, it takes $\mathcal{O}(p)$, where the p is the number of nonzeros in P, we know the p satisfies the Binomial distribution $B(dk, q)$, therefore we take the mean value of p as dkq that equals to $k \log^2 n$, where k is $\varepsilon^{-2} \log n$. Then, take the random weight incorporation into account, we have the

Table 3.5: Computational time costs for lena with 256×256 by FJLT, RI-FJLT and NMF-NMF-SQ hashing algorithms

Computational Cost	FJLT	RI-FJLT	NMF-NMF-SQ
time (s)	1.93	2.43	5.55

total computation cost of the FJLT hashing as ($d = m^2$ in our case)

$$C_{FJLT} = \mathcal{O}(m^2(1 + 2\log m)) + \mathcal{O}(k(1 + \log^2 n)). \tag{3.21}$$

- RI-FJLT: Except for the cost of FJLT hashing, we need to take the bilinear interpolation that requires $\mathcal{O}(m^2)$ and Fourier transform that takes $\mathcal{O}(m^2 \log m)$ by FFT into account. Consequently, the cost of RI-FJLT is

$$C_{RI-FJLT} = \mathcal{O}(m^2(2 + 3\log m)) + \mathcal{O}(k(1 + \log^2 n)). \tag{3.22}$$

Here, we specify that $k \approx 5m$ in our case and also take other parameters into account. Obviously the FJLT and RI-FJLT hashing roughly require a lower computational cost than that of NMF-NMF-SQ. To have an intuitive feeling of the computational costs required by different algorithms, we also test on a standard image Lena with size 256×256 by using a computer with Intel Core 2 CPU (2.00 GHz) and 2G RAM. The required computational time is listed in Table 3.5, which shows that the FJLT and RI-FJLT hashing are much faster than NMF-NMF-SQ hashing. Note that the costs are based on a length 20 of the hash vectors in our experiments. Increasing the length of hash vectors will enhance the identification accuracy but will require more computational costs. This trade-off will be further studied in the future.

3.7 Conclusion

In this chapter, we have introduced a new dimension reduction technique– FJLT, and applied it to develop new image hashing algorithms. Based on our experimental results, it is noted that the FJLT-based hashing is robust to

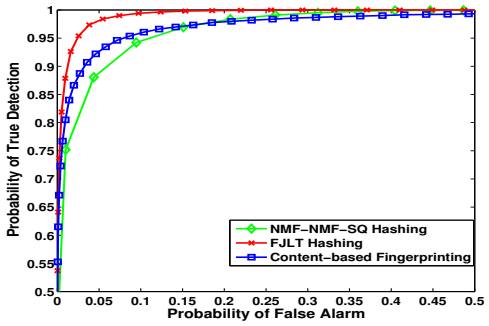
a large class of routine distortions and malicious manipulations. Compared with the NMF-based approach, the proposed FJLT hashing can achieve comparable, sometimes better, performances than those of NMF, while requiring less computational cost. The random projection and low distortion properties of FJLT make it more suitable for hashing in practice than the NMF approach. Further, we have incorporated Fourier-Mellin transform to complement the deficiency of FJLT hashing under rotation attacks. The experimental results confirm the fact that generating a hash descriptor based on a certain type of features to resist all types of attacks is highly unlikely in practice. However, for a particular type of distortion, it is feasible to find a specific feature to tackle it and obtain good performance. These observations motivate us to propose the concept of content-based fingerprinting as an extension of image hashing and demonstrate the superiority of combining different features and hashing algorithms.

We note that the content-based fingerprinting approach by using FJLT and RI-FJLT still suffers from some distortions, such as Gaussian noise and Gamma correction. One solution is to further find other features that are robust to these attacks/manipulations and incorporate them into the proposed scheme to enhance the performance. Future work will include how to incorporate other robust features (such as the popular SIFT-based features) and secure hashing algorithms to optimize the content-based fingerprinting framework and at the same time explore efficient hierarchical decision-making schemes for identification.

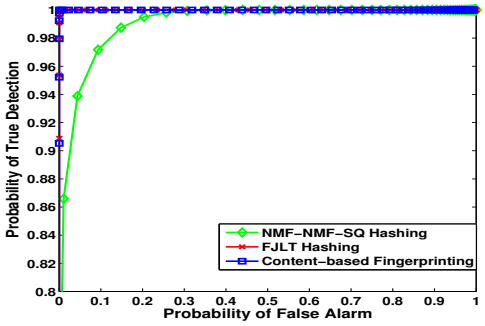
Furthermore, we plan to explore the variations of the current FJLT hashing. Similar to the NMF-based hashing approach (referred as NMF-NMF-SQ hashing in [74]) where the hash is based on a two-stage application of NMF, we can modify the proposed FJLT hashing into a two-stage FJLT-based hashing approach by introducing a second stage of FJLT as follows: Treat the intermediate hash IH as a vector with length $k \times N$, and then reapply FJLT to obtain a representation of the vector IH with further dimension reduction. Compared with our current one-stage FJLT-based hashing, the length of intermediate hash IH could be further shortened by the second FJLT and the security would be enhanced in the two-stage FJLT

hashing. However, the robustness of a two-stage FJLT-based hashing under attacks such as cropping may degrade, since now each component in the modified hash vector is contributed by all the subimages by random sampling. Therefore, the distortion of local information in one subimage could affect the whole hash vector rather than a couple of hash components. The computation cost can also be a concern. We will investigate these issues in the future work.

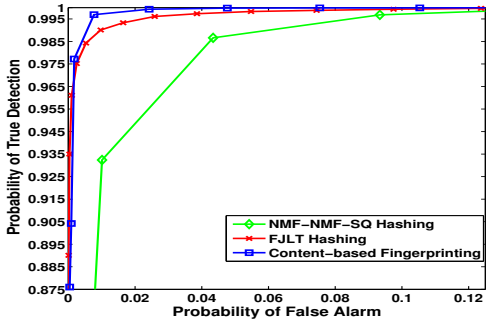
Another concern that is of great importance in practice but is rarely discussed in the context of image hashing is automation. Automatic estimation/choice of design parameters removes the subjectivity from the design procedure and can yield better performances. For instance, algorithms for automating the design process of image watermarking have already been implemented in the literatures [19, 86, 87]. However, to our knowledge, this automated solution has not yet been explored in the context of image hashing. Our preliminary study in [25] demonstrated that using a genetic algorithm (GA) for automatic estimation of parameters of the FJLT hashing could improve the identification performance. However, choosing the appropriate fitness function is challenging in automated image hash. We plan to investigate different fitness functions and how the GA algorithm can incorporate other factors (such as keys) and other constraints (such as the hash length).



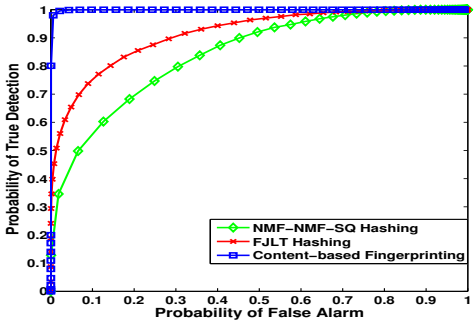
(a) ROC curves under Gaussian noise



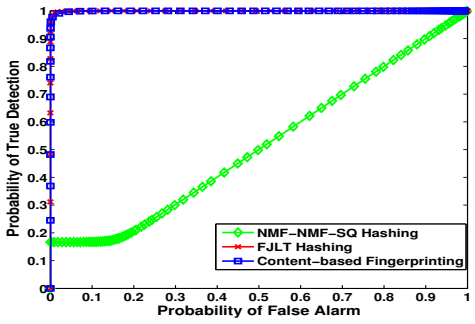
(b) ROC curves under speckle noise



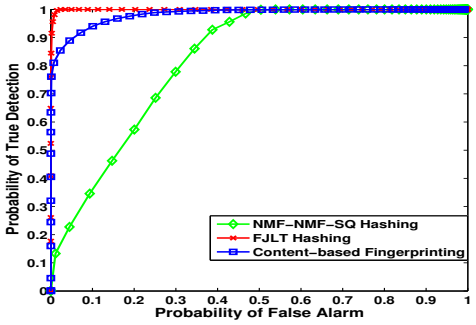
(c) ROC curves under salt & pepper noise



(d) ROC curves under rotation attacks



(e) ROC curves under cropping



(f) ROC curves under Gamma correction

Figure 3.8: The ROC curves of NMF-NMF-SQ hashing, FJLT hashing, and content-based fingerprinting under six types of attacks respectively. (a) Gaussian noise; (b) Speckle noise; (c) Salt&pepper noise;(d) Rotation attacks; (e) Cropping; (f) Gamma correction.

Chapter 4

Perceptual Image Hashing Based on Shape Contexts and Local Feature Points

4.1 Introduction

As a tradeoff between geometric invariance and robustness against classical attacks, Monga *et.al* [73] proposed an image hashing method using local feature points, which have been widely studied for object recognition and image retrieval applications in computer vision and pattern recognition communities [95]. Feature points are local image patterns which differ from their immediate neighbourhoods according to some defined characteristics, such as corners (via Harris), blobs (via Hessian), and salient regions (via MSER). The desirable property of feature points is their invariance under large geometric transforms. [73] extracted local feature points detected by end-stopped wavelets and generated the hashes using adaptive quantization based on the probability distribution of the features. This histogram-based hash is robust against JPEG compression, small rotation and scaling as well as some other image processing attacks. However, this hash ignores the local distribution of the detected feature points.

In this chapter, we developed a new image hashing algorithm using local feature points to overcome the concerns presented above. The contributions are mainly twofold: First, since feature point detection is critical in image hashing in terms of robust feature extraction, we propose using the popular scale invariant feature transform (SIFT) [55] to detect robust feature points

and incorporating the Harris criterion to select the most stable points which are less vulnerable to image processing attacks. Secondly, based on these robust feature points, to characterize also local information, we introduce the shape contexts [9] into hash generation to represent the geometric distribution of the detected feature points. Also, the proposed shape contexts based hashes could be applied to detect and localize content tampering, since the spatial structure of the image content has been embedded into the hash. Part of the work was presented in [62].

Although the proposed hashing shares some ideas with the popular spatial bag-of-words model [7, 15] in large scale image retrieval, including the detection of local feature points such as SIFT, and the incorporation of geometric distribution of these local feature points into the matching and retrieval, their application scenarios are different and thus require different concerns. The bag-of-words model is proposed to retrieve the images with similar objects or from same categories, where the major concern is how to deal with images taken from different viewpoints or with occlusions. Therefore a large visual word vocabulary of local feature descriptors is usually generated using clustering and such huge dimensional feature vectors are used to retrieve the images with similar contents. However, since image hashing mainly aims at protecting the copyright of digital images, a compact and secure signature is generated to represent each image. Also different distortions are of interest in imaging hashing. Its robustness is evaluated against image distortions arising from transmission noise, lossy compression and geometric attacks etc. Such post-processing distortions and attacks generally don't change the image content perceptually and introduce viewpoint changes or large occlusions.

4.2 Robust Local Feature Points

Local features, such as corners, blobs and regions, have been widely used for object detection, recognition and retrieval purposes in computer vision. The intrinsic advantages of these local features are their invariance under geometric transforms. However, their robustness against classical attacks,

especially additive noising and blurring, is limited. A comprehensive review of the state-of-art local features can be found in [95]. Among various local feature detectors and descriptors, SIFT [55] was shown to be relatively optimal considering the tradeoff between robustness, distinctiveness and efficiency. We will briefly review SIFT first and then propose incorporating the Harris criterion to improve its robustness against classical image processing attacks. The extracted robust local features will then be used to generate hashes in Section 4.3.

4.2.1 Scale Invariant Feature Transform Review

SIFT mainly consists of three steps: scale-invariant points detection and localization, orientation assignment, and local descriptors generation.

Scale-invariant points detection and localization

The local feature points detected as the candidates of scale-invariant keypoints are based on the searching for local extrema in a series of difference-of-Gaussian (DOG) images in the scale space σ.

The construction of DOG is proceeded as follows: Image $I(x, y)$ is first convolved with a series of Gaussian kernel functions $G(x, y, \sigma)$ with consecutively incremental scales $\sigma = \{\sigma_1, \sigma_2, ..., \sigma_n\}$, where $\sigma_1 < \sigma_2 < ...\sigma_n$:

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y), \tag{4.1}$$

where the 2-D Gaussian kernel function is

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}. \tag{4.2}$$

Then, a DOG is produced by two Gaussian blurred images with nearby scales $k\sigma$ and σ as

$$\begin{aligned} D(x, y, \sigma) &= L(x, y, k\sigma) - L(x, y, \sigma) \\ &= (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y). \end{aligned} \tag{4.3}$$

With the series of DOG images, the local maximum and minimum are

detected as the candidates of keypoints by comparing each pixel to its 26 neighbours in 3×3 regions at the current and adjacent scales [55]. The final locations of keypoints are localized to sub-pixel accuracy by fitting a 3D quadratic function to the selected candidates to determine the interpolated position of the maximum, which rejects some candidates of keypoints with low contrast. Furthermore, the Hessian matrix is computed at the location and scale of each candidate of keypoints. The ones with large principal curvature are rejected for eliminating edge response.

Essentially, the DOG detector could be attributed to the detector for blob structures in the image content, since it provides a close approximation of the scale-normalized Laplacian of Gaussian [55],

$$G(x, y, k\sigma) - G(x, y, \sigma) \approx (k-1)\sigma^2 \nabla^2 G. \tag{4.4}$$

Substituting Eqs. 4.4 into Eqs. 4.3 and using the property of convolution, we could obtain that

$$\begin{aligned} D(x, y, \sigma) &\approx (k-1)\sigma^2 \nabla^2 G * I(x, y) \\ &= (k-1)\sigma^2 G * \nabla^2 I(x, y), \end{aligned} \tag{4.5}$$

where $\nabla^2 I(x, y)$ is the Laplacian operator commonly used to detect edges and corners in images. Generally the 2-D Laplacian operator, $\nabla^2 = \partial^2/\partial x^2 + \partial^2/\partial y^2$, is not isotropic when it is discretized as in image processing. However, we can see that the difference of Gaussian in Equ. 4.5, which is an approximation of Laplacian of Gaussian, is isotropic, since $\nabla^2 G$ is rotationally invariant. In this sense, the DOG provides better robustness against geometric transforms of images compared with other gradient-based feature points detectors such as Harris and Hessian etc.

Orientation assignment

Orientation assignment for each keypoint is very important, since the corresponding descriptor can be represented relative to this orientation and give rise to rotation invariance. It is determined by the peak of the orienta-

tion histogram formed by the gradient orientations of the detected keypoint within its neighbourhood. The orientation histogram consists of 36 bins covering 360 degree range of orientations and weighted by the magnitude of the corresponding gradient and a Gaussian circular window, which is to reinforce the weights of gradients in the centre of neighbourhood and improve the robustness against additive noise. The details could be found in [55].

SIFT descriptor

Based on the position, scale and orientation of each keypoint, the corresponding descriptor is generated within its local region of the corresponding blurred image in scale space. The 16×16 local neighbourhood of the keypoint is divided into 4×4 subregions and rotated relative to its orientation. Within each subregion, the gradient magnitude and orientation are computed, and then the magnitudes weighted by a Gaussian circular window are accumulated into the orientation histogram with 8 directions. Therefore, each keypoint has a descriptor with $4 \times 4 \times 8 = 128$ dimensions. The SIFT descriptor has been shown to provide satisfied distinctiveness for point matching and robustness against image processing attacks and geometric transforms [71]. Recently, many works were proposed to improve the distinctiveness of SIFT descriptor, such as GLOH [71], PCA-SIFT [107], SURF [8], but they are still based on the DOG detector. Since we believe that the robustness of keypoint detector is more important for image hashing, we use the original SIFT descriptor in this work.

4.2.2 Robust Keypoints Detection Using Harris Criterion

To design a robust image hashing against various attacks, robust feature extraction is the most important step. Although the DOG detector of SIFT provides satisfying performances under geometric transforms, its poor robustness against attacks such as additive noise and blurring limits its direct applications in image hashing. As shown in Figure 4.1 (a)-(d), it is clearly noted that some false positive keypoints are detected in images with additive noise and some true keypoints are missed since the blurred image loses

(a) 578 (b) 692

(c) 436 (d) 836

(e) 81 (f) 83

(g) 73 (h) 87

Figure 4.1: A performance comparison of keypoints detection between SIFT in {(a), (b), (c), (d)} and the proposed SIFT-Harris in {(e), (f), (g), (h)} on the original image, the images under Gaussian noise (GN) ($\sigma = 0.1$), blurring (GB) ($\sigma = 0.5$, 5×5 window) and JPEG compression (QF=10) attacks. The quantities are the total numbers of the keypoints detected in the corresponding images by SIFT and the proposed SIFT-Harris detector.

some details. To extract robust local features, it is desired to select the most stable keypoints under various distortions and attacks.

By carefully reviewing the survey on feature point detectors [95], we note that Harris corner could provide stable detection performance with high repeatability and localization accuracy under various distortions and geometric transform, though it is still sensitive to additive noise. Therefore we propose incorporating the Harris criterion to select the most stable SIFT keypoints for image hashing.

The Harris detector/criterion [33] is based on the auto-correlation matrix, which represents the gradient distribution within a local region of the selected point. For an image $I(x, y)$, the auto-correlation matrix M at point (x, y) is represented as:

$$M = \sum_{x,y} w(x, y) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} \tag{4.6}$$

where $w(x, y)$ is a window to determine the accumulated region, and I_x and I_y are the image gradients in x and y axis respectively. To make the matrix isotropic, we use the Gaussian kernel function as the weighted window. Generally, if both eigenvalues, λ_1 and λ_2 of matrix M, are sufficiently large positive values, this point (x, y) is a corner point. Instead of computing the two eigenvalues, Harris [33] proposed an alternative criteron to evaluate the corner points as

$$H = \lambda_1 \lambda_2 - \kappa(\lambda_1 + \lambda_2)^2 = det(M) - \kappa \; trace^2(M) \tag{4.7}$$

where κ is a coefficient with value $0.04 \sim 0.15$ empirically. We set its default value as 0.06.

Given a set of SIFT points $P = \{p_i(x, y, \sigma, \theta)\}_{i=1}^{N}$, where x and y are the coordinates and σ and θ are the scale and orientation parameters respectively, we compute the Harris response $H_i^{\sigma}(x, y)$, where σ means the standard deviation of the Gaussian kernel window used to compute the auto-correlation matrix M, and set the threshold to select robust SIFT points as

$$Thre = \frac{\alpha}{N} \sum_{i=1}^{N} H_i^{\sigma}(x, y), \tag{4.8}$$

where α is an adjustable parameter to control the robust points selection. Empirically $\alpha \in [0.1, 0.5]$ and we choose 0.5 as the default value. With this threshold, we could achieve stable keypoints detection even in images with additive noise or blurring, as illustrated in Figure 4.1 (e)-(h). The underlying reason is that such a thresholding helps keeping the most stable local patterns with higher gradients distribution but rejecting the keypoints with lower gradient distribution, which are more likely introduced by additive noise. Also, the Harris-based threshold is self-adaptive and can yield relatively stable detection performance.

4.2.3 Detection Evaluation

To further illustrate the effect of Harris criterion on robust SIFT keypoints selection, we define a robust function F to evaluate the performance of SIFT and SIFT-Harris detectors. Let P_o be the set of keypoints detected from the original image and P_d be the set of keypoints detected from its distorted copy, we define the robust function F as:

$$F = \frac{|P_o \bigcap P_d|}{|P_o \bigcup P_d|} \tag{4.9}$$

where $|\cdot|$ means the cardinality of a set, which is a measure of the number of distinct elements of the set. When the value of F approaches 1, it means that we have exactly the same set of keypoints detected from both the original image and its distorted copy. This F value is a criterion to evaluate the stability of detected keypoints under various distortions.

The result comparison of SIFT and SIFT-Harris detectors are summarized in Figure 4.2. We mainly focus on the content-preserving distortions including additive noise (e.g. Gaussian noise (GN) with var=0.0025, salt & pepper noise (SPN), and speckle noise (SN) with var=0.005), blurring (Gaussian blurring (GB) with $\sigma = 2.5$, circular blurring (CB) with ra-

dius=1, motion blurring (MB) with len=2 and $\theta = 45^0$), JPEG compression (JP) with QF=10, rotation (RT) with $\theta = 5^0 \sim 25^0$ and scaling (SC) with $factor = 0.5 \sim 1.5$. The F values of SIFT and SIFT-Harris detectors are averaged based on the tested results of 20 images under above attacks. Obviously, the SIFT-Harris detector outperforms the original SIFT detector over most content-preserving distortions and provides slightly worse performances under salt & pepper noise and rotation attacks. Hence, the Harris criterion mainly boosts the detection performance under classic signal processing attacks but provides less help on rotation attacks. In addition, we also compare with the end-stopped wavelet (end-stopped) approach, which is used to detect the local feature points in [73]. We note that generally the average F values of the end-stopped wavelet approach under content preserving distortions are close to but slightly worse than those of the original SIFT detector, and are consistently much lower than those of the proposed SIFT-Harris detector. In conclusion, the proposed SIFT-Harris detector could yield more stable local feature points.

Moreover, we also investigate the benefit of robust keypoints against content preserving manipulations for the purpose of content identification. Since it is unlikely to exactly obtain the identical set of keypoints detected in both the original and distorted images, [75] proposed using the Hausdorff distance that is insensitive to partial failure of keypoints detection to measure the similarity between the coordinates of two sets of feature points detected in the original and distorted images. Similarly, we employ the Hausdorff distance to compare the proposed SIFT-Harris detector with the end-stopped wavelet detector [73] in terms of the robustness against content preserving manipulations. We select 8 benchmark images (e.g. Lena, Baboon, Peppers etc.) and list the average Hausdorff distances between original images and modified copies under 12 different attacks, as shown in Table. 4.1. Here the vectors are simply the coordinates of the top 20 detected stable keypoints. We note that the average Hausdorff distances of keypoints detected by the SIFT-Harris detector are generally smaller than those of the end-stopped wavelet detector except for the cropping and shearing cases where the distances by SIFT-Harris are slightly higher. The proposed SIFT-Harris

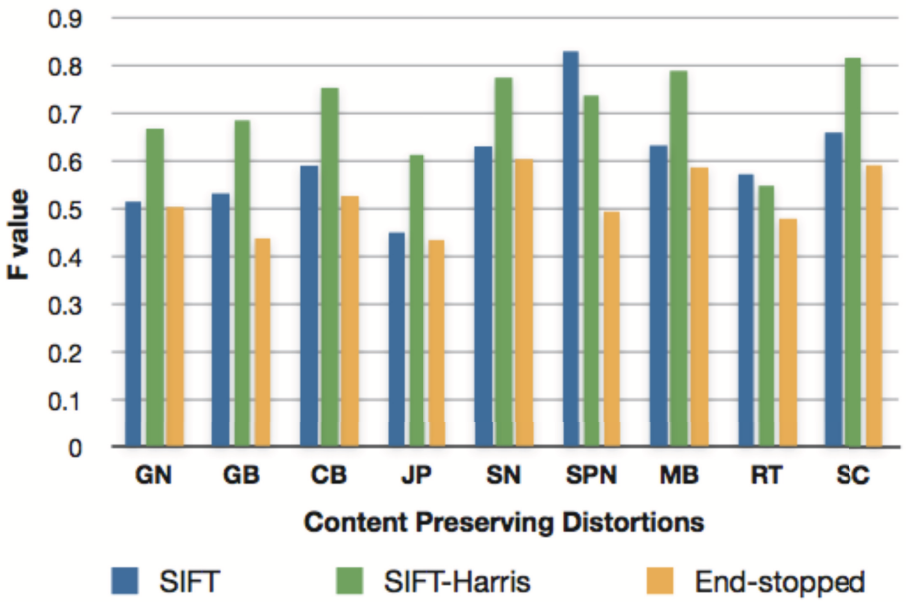


Figure 4.2: The average F values of the end-stopped wavelet [73], the SIFT detector and the proposed SIFT-Harris detector over 20 images under 9 types of content-preserving distortions.

approach is relatively more robust under most content preserving manipulations, especially for blurring and additive noise attacks.

Although similar to [42], we could simply use the coordinates of the detected stable keypoints to directly form the image hashes and such hashes have shown good robustness against geometric attacks [42], such hashing is not relatively compact and its lack of security is a concern in practice. Monga *et.al* in [73] further proposed a global histogram scheme to generate image hashes based on the wavelet coefficients of the detected feature points and introduced pseudo-randomization to enhance the security of the proposed hashing. However, though being robust against compression and geometric attacks, it is sensitive to blurring and noising attacks. Also, the global histogram doesn't take the local distribution of the feature points into consideration. Therefore, we plan to seek an alternative way to fully take advantages of the robust local feature points.

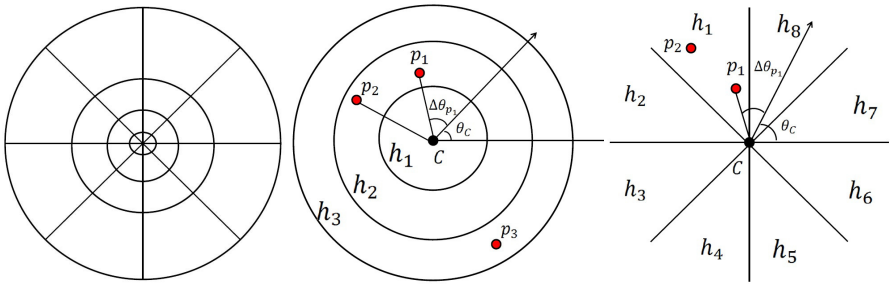
Table 4.1: Average Hausdorff distances between the coordinates of the top 20 keypoints detected in the original image and manipulated copies using the proposed SIFT-Harris and end-stopped [73] detectors.

Manipulation	SIFT-Harris	End-stopped
Additive Noise		
Gaussian Noise ($var = 0.005$)	0.106	0.120
Salt&Pepper Noise ($var = 0.01$)	0.070	0.177
Speckle Noise ($var = 0.01$)	0.067	0.103
Blurring		
Gaussian Blurring ($\sigma = 0.5$)	0.040	0.070
Circular Blurring (radius = 0.1)	0.066	0.085
Motion Blurring ($len = 2$, $\theta = 45^0$)	0.047	0.074
Geometric Attacks		
Rotation ($\theta = 10^0$)	0.257	0.298
Cropping (boundary= 8%)	0.171	0.132
Scaling ($factor = 0.5$)	0.078	0.144
Shearing ($\theta \sim 5\%$)	0.173	0.157
JPEG Compression (QF= 10)	0.107	0.193
Gamma Correction ($\gamma = 1.3$))	0.047	0.065

4.3 Image Hashing Based on Shape Contexts

The application of local feature points such as SIFT to image copy detection is not new [20, 41]. Most of these works mainly detect image copies or near-duplicate copies by matching the high-dimensional local feature descriptors of keypoints. However, this is not feasible in image hashing, where we have to compress the robust features into a compact hash and match the hashes during the detection stage. In [80], the authors proposed a neat way to encode the geometric relationships between SIFT points into a short signatures. However they only investigated the robustness against limited attacks such as rotation, cropping, and compression.

In this section, we propose to use shape contexts, which is a promising method to measure shape similarity for object recognition, to generate image hashes based on the robust local feature points detected in Section 4.2. The motivation of the proposed approach lies in the fact that the distribution



(a) Original shape contexts (b) RSCH (c) ASCH

Figure 4.3: The diagram of the original shape contexts and the proposed shape context hashing: RSCH and ASCH.

of local feature points composes the content structure of images. Treating this geometric structure as an abstract object, we could use a descriptor to represent this structure as an unique signature. Below we will first introduce the basic concept of shape contexts, and then propose two image hashing algorithms using shape contexts and robust feature points, the radial shape context hashing (RSCH) and the angular shape context hashing (ASCH).

4.3.1 Shape Contexts

Given a set of points $P = \{p_i\}_{i=1}^{N}$, which are sampled from the contour of an object, the shape context of the point p_i with respect to the reference point p_c is defined in [9] as

$$h_i(k) = \#\{p_i \neq p_c \ : (p_i - p_c) \in bin(k)\}, \tag{4.10}$$

where $p_i \in P$ and $bin(k)$'s are uniform in log-polar coordinates as shown in Figure 4.3a with the centre located on p_c. Hence, the shape context of each point is a coarse histogram, which represents the relative positions of other points to the reference point. It has been identified that this descriptor is highly robust to shape deformation and offers a globally discriminative characterization, which is effective to solve shape matching and transform model estimation problems.

4.3.2 Shape Contexts-Based Image Hashing

Since shape contexts could provide an outstanding description of the geometric structure of shapes, we could embed the geometric distribution of robust local feature points as well as their descriptors into shape context to generate image hashes. Such hashing is not only based on the image content itself, but also takes its distribution into account. However, the original shape context is designed to be computed for each point sampled from the object contour, meaning that we have N shape contexts if we have N local feature points. Obviously, it provides a rich descriptor to represent the shapes but has to be compressed to be used for hashing directly.

Considering an observation in image content identification and authentication scenarios that generally perceptually insignificant attacks and malicious manipulations on image content would not lead to viewpoint changes, the center of an image is generally preserved (except for some cropping attacks) and relative stable under certain geometric attacks such as rotation, shearing etc. It motivates us that we could generate shape contexts with the reference point in the center and obtain a compact signature for the image. Another reason of avoiding computing shape context for each local feature point in hashing is that keypoints detection could not guarantee to yield exactly the same local feature points when the image is under different attacks and manipulations. As a tradeoff, we propose two new image hashing algorithms, called radial shape context hashing (RSCH) and angular shape context hashing (ASCH), to generate hashes using shape contexts with respect to the central reference point:

Radial Shape Context Hashing (RSCH)

Given a set of local keypoints $P = \{p_i(x, y)\}_{i=1}^{N}$ and their corresponding local descriptors $D = \{d_{p_i}(x, y)\}_{i=1}^{N}$, the basic steps of RSCH are as follows:

- *Step 1*: Given the coordinates of the central point $C = (x_c, y_c)$ and the required length of the hash L, construct bins $B = \{b(k)\}_{k=1}^{L}$ of shape contexts with increment $l = max(x_c, y_c)/L$ in radial direction

of polar coordinates as shown in Figure 4(b),

$$b(k) = \{p_i \in P \ : (k-1)l \leq \|p_i - C\| < kl\}, \tag{4.11}$$

where $\|p_i - C\|$ is the relative distance between p_i and the central point C.

- *Step 2*: Generate pseudorandom weights $\{\alpha_k\}_{k=1}^{L}$ from the normal distribution $N(u, \sigma^2)$ using a secret key. Each α_k is a random vector with 128 dimensions to be consistent with the dimensions of SIFT descriptors.

- *Step 3*: Let $H = \{h_k\}_{k=1}^{L}$ be the hash vector, we have each component h_k as

$$h_k = \sum_{p_i \in b(k)} w_{\left\lceil \frac{L \triangle \theta_{p_i}}{2\pi} \right\rceil} \langle \alpha_k, d_{p_i} \rangle \tag{4.12}$$

where $\triangle\theta_{p_i} = (\theta_{p_i} - \theta_C) \in (0, 2\pi)$ is the relative difference of orientations between p_i and the central point C. We will address the problem of estimating θ_C later. The weight $w_{\lceil L \triangle \theta_{p_i}/2\pi \rceil} \in W = \{w_i\}_{i=1}^{L}$, which is the set of random weights generated from the uniform distribution $U(0.5, 1)$, is introduced to differentiate the points which are located at different orientations but in the same hash bin $b(k)$ along the radial direction.

Angular Shape Context Hashing (ASCH)

Given a set of local keypoints $P = \{p_i(x, y)\}_{i=1}^{N}$ and their corresponding local descriptors $D = \{d_{p_i}(x, y)\}_{i=1}^{N}$, the basic steps of the ASCH are as follows:

- *Step 1*: Given the coordinates of the central point $C = (x_c, y_c)$ and the required length of the hash L, construct bins $B = \{b(k)\}_{k=1}^{L}$ of shape context with increment $l = 2\pi/L$ in angular direction of the polar coordinates as shown in Figure 4(c),

$$b(k) = \{p_i \in P \ : (k-1)l \leq (\theta_{p_i} - \theta_C) < kl\}, \tag{4.13}$$

where $(\theta_{p_i} - \theta_C) = \triangle\theta_{p_i} \in [0, 2\pi)$ as defined above.

- *Step 2*: Generate pseudorandom weights $\{\alpha_k\}_{k=1}^{L}$ from the normal distribution $N(u, \sigma^2)$ using a secret key. Each α_k is a random vector with 128 dimensions to be consistent with the dimensions of SIFT descriptors.

- *Step 3*: Let $H = \{h_k\}_{k=1}^{L}$ be the hash vector, we have each component h_k as

$$h_k = \sum_{p_i \in b(k)} w_{\left\lceil \frac{L\|p_i - C\|}{\|C\|} \right\rceil} \langle \alpha_k, d_{p_i} \rangle \tag{4.14}$$

  where $\|p_i - C\|$ is the same as defined above and $\|C\| = \sqrt{x_c^2 + y_c^2}$ is the normalization factor. The weight $w_{\lceil \|p_i - C\|/\|C\| \rceil} \in W = \{w_i\}_{i=1}^{L}$, which is the set of random weights generated from uniform distribution $U(0.5, 1)$, is introduced to differentiate the points which have different distances to the central point but in the same hash bin $b(k)$ along the angular direction.

Central Orientation Estimation: The central orientation θ_C is significantly important for both ASCH and RSCH, since we need it as a reference direction to calculate the relative difference of orientations between the local feature point p_i and the central point C. However, estimating θ_C based on local gradient distribution is not reliable due to different image processing attacks. Hence, we propose an alternative solution based on Radon transform [48] to estimate an accurate *reference orientation* of C rather than the real orientation.

Radon transform is the integral transform consisting of the integral of a function over straight lines. Given a 2D function $f(x, y)$ and a line p with orientation θ as shown in Figure 4.4, the radon transform of $f(x, y)$ is the integral of orthogonal projections to line p:

$$R_f(p, \theta) = \int_{-\infty}^{\infty} f(p\cos\theta - q\sin\theta, p\sin\theta + q\cos\theta)\mathrm{d}q \tag{4.15}$$

where q is the orthogonal axis of line p. Here we have

$$x = p\cos\theta - q\sin\theta, \tag{4.16}$$

$$y = p\sin\theta + q\cos\theta. \tag{4.17}$$

Based on the radon transform, we could accurately estimate the reference orientation of the central point C as follows:

- *Step 1*: Select the circular neighbourhood of C with radius = 64, and denote this region as a 2D function $f(x, y)$. Then compute the radon transform of $f(x, y)$ from 0 to 2π and get $R_f(p, \theta)$, where $\theta \in (0, 2\pi)$.

- *Step 2*: Choose a reference point p_r on the p axis with a neighbourhood $\Omega \in [p_r - t, p_r + t]$ as shown in Figure 4.4, the reference orientation θ_C could be estimated by

$$\theta_C = \arg\max_{\theta} \sum_{p=p_r-t}^{p_r+t} R_f(p, \theta), \quad \theta \in (0, 2\pi). \tag{4.18}$$

Here θ_C is not the exact orientation of the central point C. However, it provides us a reference orientation, which could be used to calculate the relative differences of orientations between C and other keypoints. Note that the choice of the reference point p_r could not be the middle point, since its projection is symmetric for θ and $\theta + \pi$. In this work, we choose the quarter point p_r as shown in Figure 4.4.

The underlying reason to justify this scheme could be supported by the properties of radon transform. Suppose the 2D function $f(x, y)$ in Figure 4.4 is rotated by an angle ϕ to have $f'(x, y) = f(x\cos\phi - y\sin\phi, x\sin\phi + y\cos\phi)$, we have its radon transform as:

$$R_{f'}(p, \theta) = R_f(p, \theta + \phi). \tag{4.19}$$

When we find the maximum values of $R_f(p, \theta)$ and $R_{f'}(p, \theta)$ for $\theta \in (0, 2\pi)$, the corresponding orientation estimates θ_C^f and $\theta_C^{f'}$ could be used to estimate

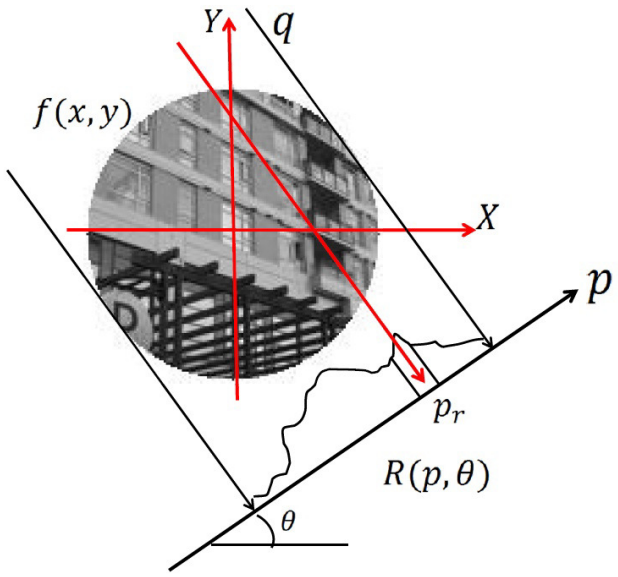


Figure 4.4: The radon transform $R(p, \theta)$ of a 2D function $f(x, y)$

the angle ϕ as:

$$\phi = \theta_C^{f'} - \theta_C^{f}. \tag{4.20}$$

Since the change of the reference orientation θ_C is corresponding to the rotation of the image, it means that the relative differences of orientations between the central point and other keypoints are rotation invariant.

Suppose there is a distortion, i.e. $\tilde{f}(x, y) = f(x, y) + \Delta_f$, which would introduce extra perturbations on the radon transform, i.e. $R_{\tilde{f}}(p, \theta) = R_f(p, \theta) + \Delta_{\tilde{f}}$. Under the assumption that $R_f(p, \theta) \gg \Delta_{\tilde{f}}$, we have

$$\arg\max_{\theta} \sum_{p=p_r-t}^{p_r+t} R_{\tilde{f}}(p, \theta) = \arg\max_{\theta} \sum_{p=p_r-t}^{p_r+t} R_f(p, \theta). \tag{4.21}$$

Since the image usually undergoes content-preserving manipulations and the perturbation Δ_f is much smaller than $f(x, y)$, the assumption usually holds. However, when intensive distortions are considered, this assumption is too strict to be held. Hence, we introduce an adjustable factor t, which is the radius of the integration neighbourhood $\Omega \in [p_r - t, p_r + t]$ to reinforce

the assumption. A larger t would guarantee the assumption more stable at the cost of extra computation. We use $t = 3$ in our later experiments.

4.4 Experimental Results and Analysis

We plan to evaluate the proposed image hashing algorithms from four aspects. The first one is their perceptual robustness against content-preserving manipulations, which is important for content-based image identification and retrieval. It is desired that perceptually identical images under distortions would have similar hashes. The second one is its application to image tampering detection, which aims to localize the tampered contents in images for the purpose of image authentication. The third one is the unpredictability of the proposed hashing measured by differential entropy, a necessary property of the security of hashing algorithms. The last one is the computation cost.

4.4.1 Evaluation of Perceptual Robustness

Database and Content-Preserving Manipulations:

We construct a database with over 107000 images. In this database, there are 1000 original grayscale nature images. For each original image with size 256×342, we generate 106 distorted versions by manipulating the original image according to twelve classes of content-preserving distortions, which include additive noise, blurring, JPEG compression, geometric attacks and brightness changes etc. The motivation to construct such a database is to simulate possible quality distortions of digital images due to the noise in transmission channel, lossy quantization, and geometric manipulations. The details are given in Table 4.2. For the additive noise and blurring attacks, the distortion is introduced based on an acceptable quality range (e.g. PSNR$\geq 23dB$). All the operations are implemented using Matlab.

Table 4.2: Content-preserving manipulations and parameters setting

Manipulation	Parameters Setting	Copies
Additive Noise		
Gaussian Noise	$variance \in (0.0005 \sim 0.005)$	10
Salt&Pepper Noise	$variance \in (0.001 \sim 0.01)$	10
Speckle Noise	$variance \in (0.001 \sim 0.01)$	10
Blurring		
Gaussian Blurring	filter size: 3, $\sigma \in (0.5 \sim 5)$	10
Circular Blurring	radius $\in (0.2 \sim 2)$	10
Motion Blurring	len: $1 \sim 3$, $\theta \in \{0^0, 45^0, 90^0\}$	9
Geometric Attacks		
Rotation	$\theta = 2^0 \sim 30^0$	8
Cropping	boundary: $2\% \sim 10\%$	9
Scaling	factor: $0.5 \sim 1.5$	5
Shearing	$\theta \in (1\% \sim 10\%)$	10
JPEG Compression	Quality Factor $\in (10 \sim 50)$	5
Gamma Correction	$\gamma \in (0.7 \sim 1.3)$	10

Identification and Evaluation Measures

The evaluation for the perceptual robustness of the proposed image hashing algorithms is conducted in two aspects: identification accuracy and receiver operating characteristics (ROC) analysis.

Identification accuracy: Let $S = \{s_i\}_{i=1}^N$ be the set of original images in the database, we define a corresponding hash space $H(S) = \{H(s_i)\}_{i=1}^N$, where $H(s_i) = \{h_1(s_i), h_2(s_i), .., h_n(s_i)\}$ is the hash vector with length n for image s_i. We apply the Euclidean distance $D(H(s_1), H(s_2))$ as the performance metric to measure the discriminating capability between two hash vectors $H(s_1)$ and $H(s_2)$. Given a query image Q, we first generate its hash $H(Q)$ and calculate its distance to each original image in the hash space $H(S)$. Intuitively, the query image Q is identified as the $\hat{i}$th original image as:

$$\hat{i} = \arg\min_i \{D(H(Q), H(s_i))\}. \tag{4.22}$$

The simple identification process described above can be considered as a

special case of the K-nearest-neighbour classification approach with $K = 1$. Here K is set as 1 because we only have one copy of each original image in the current database. For a more general case, if we have K multiple copies of each original image with no distortion or with only slight distortions, we could adopt the K-nearest-neighbour (KNN) classifier. The identification accuracy is the fraction of the distorted copies of images that are correctly classified. It's worth mentioning that the real-value hashes could be further encoded into binary strings and the Hamming distance could be used as the distance measure, which is more computationally efficient than the Euclidean distance.

Receiver operating characteristics analysis: We also study the ROC curve [26] to illustrate the identification performances of the proposed image hashing algorithms and compare them with the state-of-art NMF hashing and FJLT hashing. The ROC curve depicts the relative tradeoffs between benefits and cost of the identification and is an effective way to compare the performances of different hashing approaches. To obtain ROC curves, we define the probability of true identification $P_T(\xi)$ and probability of false alarm $P_F(\xi)$ as

$$P_T(\xi) = Pr(D(H(I), H(I_M)) < \xi) \tag{4.23}$$

$$P_F(\xi) = Pr(D(H(I), H(I'_M)) < \xi) \tag{4.24}$$

where ξ is the identification threshold. Images I and I' are two distinct original images and the images I_M and I'_M are manipulated versions of I and I', respectively. Ideally, we hope that the hashes of the original image I and its manipulated version I_M should be similar and thus be identified accurately, while the distinct images I and I'_M should have different hashes. In other words, given a certain threshold ξ, a better hashing should provide a higher $P_T(\xi)$ with a lower $P_F(\xi)$. Based on all the distances between manipulated images and original images, we could generate a ROC curve by sweeping the threshold ξ from the minimum value to the maximum value.

Table 4.3: Identification accuracy performances by RSCH, ASCH, R&A SCH, NMF, FJLT, RI-FJLT hashing algorithms under different attacks (the length of the hash vector $L = 20$).

Manipulations	RSCH	ASCH	R&A SCH	NMF	FJLT	RI-FJLT
Additive Noise						
Gaussian Noise	83.01%	91.22%	91.67%	99.5%	100%	86.53%
Salt&Pepper Noise	89.77%	92.38%	92.95%	99.6%	100%	99.85%
Speckle Noise	93.5%	95.39%	96.42%	99.58%	100%	86.74%
Blurring						
Gaussian Blurring	78.99%	90%	88.9%	99.6%	100%	31.45%
Circular Blurring	83.8%	90.83%	89.6%	99.58%	100%	53.79%
Motion Blurring	96.08%	98.27%	98.19%	99.6%	100%	72.59%
Geometric Attacks						
Rotation	82.61%	88.76%	90.75%	30.15%	59.99%	84.68%
Cropping	93.17%	96.93%	96.54%	50.4%	95.41%	69.19%
Scaling	77.8%	90.02%	88.18%	99.5%	100%	81.62%
Shearing	78.88%	89.72%	92.26%	72.6%	98.25%	90.91%
JPEG Compression	91.8%	95.72%	96.18%	99.6%	100%	88%
Gamma Correction	93.91%	95.97%	95.57%	0.13%	61.65%	0.4%

Results

We test the proposed image hashing approaches, the RSCH and ASCH. The selected length of the hash vector of RSCH and ASCH is $L = 20$. With each component being 2-byte, the total hash length is 320 bits, which is relatively short. We also compare the proposed schemes with the current state-of-art image hashing algorithms using NMF [74] , FJLT and rotation invariant FJLT (RI-FJLT)[60]. The default parameters of NMF, FJLT and RI-FJLT hashing could be found in [60].

First, we illustrate the identification accuracy of different hashing approaches in Table. 4.3. It is desired that the images with content-preserving distortions can still be correctly classified to the corresponding original image, no matter what kinds of manipulations are taken. From the results, we could observe that the NMF hashing and FJLT hashing are still superior to the proposed local image hashing algorithms using feature points under additive noise, blurring, scaling and compression attacks, although the proposed RSCH and ASCH achieve comparable identification performances. The underlying reason is that NMF and FJLT hashing generate hashes by

extracting robust features from pre-defined image patches, which are determined by the secret key. Since the locations and sizes of image patches are fixed and invulnerable to the additive noise and blurring attacks etc., the essential features are preserved when generating hashes using matrix factorization or random projection. While in the proposed approaches, detection of local feature points is sensitive to the additive noise and blurring attacks. From the test results in Figure 4.2, we can see that the identification performance is still relatively sensitive to these distortions, even though we introduced the Harris criterion to select the most stable keypoints and obtained some improvements. Hence, the local features extracted within the neighbourhood of keypoints are not so stable compared with the pre-defined and fixed image patches in NMF and FJLT.

The advantages of generating hashes based on local feature points lie in the robustness against geometric transforms, especially the rotation attacks. Since the locations to extract robust features (e.g. SIFT descriptor) are determined by detected keypoints, the corresponding hashes are invariant to rotation transforms and the proposed RSCH and ASCH could achieve better identification accuracy, as shown in Table. 4.3. In contrast, the NMF and FJLT hashing approaches are sensitive to rotation attacks due to the changes of pre-defined locations arising from the transform of coordinates. Here we especially added the rotation-invariant FJLT (RI-FJLT) [60] as a comparison. The RI-FJLT combines FJLT and Fourier-Mellin transform [92] to mainly focus on improving the robustness of hashing against rotation attacks at the cost of sensitivity to blurring and noising attacks. From Table. 4.3, the identification performance of the proposed schemes is better than the RI-FJLT. As for the cropping attacks, since the cropped areas are mainly the boundaries of images, where few keypoints are located, the performance of the proposed approach is less affected. But for NMF, FJLT, and RI-FJLT, the performance would degrade when the pre-selected patches are located close to the boundaries of images. For the scaling attacks, since the tested image would be first scaled to a default size (e.g. 256×342) due to the prescaling step, some details would be lost during the downsampling or upsampling process and the effect would be similar to blurring attacks.

The results in Table. 4.3 support this statement.

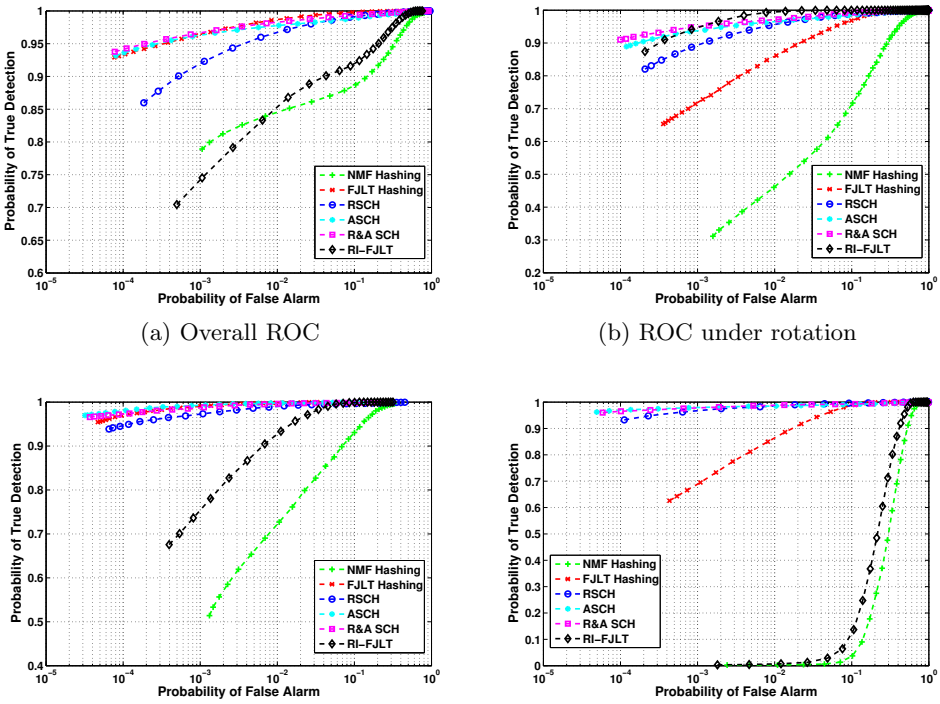
The proposed image hashing approaches can achieve better performances under Gamma correction, probably due to that the SIFT descriptors are invariant to brightness changes. SIFT descriptors are essentially gradient histograms, which are designed to represent relative difference of pixel values within the neighbourhood of detected keypoints. Therefore the brightness changes would have less effect on the gradient distribution. While for the hashing that extracts features from image patches directly as in NMF, FJLT, and RI-FJLT hashing, the brightness changes would inevitably introduce additional distortions in the matrix factorization and random projection process and thus degrade the identification performances. A possible solution to improve the performances of the original NMF and FJLT hashing is to incorporate some pre-processing steps that transform the image patches into luminance invariant domain before feature extraction.

Generally, we observe that ASCH relatively outperforms RSCH, which indicates that the distribution of feature points in the angular direction has better discriminative capacity than the distribution in the radial direction. Intuitively considering the distributions on both directions may further improve the performance on geometric attacks. We therefore propose a simple joint RSCH-ASCH scheme by avoiding losing the track to the geometric distribution of local patterns. To have a fair comparison by maintaining the same hash length in the joint hash, we generate RSCH and ASCH hashes with length $L/2$. For an image, given the RSCH hash $H_r = \{h_r(i)\}_{i=1}^{L/2}$ and the ASCH hash $H_a = \{h_a(i)\}_{i=1}^{L/2}$, we simply define the joint RSCH-ASCH hash $H_{r\&a} = \{h_{r\&a}(i)\}_{i=1}^{L}$ by concatenating two hash vectors as:

$$h_{r\&a}(i) = \{h_r^{\pi}(1)..., h_r^{\pi}(L/2), h_a^{\pi}(1)..., h_a^{\pi}(L/2)\} \tag{4.25}$$

where $H^{\pi} = \{h^{\pi}(i)\}$ means an arbitrary permutation of $H = \{h(i)\}$ determined by a secret key, which is incorporated to further enhance the security. The corresponding identification accuracy is shown in Table. 4.3. By preserving the same length $L = 20$ as RSCH and ASCH, the joint RSCH-ASCH achieves better identification performances under most attacks, and yields

slight degradations under blurring and Gamma correction compared with ASCH.



(a) Overall ROC

(b) ROC under rotation

(c) ROC under cropping

(d) ROC under Gamma correction

Figure 4.5: The ROC curves of the proposed shape contexts based image hashing approaches when compared with the state-of-art NMF, FJLT, and RI-FJLT hashing approaches.

We then present a statistical comparison of different image hashing approaches by studying the corresponding ROC curves, as shown in Figure 4.5. Since the major improvements of the proposed hashing schemes lie in the robustness against geometric attacks and brightness changes, we mainly present the ROC curves under these manipulations. With the same probability of false alarm $P_F(\xi)$, a better hashing approach could achieve a higher probability of true identification $P_T(\xi)$. In other words, the ROC curve is to measure the similarity of hashes between true query images and original images under a selected false classification rate. ROC curves provide a

tradeoff between the true retrieval and misclassification to select, when a uniform threshold is applied for identification.

It is noted that the proposed hashing approaches achieve the best overall robustness as shown in Figure 4.5a under all manipulations listed in Table 4.3, which is consistent with the reported identification performances. The ASCH and the joint RSCH-ASCH are slightly better than FJLT hashing, which mainly arises from the improvements in geometric attacks and brightness changes. We also illustrate the ROC curves under rotation, cropping and brightness changes respectively to show the improvements arising from the geometric invariance of local feature points and illuminance invariance of SIFT descriptors. Compared with RSCH and ASCH, the joint RSCH-ASCH scheme takes the geometric distributions of local feature points in both radial and angular directions into account and achieves better overall identification performance.

4.4.2 Evaluation of Tampering Detection

Since the proposed shape contexts based hash is based on the local distribution of the keypoints rather than the one based on global histogram as in [73], the geometric structure of the image, which is represented by local feature points, is embedded into the compact hash. Therefore, the proposed hash can distinguish images with similar contents but different structures and thus could be useful for image tampering detection. Image tampering detection is to localize the artificial modifications in image content. Some recent works [58, 80, 93] have illustrated the use of multimedia hashes for identifying the artificial tampering image content.

Given an image I and its tampered copy I^T, the tampered area B^T could be localized by using their RSCH $H_r = \{h_r^k(I)\}_{k=1}^L$ and $H_r^T = \{h_r^k(I^T)\}_{k=1}^L$ in the radial direction and ASCH $H_a = \{h_a^k(I)\}_{k=1}^L$ and $H_a^T = \{h_a^k(I^T)\}_{k=1}^L$ in the angular direction as

$$B_r = \{bin_r(k) : \left| h_r^k(I) - h_r^k(I^T) \right| \geq \delta\}, \tag{4.26}$$

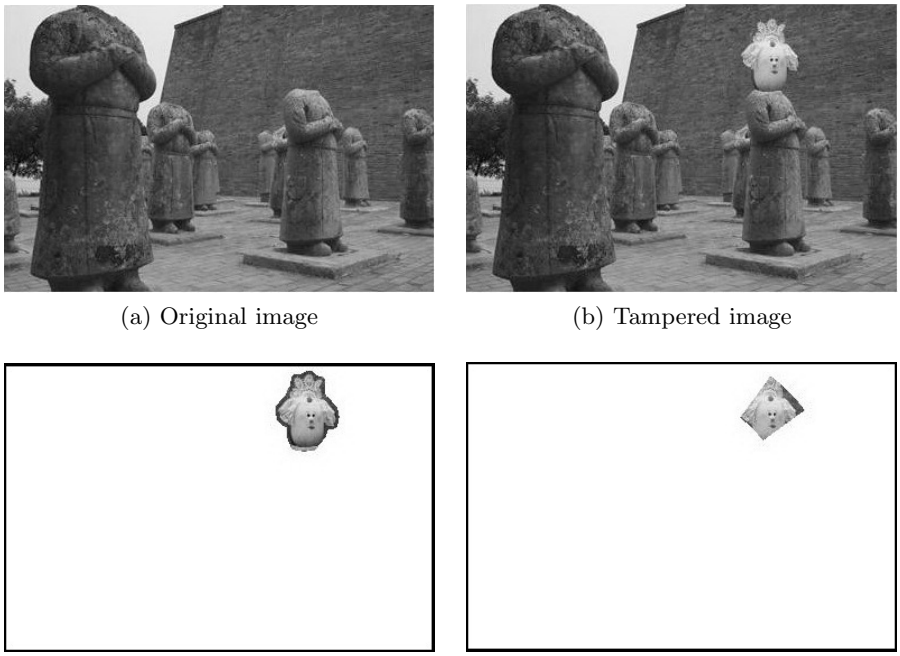
$$B_a = \{bin_a(k) : \left| h_a^k(I) - h_a^k(I^T) \right| \geq \delta\}, \tag{4.27}$$

$$B^T = B_r \cap B_a, \tag{4.28}$$

where B_r and B_a are the sets of histogram bins in radial and angular directions, whose corresponding normalized point-wise distances of hashes between the image I and its tampered copy I^T are larger than the threshold δ.

The performance of the proposed approach on tampering detection mainly relies on two factors. The first one is the length of RSCH and ASCH hash vectors, which decides the resolution of tampered area that could be localized. The longer the hashes are, the more precise detections could be achieved. However, minor distortions on the image would also perturb the detection and introduce higher false alarm rates when the histogram bins are too fine. The second one is the threshold δ, which is used to determine the tampered local histogram bins. We investigate these two factors by ROC analysis on a group of tampered images considering both the combination of the different hash lengths of RSCH and ASCH and the threshold δ. We used the CASIA tampered image detection evaluation database [1] and applied the proposed approach on a group of 10 image pairs, which includes the original image and tampered copy. An example is shown in Figure 4.6a and Figure 4.6b. For each image pair, we calculated their RSCH and ASCH hashes with three different combinations (ASCH-15 & RSCH-5, ASCH-20 & RSCH-10, ASCH-25 & RSCH-15) and normalized the hash distances between original images and tampered copies to [0, 1]. The design of the combinations is mainly to make the intersection between RSCH and ASCH be approximately square. Then we generate the ROC curve by sweeping the threshold δ from 0 to 1 as shown in Figure 4.7, in which the true positive rates and false alarm rates are averaged based on the results from 10 image pairs.

From the ROC curves in Figure 4.7, we observed that the coarse histogram bins based on ASCH-15 & RSCH-5 has a higher true positive de-



(a) Original image (b) Tampered image

(c) Ground truth (d) Tampering detection

Figure 4.6: An example of image tampering detection using the proposed shape contexts based image hashing approaches.

tection rate at the cost of a little higher false alarm. This combination is mainly useful when the tampered area is relatively large. The finer histogram bins based on ASCH-25 & RSCH-15 has relatively lower false alarm at low true positive detection rate, while ASCH-20 & RSCH-10 is relatively a good tradeoff between the true positive rate and false alarm. From the ROC curve, a suitable threshold δ could be picked up to yield a relatively high true positive rate with low false alarm (e.g. we set $\delta = 0.5$). An example of tampering detection is shown in Figure 4.6d. Compared with the ground truth in Figure 4.6c, the proposed approach could only roughly localize the tampered region on images, limited by the default shape of histogram bins. Additional features as suggested in [58] could further improve the precise tampering detection. Also, to optimally allocate the histogram bins for RSCH and ASCH, a recursive way using a coarse-to-fine strategy

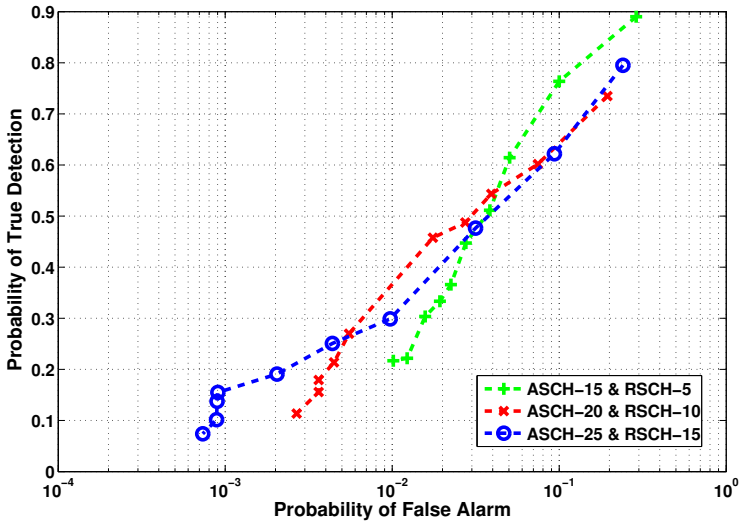


Figure 4.7: The ROC curves for tampering detection

could be applied, though its extra computational cost should be taken into consideration.

It is worth emphasizing that the NMF, FJLT as well as the the global histogram based hashing [73] are not applicable for tampering detection. In this sense, the proposed hashing scheme is more generally applicable.

4.4.3 Unpredictability Analysis

We also investigate the security of the proposed hashing approach in terms of unpredictability that arises from the key-dependent randomization. Higher amount of randomness in the hash values makes it harder for the adversary to estimate and forge the hash without knowing the secret keys. Since it is believed that a high differential entropy is a necessary property of secure image hashing, we evaluate the security of the proposed image hashing in terms of unpredictability by quantifying the differential entropy as in [92].

The differential entropy of a continuous random variable x is given by

$$H(x) = \int_{\Omega} f(x) \log \frac{1}{f(x)} \mathrm{d}x, \tag{4.29}$$

where $f(x)$ means the probability density function (pdf) of x and Ω means the support area of $f(x)$. Since the construction of ASCH is similar to RSCH, we take RSCH as an example to derive the differential entropy of its hash element. Suppose that M keypoints $\{p_i\}_{i=1}^{M}$ are detected within the histogram bin $b(k)$ in RSCH, and each of them has a local descriptor $d_i = \{d_{ij}\}_{j=1}^{N}$ with dimensions N, the hash $h(k)$ is

$$h(k) = \sum_{p_i \in b(k)} w_{\left\lceil \frac{L\triangle\theta_{p_i}}{2\pi} \right\rceil} \langle \alpha_k, d_{p_i} \rangle \tag{4.30}$$

where the pseudorandom weights $\alpha_k = \{\alpha_{kj}\}_{j=1}^{N}$ are generated from the normal distribution $N(u, \sigma^2)$ using the secret key. Here the weights $w_{\left\lceil L\triangle\theta_{p_i}/2\pi \right\rceil} = w_i \sim U(0.5, 1)$ are introduced to differentiate the keypoints in the same histogram bin $b(k)$ of RSCH with different orientations. For simplicity, we don't take its randomness into consideration and treat them as fixed weights. It is known that the sum of normal random variables still follows a normal distribution. Hence $h(k)$ is normal distributed with expectation value and variance as

$$\mu_h = E(h(k)) = \mu \sum_{i=1}^{M} w_i \sum_{j=1}^{N} d_{ij}, \tag{4.31}$$

$$\sigma_h^2 = Var(h(k)) = \sigma^2 \sum_{i=1}^{M} w_i^2 \sum_{j=1}^{N} d_{ij}^2. \tag{4.32}$$

Therefore we have its differential entropy as

$$H(h(k)) = \frac{1}{2} \log(2\pi e \sigma^2 \sum_{i=1}^{M} w_i^2 \sum_{j=1}^{N} d_{ij}^2) bits. \tag{4.33}$$

Obviously, the differential entropy depends on the number of keypoints detected in images and their corresponding descriptors, which are determined by the image content. But for a random variable with the same bounded variance, the differential entropy is maximized when it follows normal distribution. Since the generated hash $h(k) \sim N(\mu_h, \sigma_h^2)$, the differential entropy $H(h(k))$ is maximized, given the detected local feature points as well as the

Table 4.4: The average CPU times required by the proposed SCH, FJLT, and NMF hashing approaches.

Computational cost	SCH	FJLT	NMF
time (s)	3.91	0.638	0.9391

descriptors.

We would like to mention that a higher differential entropy is only one aspect of a secure image hashing [74, 92], which also includes other factors such as key diversity and prior knowledge possessed by adversaries. Also, differential entropy in the continuous form is not invariant under scaling and translation as stated in [42], and may not be a proper interpretation as the uncertainty of the hashing. Therefore, there lacks of a universal measure to comprehensively evaluate the security of image hashing. Interested reader could refer to the references in [42, 64, 92].

4.4.4 CPU Time Cost

Compared with the FJLT and NMF hashing, which use pre-fixed regions of interest determined by a secret key for feature extraction, the major and additional computation cost of the proposed shape contexts based hashing lies in the robust local keypoint detection. Therefore, the computation cost of the proposed hashing is higher than the FJLT and NMF hashing. As an example, we test these approaches on 50 images using a computer with Intel Core i7 CPU (2.67 GHz) and 3GB RAM and report the average computational time in Table 4.4.

4.5 Conclusion

In this chapter, we proposed the shape contexts based hashing approaches using local feature points and investigated their perceptual robustness against a large class of content-preserving manipulations, image tampering detection as well as the security and computation cost issues. Based on the geometric invariance of the state-of-art SIFT keypoints, we incorporate Harris criterion to select a subset of keypoints that are the most stable under ad-

dition noise, blurring, and compression distortions. From the viewpoint of object recognition, the distribution of these robust keypoints composes the content structure of images, we therefore introduce the shape contexts to embed the geometric distribution as well as the corresponding descriptors of keypoints into a short hash vector and propose shape context based image hashing algorithms, i.e. RSCH in the radial direction and ASCH in the angular direction. Since the RSCH or ASCH only considers the distribution of keypoints in one direction, we propose a simple joint RSCH-ASCH as an alternative method to consider the distribution in both directions, which can achieve better overall perceptual robustness. Since the shape contexts based hashes embed the geometric structure of the image, another advantage of the proposed hashing is its applicability in image tampering detection.

However, image hashing using feature points still has limitations when considering the distortions of additive noise and blurring in large scale (e.g. $PSNR < 22dB$). Even for the rotation attacks, since the image is transformed into another coordinate space with interpolations, which locally modify the images without introducing significant perceptual distortion, the detected keypoints are not exactly the same in the original image and its rotated copy and thus we could only get around 90% identification accuracy even using shape contexts hashing, as shown in Table 4.3. Our experience suggests that generating a single type of image hash based on certain features to resist all types of manipulations is highly unlikely, while it is relatively easy to find a specific feature to robustly resist certain manipulations. Obviously, combining multiple hashes in a joint decision making framework can be a promising direction for digital image hashing, as presented in the content-based fingerprinting framework in Section 3.5. However, how to balance the compactness of hashes and its robustness will be the critical issues to deal with and the optimal tradeoff has to be taken into consideration.

Chapter 5

Compressed Binary Image Hashes Based on Semi-Supervised Spectral Embedding

5.1 Introduction

Currently, almost all existing image hashing works focus on intermediate hash generation, which includes the steps of robust feature extraction and signature generation, to generate real-valued image hashes. Since most robust features are usually robust against certain types of distortions and attacks and it is hard, if not feasible, to extract robust features which can resist all types of distortions and attacks, a tradeoff has to be made when designing image hashes robust against different distortions and attacks, such as the classical image processing attacks (e.g. additive noise, blurring, compression, etc.) and geometric attacks (e.g. rotation, cropping etc.) [60, 63]. Therefore, how to efficiently take advantages of different features together to enhance the overall robustness of image hashing, or how to efficiently design a combined, superior image hashing approach based on different types of image hashes, is a topic of great importance in practice but less studied in the current literature. Moreover, few works in the current literatures illustrate how to compress the real-valued image hashes into binary image hashes in the post-processing step, while it is a critical issue for fast retrieval and efficient storage in practice.

In this chapter, we assume the availability of real-valued image hashes and focus on the topic of combining different types of image hashes and jointly compressing them into a short binary image hash. Compared with the real-valued image hashes, the binary image hashes require less memory space, support the Hamming distance similarity measure directly and thus are more suitable for fast retrieval and identification. Also, the generated joint binary hash could enhance the overall robustness against various attacks and distortions by taking advantages of different types of real-valued image hashes. More important, we would like to emphasize that the proposed binary image hashing approach presents a fundamental departure from existing methods in image hashing: Previous image hash generation methods mainly focus on extracting robust features against various distortions and attacks, and the feature compression and post-processing steps are image independent. Basically, no prior knowledge from distinct images or distorted copies is used directly for image hash generation of each image. While in this chapter, for the first time, prior information from virtual image distortions and attacks is explored in image hash generation when compressing the real-valued hashes into the binary ones. More specifically, the proposed binary image hashing scheme takes advantages of the extended hash feature space from virtual distortions and attacks and generates the binary hash for each image based on Laplacian spectral embedding. The contributions of this work include: 1) We first propose compressing the real-valued hashes into binary signatures with the help of virtual prior attacked hash space (VPAHS), which is produced by applying virtual prior distortions and attacks on the training images and generating hashes under such simulated virtual distortions and attacks; 2) We extend the spectral embedding [91, 98] idea into the image hashing area and learn the optimal spectral embedding in a semi-supervised way based on VPAHS to project the real-valued image hashes into a binary one, which could efficiently maintain the overall robustness from different types of real-valued image hashes without increasing the hash length.

5.2 Motivation

5.2.1 Related Literature Review

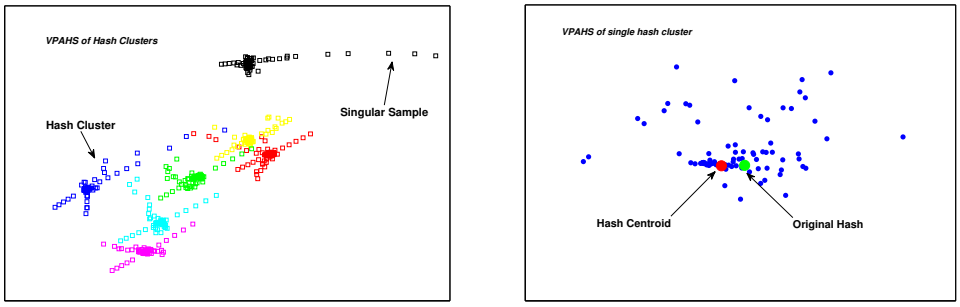
In practice, the intermediate real-valued image hashes are further converted into binary signatures in the post-processing module. In [96], Venkatesan *et al*. used randomized quantizer to quantize statistics of wavelet coefficients and decoded them by the first-order Reed-Muller error-correcting decoder to produce the binary hash signature. Quantization and error-correcting coding (ECC) are efficient ways to enhance the robustness of features by cancelling the influence of small perturbations. Swaminathan *et al*. [92] applied gray coding to obtain the binary sequences of quantized intermediate hashes and compressed them by using the third-order Reed-Muller decoder. Also, distributed source coding, such as Slepian-Wolf [52], and Wyner-Ziv [37, 93] can be applied for intermediate hash compression, which assumes that some side information is available at the decoder side. Although, ECC or distributed source coding is able to correct some distortions in the compressed messages to further enhance the robustness of quantized intermediate hashes, the collision probability between distinct image hashes is also increased, which leads to extra false alarms in practice.

Essentially, consider the intermediate hashes as real-valued hash features, the compression step of image hashing becomes an embedding problem of embedding the real-valued feature space into compact binary hash space, which is one of the most popular research areas nowadays. The main idea of embedding learning is to formulate binary projections such that similar real-valued feature vectors have the similar binary codes, given the distance functions. The state-of-art embedding methods could be mainly divided into two categories: 1) Data independent embedding without learning, such as Locality Sensitive Hashing (LSH) [23], random fourier features [79], minhash [17] etc. Based on the asymptotic theoretical property, similar features can be embedded into the same binary codes with high probabilities using random projections, at the cost of relaxing the retrieval process to an approximate nearest neighbour problem with some tolerant errors. Since the embedding is independent of the data distribution, this category of methods

may not be optimal to certain data with specific distributions, but more suitable for heterogeneous ones. 2) Data dependent embedding with learning, such as Spectral Hashing [98], Kernelized LSH (KLSH) [46], Semantic Hashing [82], Product Quantization [36] etc. In this category, the embedding process is trained to optimally fit specific data distributions and specific distance functions, which produce better binary codes to preserve the local similarity. Recently, some works such as LDAHash [91], sequential projection learning for hashing (SPLH) [97] etc., introduce the label information into embedding learning and render it to a supervised learning process, which produces the binary codes with lower false positive matching rate.

5.2.2 VPAHS and Motivation

In the conventional image hashing approaches, the image hash generation is a robust feature compression and encryption process without any learning stage. Also, the image identification process is generally formulated as a nearest neighbour decision making problem given Euclidean or Hamming distance without employing any advanced supervised classifiers. The underlying reason of generally not including learning in the identification decision making process is as follows: Image hashing is an infinite clustering problem, which takes each original image as a new cluster/class and all its distorted copies are assumed to lie in the neighborhood of the centroid (i.e. represented by the original image). Therefore, if advanced supervised classifiers such as Support Vector Machine (SVM) are used, they have to be re-trained whenever a new original image is registered in the database. This re-training may incur heavy computational burden when thousands of images are registered and thus it is not feasible in practice. In the current literature of image hashing, prior information (e.g. distorted copies of original images) is generally not explored in both the image hash generation process and the decision-making process. We noted one attempt of exploring prior information in decision-making: In [34], the authors uses virtual attacks to generate extended feature sets for training supervised classifiers (e.g., SVM) and detecting image copies, where the classifiers have to be retrained for



(a) VPAHS of multiple hash clusters (b) VPAHS of a single hash cluster

Figure 5.1: The examples of VPAHS and hash clusters based on the FJLT image hashing scheme

new original images. To our best knowledge, we are not aware of similar efforts of exploring prior information in image hash generation.

However, we feel that efficient learning can be incorporated into the image hash generation process by taking advantages of the prior information at the owner side. More specifically, since image hashing is desired to be robust against traditional image processing distortions and some geometric transforms, the variations of the intermediate hashes from distorted copies may follow certain specific distributions, which motivates us to incorporate the data dependent binary embedding idea to fit such specific distributions and further compress the real-valued image hashes into more compact binary hashes. To explore the variations of the intermediate real-valued hashes under certain types of distortions, we could apply virtual prior attacks (e.g. additive noise, blurring, compression etc.) on original images to generate simulated distorted copies as well as the corresponding intermediate real-valued hashes. The hashes of the training original images and their distorted copies compose the so-called virtual prior attacked hash space (VPAHS), which could represent the distribution of the intermediate hashes under certain virtual distortions and attacks.

An example of VPAHS for FJLT image hashing [60] is shown in Fig-

ure 5.1a, where we apply 12 classes of attacks in Table 5.1 on 7 original images and we generate 2-dimensional intermediate hashes for both the original images and their distorted copies. Ideally, we should observe 7 hash clusters lying in the hash space and the hashes of the distorted copies should lie in the neighbourhood of the corresponding original hash. From the example in Figure 5.1a, we can see that most of the distorted copies cluster well to the corresponding original image, mainly because FJLT hashing is robust against a large class of image process distortions such as additive noise, blurring, compressing etc. However, we also note that some distorted copies under geometric attacks (e.g., rotation transforms) distribute far from the corresponding original image (e.g. denoted by "singular sample"), due to the fact that FJLT hashing is sensitive to geometric attacks.

From Figure 5.1b by zooming into one hash cluster, we note a critical observation that the image hash of the original image actually may not be the *centroid* of its cluster. Here we have the following definition:

Definition 1 *For a certain original image and its various distorted copies, suppose the corresponding hashes follow a distribution $p(x)$, the hash centroid is defined as $\tilde{x}$, where*

$$\tilde{x} = \arg\max_{x} p(x). \tag{5.1}$$

This definition of hash centroid is the same as the *mode* of a probability distribution function. We expect that most distorted image hashes should distribute around the hash centroid if the image hashing approach is robust against the corresponding distortions

Obviously, from Figure 5.1b we could observe that the image hash arising from the original image is not the hash centroid, which means that the similarity measures between the distorted hashes and original hashes are biased, if we consider the hashing problem as a clustering problem. Therefore we may need to estimate the exact hash centroids in VPAHS to facilitate the incorporation of the learning stage into image hash generation. Here we propose obtaining an optimal estimate of the hash centroid using kernel density estimation (KDE) [13]. Let the data set $X = \{x_i\}_{i=1}^{n}$ be i.i.d. sampled from an unknown distribution $f(x)$, the estimated distribution function

$\tilde{f}(x)$ can be expressed as

$$\tilde{f}(x) = \frac{1}{nh} \sum_{i=1}^{n} K(\frac{x - x_i}{h}) \tag{5.2}$$

where K is usually chosen as Gaussian kernel and h is the smoothing parameter. Based on the estimated distribution $\tilde{f}(x)$, the corresponding *mode* or *centroid* could be obtained as the value which yields the maximum $\tilde{f}(x)$.

Based on the VPAHS concept and the estimated hash centroids of training images, in the next section, we will introduce the data dependent spectral embedding ideas [91, 98] and propose a novel semi-supervised spectral embedding scheme based on VPAHS to compress the real-valued intermediate image hashes into compact short binary hashes. By fitting the embedding to the specific distribution illustrated based on VPAHS, the generated binary hash is optimal for preserving the similarity of intermediate hashes given the cost function. Also, for new images, the corresponding binary hash codes are easy to compute without re-training the model.

5.3 Proposed Binary Image Hash Generation

5.3.1 Spectral Embedding Ideas Review

Within the current data-dependent embedding methods, spectral hashing (also referred as spectral embedding) has been shown to be one of the state-of-art approaches for compact binary codes construction, especially for compressing the state-of-art SIFT and GIST image descriptors [98]. Suppose we have a set of feature vectors $X = \{\mathbf{x}_i\}_{i=1}^{n}$, where $\mathbf{x}_i \in R^d$, the similarity between each pair of feature vectors is measured by Euclidean distance in terms of Gaussian kernel as $w(i,j) = exp(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{\varepsilon^2})$. The goal of spectral hashing is to find a set of corresponding binary codes $B = \{\mathbf{y}_i\}_{i=1}^{n}$, where $\mathbf{y}_i \in \{-1, 1\}^k$, so that the overall weighted Hamming distances between each

pair of the k-length binary codes are minimized as:

$$\begin{aligned} minimize: \quad & \sum_{i,j} w(i,j) \left\| \mathbf{y}_i - \mathbf{y}_j \right\|^2 \\ s.t.: \quad & \sum_{i=1}^{n} \mathbf{y}_i = \mathbf{0} \\ & \frac{1}{n} \sum_{i=1}^{n} \mathbf{y}_i \mathbf{y}_i^T = \mathbf{I}, \end{aligned} \tag{5.3}$$

where $\mathbf{0}$ means the k-length zero vector and $\mathbf{I}$ denotes the $k \times k$ identity matrix, the constraint $\sum_i \mathbf{y}_i = \mathbf{0}$ guarantees that each bit has equal chance to be -1 or 1, and $\frac{1}{n} \sum_i \mathbf{y}_i \mathbf{y}_i^T = \mathbf{I}$ decorrelates each bit of the codes to minimize the redundancy. Denote by a $n \times k$ matrix $Y = \{\mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_n\}^T$, a diagonal $n \times n$ matrix $D(i,i) = \sum_j w(i,j)$, and a $n \times n$ weight matrix W, whose entries are $w(i,j)$'s. The above problem could be rewritten as

$$\begin{aligned} minimize: \quad & trace(Y^T(D-W)Y), \\ s.t.: \quad & Y^T 1 = 0, \\ & Y^T Y = I. \end{aligned} \tag{5.4}$$

By relaxing the constraint $Y(i,j) \in \{-1, 1\}$, the solutions of the above problem are $\{\mathbf{y}_1^L, ..., \mathbf{y}_k^L\}$, i.e. the k eigenvectors of the Laplacian matrix $L = (D - W)$ associated with the minimal eigenvalues, excluding the one with zero eigenvalue. For the j-th training data $\mathbf{x}_j$, the final binary codes $\tilde{\mathbf{y}_j}$ could be obtained based on the corresponding j-th elements of the eigenvectors $\mathbf{y}_i^L$'s as

$$\tilde{\mathbf{y}_j} = sgn(\mathbf{y}_1^L(j), \mathbf{y}_2^L(j), ..., \mathbf{y}_k^L(j)), \quad \tilde{\mathbf{y}_j} \in \{-1, 1\}^k. \tag{5.5}$$

Spectral hashing has been widely applied for compressing the state-of-art SIFT and GIST image descriptors to benefit fast image retrieval. Essentially, it is a unsupervised binary code learning process, which does not take the label information into account.

Recently, some works such as LDAHash [91] introduce the label information into the cost function for learning embedding and can reduce the

false positive rate by rendering the problem as a supervised learning problem. Given the set of n feature vectors, $X = \{\mathbf{x}_i\}_{i=1}^{n}$, LDAHash first divides them into the positive set of pairs $\mathcal{P}$ (e.g., SIFT descriptors from same objects under different viewpoints) and the negative set of pairs $\mathcal{N}$ (e.g., SIFT descriptors from different objects), then it seeks the affine embeddings of the form

$$\mathbf{y} = sgn(\mathbf{P}\mathbf{x} + \mathbf{t}), \tag{5.6}$$

where $\mathbf{x} \in R^d$ is the feature vector, $\mathbf{P}$ is a $k \times d$ matrix, $\mathbf{t}$ is a $k \times 1$ threshold vector, and $\mathbf{y}$ is the corresponding $k \times 1$ binary vector, which is constructed to minimize the cost function

$$L = \alpha \mathrm{E}\{\left\|\mathbf{y} - \mathbf{y}'\right\|^2 \mid \mathcal{P}\} - \mathrm{E}\{\left\|\mathbf{y} - \mathbf{y}'\right\|^2 \mid \mathcal{N}\} \tag{5.7}$$

where α is a parameter to control the tradeoff between false positive and false negative rates, and $\mathrm{E}\{\cdot \mid \mathcal{P}\}$ and $\mathrm{E}\{\cdot \mid \mathcal{N}\}$ are the conditional expectations on the training set of positive and negative pairs of feature vectors, denoted by $\mathbf{x}$ and $\mathbf{x}'$, respectively. Relaxing the problem L to $\tilde{L}$ by removing the sign function, the cost function becomes

$$\begin{aligned} \tilde{L} &= \alpha \mathrm{E}\{\left\|\mathbf{P}\mathbf{x} - \mathbf{P}\mathbf{x}'\right\|^2 \mid \mathcal{P}\} - \mathrm{E}\{\left\|\mathbf{P}\mathbf{x} - \mathbf{P}\mathbf{x}'\right\|^2 \mid \mathcal{N}\} \\ &= \alpha \cdot trace\{\mathbf{P}\Sigma_{\mathcal{P}}\mathbf{P}^T\} - trace\{\mathbf{P}\Sigma_{\mathcal{N}}\mathbf{P}^T\} \\ &\propto trace\{\mathbf{P}\Sigma_{\mathcal{P}}\Sigma_{\mathcal{N}}^{-1}\mathbf{P}^T\}, \end{aligned} \tag{5.8}$$

where $\Sigma_{\mathcal{P}}$ and $\Sigma_{\mathcal{N}}$ are the covariance matrix of the feature vectors in the sets of positive pairs and negative pairs respectively. The solution of $\mathbf{P}$ to minimize the above cost function $\tilde{L}$ is the k eigenvectors of $\Sigma_{\mathcal{P}}\Sigma_{\mathcal{N}}^{-1}$ associated with the k minimal eigenvalues. Obviously, LDAHash incorporates the label information (e.g. similar features from same objects under different viewpoints and distinct features from different objects) into the embedding learning process to optimize the constructed binary codes, which have lower false positive rates compared with the unsupervised spectral hashing.

As mentioned in Section 5.2.2, image hashing is an infinite clustering problem that takes each original image as a cluster. Because of this, the

unsupervised code learning in image hashing may be more feasible than the supervised learning one, considering the extra burden of re-training for each new image. However, since label information can be helpful for reducing the false positive rate of the learned codes and such prior information is available at the encoder side, we would benefit by exploring such prior information, even though we may not be able to fully utilize it. Therefore, in the next section, we propose a semi-supervised spectral embedding scheme for compressing intermediate hashes into binary hash codes. Here the term "semi-supervised spectral embedding" means the following: We don't fully use the label information (e.g. pairwise labels with similar or dissimilar pairs {-1, 1}) as in supervised learning [91], but we partially explore the soft label information weighted by intra-class and inter-class similarity measures to learn the optimal binary hash embeddings, which could map hashes from perceptually identical images together in the subspace, but map the ones from visually distinct images far apart.

5.3.2 Proposed Semi-Supervised Spectral Embedding (SSE)

Referring to the VPAHS concept illustrated in Figure. 2, the hashes of the distorted images distribute within a specific neighbourhood of the hash of the original image, although the original hash may not be the centroid of the cluster. Intuitively, the hashes within the same neighbourhood should be embedded together and the others be mapped far apart. Recall the weight matrix W with $w(i,j) = exp(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{\varepsilon^2})$, where the parameter ε controls the similarity measure between the pair of data, we could set different ε's to penalize the data pairs in the same neighbourhoods and the ones within different neighbourhoods. Therefore, the proposed cost function is formulated as

$$\sum_{i,j} \left[\alpha \cdot w_1(i,j) \left\| \mathbf{y}_i - \mathbf{y}_j \right\|^2 - (1-\alpha) \cdot w_2(i,j) \left\| \mathbf{y}_i - \mathbf{y}_j \right\|^2 \right] \tag{5.9}$$

$$
\begin{aligned}
where: \qquad & w_1(i,j) \triangleq w_1(\mathbf{x}_i, \mathbf{x}_j) = exp(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{\varepsilon_1^2}) \\
& w_2(i,j) \triangleq w_2(\mathbf{x}_i, \mathbf{x}_j) = 1 - exp(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{\varepsilon_2^2})
\end{aligned}
$$

Here $\varepsilon_1 < \varepsilon_2$, since we use weight w_1 to control the similarity of the data within the same neighbourhoods and w_2 to control the similarity of the data within the different clusters. Obviously, suppose a pair of data $\mathbf{x}_i$ and $\mathbf{x}_j$ are close within the same cluster, w_1 would be approximate to 1 and w_2 would be approximate to 0, meaning that the second term has almost no effect on the cost function and only the first term is penalized. This situation is the same as the formulation of unsupervised spectral embedding. While the pair of data x_i and x_j are not within the same neighbourhood, w_1 would be small but w_2 would be close to 1. Hence, minimizing the cost function is equivalent to map the pair of data far apart. The parameters ε_1 and ε_2 control the similarity measure of the data within the same and different neighbourhoods respectively and should be chosen appropriately based on training data. Also, we introduce the parameter α as a weight to provide a trade-off between the contributions of the two terms in the cost function. We rewrite Eqn. 5.9 as

$$
\sum_{i,j} \left[\left(\alpha w_1(i,j) + \alpha w_2(i,j) - w_2(i,j) \right) \|\mathbf{y}_i - \mathbf{y}_j\|^2 \right]. \tag{5.10}
$$

Denote the combined weight as $\tilde{w}(i,j) = \alpha w_1(i,j) + \alpha w_2(i,j) - w_2(i,j)$. We note that the formulation of Eqn. 5.10 is the same as Eqn. 5.3. Hence, minimizing Eqn. 5.10 is equivalent to find the corresponding eigenvectors of $(\tilde{D} - \tilde{W})$ associated with minimal eigenvalues, where $\tilde{D}(i,i) = \sum_j \tilde{w}(i,j)$ is the diagonal matrix. Since we propose using two different ε parameters obtained from the training data to control the intra-cluster and inter-cluster similarity of data pairs rather than using the complete label information as in supervised learning, we refer our scheme as semi-supervised spectral embedding (SSE).

5.3.3 Out-of-Sample Extension

However, the embedding obtained above is only capable of constructing binary codes for the training images rather than handling new original images as well as the distorted copies. To avoid the re-training process, we need to generate the binary codes for new images based on the learned embedding, which is referred as out-of-sample extension. A classical solution of the out-of-sample extension of spectral methods is the Nystrom method [10]. The essential idea is to take advantages of the existing embedding (eigenvalues and eigenvectors) and interpolate the approximate embedding for a new input.

Suppose that a data set $T = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n\}$ contains n i.i.d samples from an unknown distribution with the density function $p(\mathbf{x})$ and a Hilbert space $\mathcal{H}_p$, we could define the eigenvalue problem of the linear operator $\mathcal{K}_p$ as

$$(\mathcal{K}_p f)(\mathbf{x}) = \int K(\mathbf{x}, \mathbf{y}) f(\mathbf{y}) p(\mathbf{y}) \mathrm{d}\mathbf{y} = \lambda f(\mathbf{x}), \tag{5.11}$$

where $K(\mathbf{x}, \mathbf{y})$ is a symmetric kernel function (not necessarily positive semi-definite), λ and $f(.)$ are the eigenvalue and eigenfunction of the linear operator $\mathcal{K}_p$. Since the density $p(\mathbf{x})$ is unknown, an "empirical" distribution $\hat{p}$ estimated from the training data set T could be used to approximate $p(\mathbf{x})$ so that we have

$$(\mathcal{K}_{\hat{p}} f)(\mathbf{x}) = \frac{1}{n} \sum_{i=1}^{n} K(\mathbf{x}, \mathbf{x}_i) f(\mathbf{x}_i) = \lambda f(\mathbf{x}), \tag{5.12}$$

which means the eigenfunction of the new input $\mathbf{x}$ could be approximated by the weighted linear combination of eigenfunctions of the training data with respect to the kernel function.

Hence, let the symmetric $n \times n$ matrix M be with entries $M(i, j) = K(\mathbf{x}_i, \mathbf{x}_j)$ and $\{\lambda_m, v_m\}$ be the m-th minimum eigenvalue and the associated eigenvector, which satisfy the eigenvalue problem $Mv_m = \lambda_m v_m$. Then for a new data $\mathbf{t}$, the m-th element of the interpolated embedding $u(\mathbf{t})$ can be

calculated as:

$$u_m(t) = \frac{1}{\lambda_m} \sum_{i=1}^{n} v_m(i) K(\mathbf{t}, \mathbf{x}_i). \tag{5.13}$$

for $m = 1, ...n$, where $v_m(i)$ means the i-th element of the vector v_m. The advantage of the Nystrom method is its adaptivity to arbitrary data distributions at the cost of calculating the kernel similarity of the new input with respect to the training data each time.

It is worth mentioning that, by assuming that the data distribution is uniform or Gaussian, the embedding could be directly calculated based on the eigenfunctions rather than interpolations in the original spectral hashing [98]. However, in our binary image hashing problem, since the distributions of intermediate hashes can be different for different image hashing methods, we feel that the Nystrom method is more promising. In practice, we use the following normalized kernel weight for the new input data $\mathbf{t}$ w.r.t the training data $\mathbf{x}_i$ based on Eqn. 5.10 as

$$\hat{K}(\mathbf{t}, \mathbf{x}_i) = \frac{1}{n} \frac{K(\mathbf{t}, \mathbf{x}_i)}{\sqrt{|E_{\mathbf{x} \in T}[K(\mathbf{t}, \mathbf{x})] E_{\mathbf{x} \in T}[K(\mathbf{x}_i, \mathbf{x})]|}}, \tag{5.14}$$

$$K(\mathbf{t}, \mathbf{x}_i) = \alpha w_1(\mathbf{t}, \mathbf{x}_i) + \alpha w_2(\mathbf{t}, \mathbf{x}_i) - w_2(\mathbf{t}, \mathbf{x}_i), \tag{5.15}$$

where $E_{\mathbf{x} \in T}[K(\mathbf{t}, \mathbf{x})]$ is the average kernel similarity of the new input data $\mathbf{t}$ with respect to the training data set T. Hence, the m-th bit of the binary code of the new input data $\mathbf{t}$ is

$$\tilde{y}_m(\mathbf{t}) = sgn(\sum_{i=1}^{n} v_m(i) \hat{K}(\mathbf{t}, \mathbf{x}_i)). \tag{5.16}$$

With the help of out-of-sample extension, the binary codes of a new input data could be obtained based on the existing learned embedding without retraining, and thus it is computationally efficient in practice.

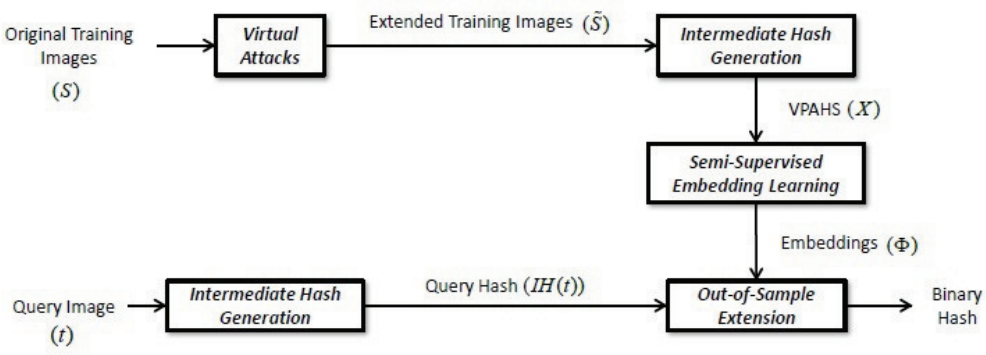


Figure 5.2: The proposed binary image hash generation using the semi-supervised spectral embedding learning

5.3.4 Proposed Method for Binary Image Hash Construction

Given a set of N original training images $S = \{s_i\}_{i=1}^{N}$ and an intermediate hash function $IH(x)$ that generates real-valued intermediate hashes, we summarize the main steps of the proposed binary hash construction approach based on semi-supervised spectral embedding, as shown in Figure 5.2.

- *Step 1*: For each original training image s_i, we apply the virtual prior attacks (referring to the experiment section) to generate an extended set of M distorted copies $\tilde{s}_i = \{s_{ij}\}_{j=1}^{M}$, and we thus have the extended training image data $\tilde{S}$. The corresponding VPAHS is obtained by applying the intermediate hash function to $\tilde{S}$, denoted as $X = \{IH(\tilde{s}_{ij})\}_{i=1,j=1}^{N,M}$.

- *Step 2*: With the well chosen parameters ε_1, ε_2 and α, construct the Laplacian matrix based on Eqn. 5.10 and find the k eigenvectors $\Phi = \{\phi_1(X), \phi_2(X), ..., \phi_k(X)\}$ with the minimal eigenvalues, where Φ is with size $N \times M$-by-k. The k-bit binary code of each training image s_{ij} could be obtained based on the corresponding l-th row of Φ, where

the index $l = (i-1)N + j$:

$$b_{s_{ij}} = \{sgn(\phi_1(l)), sgn(\phi_2(l)), ..., sgn(\phi_k(l))\}. \tag{5.17}$$

- *Step 3*: Given a query image t, we first generate the intermediate hash $IH(t)$ and then apply the out-of-sample extension to obtain the binary codes based on Eqn. 5.16. The computation cost is linear with respect to the size of the training image set, since the approximate eigenvectors of the query feature $IH(t)$ are interpolated by the weighted combination of each similarity kernel between the query image and training images.

The proposed scheme has two-fold advantages: 1) Compared with the conventional binary image hash generation based on quantization, the prior information obtained from VPAHS is incorporated into the binary embedding process. 2) For a new input data, the binary codes could be obtained directly based on the existing learned embedding. Its superior performance will be illustrated in the experiment section.

5.4 Proposed Framework for Combining Multiple Image Hashes

Generally, extracting a universal robust feature against various attacks and distortions may not be feasible. Many researchers seek to combine different robust features to generate a better hash that has superior performances at the cost of extending the length of hash [60, 63], which obviously would decrease the efficiency of the similarity measure and retrieval process. Here we extend the proposed binary image hash generation scheme to tackle the joint embedding problem, with the goal to combine different intermediate real-valued image hashes or robust features together and construct the corresponding binary codes that share their robustness without extending the length of final binary hashes.

Suppose, for n images, we have m sets of feature vectors (representing m

types of real-valued image hashes) as $\psi = \{X^1, X^2, ..., X^m\}$, where each element is an intermediate hash vector and the t-th type of image hash data is denoted as $X^t = \{\mathbf{x}_j^t\}_{j=1}^n$. The intermediate real-valued image hashes arising from different hash functions may be robust against certain attacks or distortions respectively. Obviously, different feature spaces may not be compatible for directly measuring the similarity. Recalling the semi-supervised spectral embedding based on Eqn. 5.9, the learning process is only dependent on the similarity measure of the data pair based on the Gaussian kernel rather than the dimensions of the features. Hence, we reformulate the Eqn. 5.9 as

$$\sum_{i,j} \left[\alpha \cdot \tilde{w}_1(i,j) \left\| \mathbf{y}_i - \mathbf{y}_j \right\|^2 - (1-\alpha) \cdot \tilde{w}_2(i,j) \left\| \mathbf{y}_i - \mathbf{y}_j \right\|^2 \right] \tag{5.18}$$

$$\begin{aligned} where: \qquad \tilde{w}_1(i,j) &= exp(-\sum_{t=1}^{m} \frac{\left\| \mathbf{x}_i^t - \mathbf{x}_j^t \right\|^2}{(\varepsilon_1^t)^2}) \\ \tilde{w}_2(i,j) &= 1 - exp(-\sum_{t=1}^{m} \frac{\left\| \mathbf{x}_i^t - \mathbf{x}_j^t \right\|^2}{(\varepsilon_2^t)^2}) \end{aligned}$$

By controlling parameters $\{\varepsilon_1^t\}_{t=1}^m$ and $\{\varepsilon_2^t\}_{t=1}^m$, the m feature spaces are compatible for measuring the similarity based on Gaussian kernel. Therefore we could combine different types of real-valued intermediate hashes with certain robustness into the semi-supervised spectral embedding framework. Since we still can only choose the k minimal eigenvectors for binary codes construction, the length of the final binary codes can be maintained as k for the joint embedding. In addition, via the above joint embedding, the robustness arising from different types of intermediate hashes can somehow be preserved into the final binary codes, which is superior to the ones based on certain hash features.

5.5 Experimental Results and Analysis

The focus of the chapter is to construct robust binary image hashes based on real-valued intermediate hashes. Hence we mainly evaluate the perceptual robustness of the generated binary hash codes against different content-preserving attacks and distortions. Since the similarity measurement for the proposed binary hashes is based on Hamming distance, the search and retrieval efficiency is superior to that of real-valued intermediate hashes, which is generally measured based on Euclidean distance. It is worth mentioning that, since the embedding is essentially a lossy compression process, the more robustness preserved from the real-valued intermediate image hashes, the better binary hash codes are constructed.

5.5.1 Database and Content-Preserving Manipulations

In order to evaluate the perceptual robustness of the proposed hashing algorithms against content-preserving manipulations, we first construct a database with over 107000 images, which includes 1000 original gray nature images with size 256×342 and 106 distorted copies by manipulating the original image according to 12 classes of content-preserving operations, including additive noise, blurring, JPEG compression, geometric attacks and brightness changes etc. The motivation to design such a database is to simulate possible quality distortions of digital images due to the noise in transmission channel, lossy quantization, and geometric manipulations. The details are given in Table 5.1. For the additive noise and blurring attacks, the distortion is introduced based on an acceptable quality range (e.g. PSNR$\geq 25dB$). All the operations are implemented using Matlab. The original image database and the code used to generate the overall database can be found at (http://ipl.ece.ubc.ca/multimedia.html) for public research use.

5.5.2 Identification and Evaluation Measures

Perceptual robustness is one critical criteria to evaluate the performances of image hashing schemes. Ideally, two distinct images I and I' should have

Table 5.1: Content-preserving manipulations and parameters setting

Manipulation	Parameters Setting	Copies
Additive Noise		
Gaussian Noise	$variance \in (0.0005 \sim 0.005)$	10
Salt&Pepper Noise	$variance \in (0.001 \sim 0.01)$	10
Speckle Noise	$variance \in (0.001 \sim 0.01)$	10
Blurring		
Gaussian Blurring	filter size: 3, $\sigma \in (0.5 \sim 5)$	10
Circular Blurring	radius $\in (0.2 \sim 2)$	10
Motion Blurring	len: $1 \sim 3$, $\theta \in \{0^0, 45^0, 90^0\}$	9
Geometric Attacks		
Rotation	$\theta = 2^0 \sim 30^0$	8
Cropping	boundary: $2\% \sim 10\%$	9
Scaling	factor: $0.5 \sim 1.5$	5
Shearing	$\theta \in (1\% \sim 10\%)$	10
JPEG Compression	Quality Factor $\in (10 \sim 50)$	5
Gamma Correction	$\gamma \in (0.7 \sim 1.3)$	10

different image hashes but a manipulated copy I_M of image I under a certain distortion should have a similar image hash to its original copy. Here we conduct the evaluation for the proposed scheme in two aspects: identification accuracy and receiver operating characteristics (ROC) analysis.

- *Identification accuracy*: The identification accuracy is defined as the fraction of the distorted image copies that are correctly classified to the corresponding original images. Suppose the generated final image hash for each image is a k-bit binary code in our proposed scheme, the Hamming distance could be used as the distance metric to measure the similarity between two binary image hashes h_1 and h_2 as

$$D(h_1, h_2) = \sum_{i=1}^{k} |h_1(i) \oplus h_2(i)| . \tag{5.19}$$

  With the above distance measure, we just adopt the simplest nearest-neighbour classifier to facilitate the retrieval process. If we have K

multiple copies of each original image with no distortions or with only slight distortions, we could adopt the K-nearest-neighbour (KNN) classifier instead.

- *Receiver operating characteristics analysis*: Except investigating identification accuracy, we also study the ROC curve [26] to illustrate the identification performances of the proposed binary image hash generation scheme. The ROC curve depicts the relative tradeoffs between the benefit and cost of the identification process and is an effective way to compare the performances of different image hashing approaches. Let $H(x)$ is an image hashing function that maps the image to a binary signature. To obtain ROC curves to compare hashing algorithms, we define the probability of true identification $P_T(\xi)$ and probability of false alarm $P_F(\xi)$ as

$$P_T(\xi) = Pr(D(H(I), H(I_M)) < \xi) \tag{5.20}$$

$$P_F(\xi) = Pr(D(H(I), H(I'_M)) < \xi) \tag{5.21}$$

where ξ is the identification threshold. Images I and I' are two distinct original images and the images I_M and I'_M are manipulated versions of the images I and I', respectively. Given a certain threshold ξ, a better hashing should provide a higher $P_T(\xi)$ with a lower $P_F(\xi)$. When we obtain all the distances between manipulated images and original images, we could generate ROC curves by sweeping the threshold ξ from the minimum value to the maximum value, and further compare the performances of different hashing approaches.

5.5.3 Intermediate Hashes and Baseline Methods

In this chapter, we assume the availability of real-valued intermediate hashes, and the focus is to evaluate how the robustness could be preserved by the generated binary image hashes. We compare the proposed scheme to the conventional methods that are based on quantization as the baseline meth-

ods.

Intermediate Hashing

The real-valued intermediate hashing methods we adopt in this chapter are the state-of-art image hashing schemes including FJLT hashing (FJLTH)[60] presented in Section 3 and Shape Contexts based image hashing (SCH) [63] presented in Section 4.

- *FJLTH*: It is an image hashing that is based on the Fast Johnson-Lindenstrauss transform (FJLT), which shares the low distortion characteristics of random projection but requires lower computational complexity. It is robust against classical attacks and distortions such as additive noise, compression, blurring etc.

- *SCH*: It is a very recent image hashing that takes advantages of local feature patterns such as SIFT and embeds the geometric distributions of feature points into hashes based on the shape contexts descriptors, which are robust against geometric attacks and brightness changes.

The hashes arising from these two schemes are real-valued intermediate hashes, which are usually measured by Euclidean distance metrics. We apply the proposed semi-supervised spectral embedding scheme to further compress these real-valued intermediate hashes into binary hashes and measure the similarities using Hamming distance to enhance the efficiency. Aside from the evaluation for the robustness of the constructed binary codes for each of the hashing schemes, we also investigate the performance of the joint embedding described in Section 5.4 to combine these two complementary hashing schemes together to further enhance the overall robustness of the proposed binary image hashes.

Baseline Methods

To illustrate the performance of the proposed scheme, we compare it with the conventional quantization-based post-processing method as the baseline method. Let the l-length real-valued hash vector be $H = \{h_1, h_2, ..., h_L\}$.

To generate the k- bit binary image hash, we quantize each of the entries and represent it using k/L bits by gray coding q_g as

$$B_H = \{q_g(h_1), q_g(h_2), ..., q_g(h_L)\}. \tag{5.22}$$

To fulfill the quantization within the representative range of k/L bits, the intermediate hash space may need to be well normalized first. Also, the resulting binary hashes could be further compressed by ECC decoder such as Reed-Muller decoder etc. Although ECC could further enhance the robustness by correcting small distortions in low level bits, it also would introduce extra false alarms in practice, since the correction has effects on arbitrary bits. Hence, we simply adopt quantized binary hashes as the baseline. For the distance measure applied for this baseline method, we first convert binary hashes back to integers and use Euclidean distance for classification due to the fact that each bit has different weights, which are significantly important for preserving the discriminative capabilities. For this baseline method, our preliminary study shows that such Euclidean distanced based approach yields much better identification performances than that of Hamming distance based approach.

5.5.4 Embedding Training

VPAHS Generation

To learn the semi-supervised embedding for binary codes construction, we first generate the VPAHS based on the training data, which include 100 original images and 40 distorted copies for each of them that consist of a subset of 12 classes of content-preserving distortions listed in Table 5.1 with selected parameters. Hence, we totally have 4000 images for the training stage. Then we generate the intermediate hashes using FJLTH [60] and SCH [63] for each of the training image and obtain the corresponding real-valued L-length hash vectors H_{FJLT} and H_{SCH}. We use the default length $L = 20$.

Parameters Setting

One critical issue for the proposed embedding learning scheme is the parameters setting, especially the choices of ε_1 and ε_2, which are designed to measure the intra-class similarity and inter-class similarity respectively. Here we mainly discuss the parameters setting for a single feature space, which could be further extended to handle the case of m feature spaces in joint embedding.

In Section 5.2.2, we have a glance at how distorted hashes distribute with respect to the original hash. Since the original image hash may not be the centroid of its hash cluster in VPAHS, we investigate the relations among the centroids of different clusters in the training data to choose better parameters. Let $C = \{c_i\}_{i=1}^{m}$ be the set of hash centroids estimated for the training VPAHS, which contains m original images and corresponding distorted copies, we set

$$\varepsilon_1 = \frac{\mu_1}{2} \min \|c_i - c_j\|_{i \neq j} \tag{5.23}$$

$$\varepsilon_2 = \frac{\mu_2}{2} \min \|c_i - c_j\|_{i \neq j} \tag{5.24}$$

where $\|\cdot\|$ means the Euclidean distance, and μ_1 and μ_2 are adjustable weight coefficients. Therefore, we simply use the minimal distance among the training centroids as the reference to set the similarity measure parameters ε_1 and ε_2. In the experiment, we use $\mu_1 = 8$ and $\mu_2 = 32$ to satisfy the condition $\varepsilon_1 < \varepsilon_2$. The above choices of the parameters are heuristic but not exclusive.

As for the weight parameter α that weights the contributions of the intra-class similarity and inter-class similarity in Eqn. 5.9, 0.5 can be a natural choice to balance both term in the cost function. But we set $\alpha = 0.6$ heuristically to slightly more emphasize the contribution of the inner-class similarity and it generally provides better overall performances.

Table 5.2: Identification accuracy performances of different hashing algorithms under various attacks (here $k = 100$ for the k-bit binary image hashes).

Manipulations	FJLT (R)	FJLT (Q)	FJLT (SSE)	SCH (R)	SCH (Q)	SCH (SSE)	Joint SSE
Additive Noise							
Gaussian Noise	100%	91.77%	96.26%	91.67%	71.26%	84.37%	94.73%
Salt&Pepper Noise	100%	93.82%	97.52%	92.95%	77.22%	89.32%	96.43%
Speckle Noise	100%	96.45%	98.07%	96.42%	77.8%	92.23%	98.06%
Blurring							
Gaussian Blurring	100%	89.12%	95.19%	88.9%	69.5%	82.6%	95.36%
Circular Blurring	100%	92.38%	96.56%	89.6%	77.27%	86.32%	95.2%
Motion Blurring	100%	95.77%	98.02%	98.19%	85.46%	96%	98.96%
Geometric Attacks							
Rotation	59.99%	34.95%	41.26%	90.75%	69.78%	83.26%	87.49%
Cropping	95.41%	60.22%	67.21%	96.54%	82.96%	94.1%	97.77%
Scaling	100%	95.52%	99.06%	88.18%	67.68%	77.68%	89.08%
Shearing	98.25%	66.6%	76.5%	92.26%	71.99%	86.87%	94.11%
JPEG Compression	100%	95.96%	98.96%	96.18%	74.64%	91.38%	97.28%
Gamma Correction	61.65%	27.49%	24.96%	95.57%	78.4%	91.95%	91.92%

5.5.5 Experimental Results

Performance Evaluation by Identification Accuracy

We first evaluate the performances of the proposed schemes in terms of identification accuracy. In the experiment, we assume the real-valued intermediate image hashes (e.g. FJLT hash and SCH hash) are available for each image in the database, whose lengths are set as $L = 20$ based on [60, 63]. Suppose each hash component is stored using 2 bytes, the total bits are 320 bits. We indicate the identification accuracy results of real-valued intermediate FJLT and SCH hashes based on Euclidean distance measure in Table 5.2 with "R" in the parentheses, which serve as the upper-bound performance references for evaluating the performances of the corresponding binary image hashes. Since the process to construct binary image hashes that represent the corresponding real-valued intermediate hashes is essentially a lossy compression procedure, the performance may degrade when fewer bits are used. When the length of binary codes is selected, the more robustness is preserved, the better binary image hashes are constructed.

The conventional post-processing method is based on quantization, which treats each hash component individually and quantizes them to a certain number of bits. In the experiment, we quantize each component into 5 bits and thus the final binary hashes are $L = 100$ bits long. We denote the

results of this baseline approach with "Q" in the parentheses in Table 5.2. We note that, compared with the performances of real-valued intermediate hashes (consisting of 320 bits), the identification performances of quantized binary hashes degrade for both FJLT and SCH image hashing. Since now each real-valued hash component is only quantized by 5 bits, the discriminative capability of the original image hashes decreases under the lossy compression. For FJLT hashing, since it is robust against additive noise, blurring, and compression etc, the corresponding distorted hashes are distributed densely close to the original hash and the quantized binary hashes still preserve most robustness against these attacks. As for the distortions, to which FJLT hashing is sensitive, such as geometric attacks and gamma correction, the distorted hashes are usually distributed far from the original hash and the corresponding binary hashes also have less robustness preserved. The quantization also affects the performances of SCH hashing. Since the distribution of distorted SCH hashes is not so densely close to the original hash as in FJLT hashing under additive noise and blurring attacks, the quantization further spreads the hash clusters and the final binary hashes have less discriminative capability for identification. Obviously, without considering any prior information (e.g. VPAHS), the conventional quantization method is only feasible when enough bits are used to preserve the inter-image-cluster differences, and less bits can be used if the original real-valued intermediate hashes are robust enough that the distorted hashes distribute densely close to original hashes. Therefore determining the number of bits used to quantize intermediate hashes is a critical issue in the conventional quantization method.

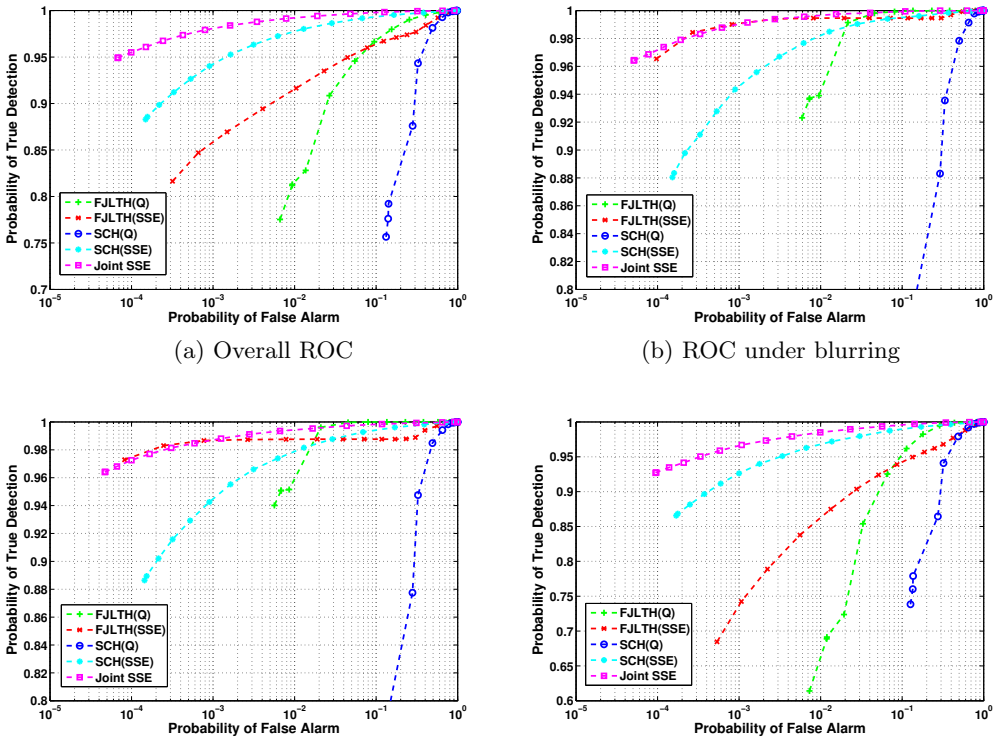
To the contrary, the proposed semi-supervised spectral embedding (SSE) scheme takes the VPAHS as prior information into consideration and seeks the best embedding to construct the binary codes that minimize the particular distance-based cost function. The constructed binary image hashes are optimal for preserving the robustness of the real-valued intermediate hashes given the similarity metrics. In this experiment, we investigate the performances of the proposed SSE when applied on the FJLTH and SCH hashes and report the results in Table 5.2. It is clear that the performances

of SSE for both FJLT and SCH are better than the quantization methods when using the same number of bits. When compared with the real-valued intermediate hashes, the average degradation on identification accuracy of the proposed SSE is around 5%, while it is $10\% \sim 15\%$ for the conventional quantization. Hence, the binary hashes based on SSE could preserve more robustness of the corresponding intermediate hashes than the conventional quantization.

Moreover, another attractive advantage of SSE is its generality for joint embedding of multiple feature spaces. With the same number of bits, SSE could embed multiple types of real-valued intermediate hashes into final robust binary hashes. In the experiment, we combine FJLTH and SCH hashes together based on the proposition of joint SSE presented in Section 5.4 and report the results in Table 5.2. Without extending the hash length, the final binary SSE hash shares the robustness against additive noise, blurring etc. from FJLTH and robustness under geometric and brightness changes from SCH, and thus it achieves globally superior performance in terms of identification accuracy.

Performance Evaluation by ROC curve

We then present a statistical comparison of the proposed semi-supervised spectral embedding scheme with the conventional quantization methods by studying the corresponding ROC curves, as shown in Figure 5.3. At a given probability of false alarm $P_F(\xi)$, a better hashing approach could achieve a higher probability of true identification $P_T(\xi)$. In other words, ROC curves provide a tradeoff between the true retrieval and misclassification by selecting the threshold for identification. From Figure 5.3a, it is noted that the proposed SSE scheme achieves better overall robustness against all attacks and distortions listed in Table 5.1, when compared with the conventional quantization methods for both FJLTH and SCH hashing. Furthermore, the joint SSE by combining the robustness of FJLTH and SCH is demonstrated to be the best approach, which is consistent with the reported results in Table 5.2. We also illustrate the ROC curves under blurring, additive noise,



(a) Overall ROC

(b) ROC under blurring

(c) ROC under additive noise

(d) ROC under geometric attacks

Figure 5.3: The ROC curves of the conventional quantization, the SSE, and the joint SSE using FJLTH and SCH.

and geometric attacks in Figure 5.3b, 5.3c, 5.3d. We note that the proposed SSE scheme is consistently better than the conventional quantization approach and the joint SSE is the best among different approaches by preserving the robustness from both FJLH and SCH.

The Length of Binary Hashes

The hash length is a critical parameter that affects the overall performance of an image hashing algorithm. A shorter length is preferred in image hashing for the high efficient similarity measure and retrieval purposes. In conventional quantization methods where each component of the hash vector is

Table 5.3: The comparison of average identification accuracy of binary image hashes based on the proposed SSE and conventional-quantization methods, when different number of bits are used.

Hash Bits	40	60	80	100	120	140
FJLTH						
Quantization	\	\	\	83.12%	89.32%	90.69%
SSE	83.24%	85.42%	88.61%	88.64%	89.12%	89.9%
SCH						
Quantization	\	\	\	84.15%	86.14%	94.13%
SSE	93.89%	94.99%	95.58%	96%	96.31%	96.48%

quantized individually, the fewer bits are used, the less robustness is preserved by the final generated binary hashes, and thus could produce high false alarm rates in classification. Since the proposed SSE scheme considers all the hash components together and seeks the best embedding to construct the binary codes, the generated final binary hash is less sensitive to the number of bits used. In the experiment, we test the proposed SSE and conventional quantization method for FJLTH and SCH in a small image database, which includes 100 original images and 106 distorted copies for each of them, as function of the numbers of bits used. The comparison of average identification accuracy is reported in Table 5.3.

It is noted that, when the number of hash bits is below 80, the identification procedure produces high false alarm rates and the identification accuracies are very poor when using the quantization methods for both FJLTH and SCH. Hence, we use "\" to indicate that the result is not acceptable and the correspond setting is not feasible. For the quantization methods for both FJLTH and SCH, when more hash bits are used, the average identification accuracy is better, since the binary hashes have higher discriminative capability which is closer to that of the real-valued intermediate hashes. However, we note that the proposed SSE scheme is not very sensitive to the number of hash bits. Even when the number of hash bits is as low as 40, the performance of the proposed SSE is still relatively good. It is clear that the proposed SSE scheme is superior to the conventional

quantization method for short binary hash generation. We also note that the performance of SSE does not increase as sharp as in the conventional quantization method when more hash bits are used. This is because that the eigenfunctions arising from the Laplacian matrix are not independent [38, 98] and the later eigenfunctions contribute less to minimize the cost function. In summary, the proposed SSE scheme is more significant when short binary image hashes are required.

5.6 Conclusion

In this chapter, we mainly propose a semi-supervised spectral embedding (SSE) method for compressing real-valued intermediate image hashes into short robust binary image hashes. Instead of quantizing intermediate image hashes individually, the proposed method takes advantages of VPAHS as the prior information and seeks the optimal embedding for constructing binary image hashes, which jointly explores the intra-image and inter-image similarities. Our experimental results demonstrate that the proposed SSE method could generate short binary hashes with better robustness when compared with the ones arising from conventional quantization methods. Furthermore, a significant contribution of the proposed SSE is its generality to multiple image hash spaces, and we propose the joint SSE that could efficiently combine multiple types of intermediate hashes together and embed them jointly into fixed-length binary hashes. The constructed binary image hashes based on the joint SSE share the robustness of the combined individual intermediate hashes and are shown to provide better overall identification performance.

However, one possible bottleneck of the proposed SSE scheme lies in its efficiency of out-of-sample extension. Since we adopt the classical Nystrom method, which is based on the weighted interpolation, the computational cost is linear with respect to the size of the training dataset. Therefore, it may not be efficient when large training data are used. An alternative way to deal with this issue is to investigate the data distribution function and obtain the exact forms of Laplacian eigenfunctions directly for an arbitrary

input instead of using weighted interpolations. However, the eigenfunction problems of Laplacian under different data distributions (except for the uniform and Gaussian ones) are still an ongoing research topic [76]. Further, the pseudorandomization technique could be incorporated into the proposed SSE scheme in the future work to further enhance the security of the final binary image hashes for preventing unauthorized access and distribution.

Chapter 6

Conclusions and Future Works

6.1 Conclusions

In this thesis, perceptual image hashing and content-based fingerprinting concepts are investigated, from the theory to applications. Several novel techniques for improving the perceptual robustness of image hashing are proposed, analyzed and evaluated on large scale image database. The conclusions of the thesis are summarized as follows:

Chapter 2 presents a literature survey of image hashing and content-based fingerprinting. For the related references proposed in the literature for image copy detection, image authentication and tampering detection etc., we categorize and discuss the algorithms according to the major components including pre-processing on images, robust feature extraction, feature compression and post-processing. For the security concern, security of image hashing arising from pseudo-randomization is also analyzed. Different from traditional review papers, all the related references are listed in lookup tables for convenient access. Furthermore, from the tables, researchers could easily obtain the perspectives of the recent developments in the research area of image hashing and content-based fingerprinting. For instance, Table 2.2 illustrates the robust features that are usually adopted in previous image hashing works. The more certain features are studied in related works, generally the better chance that the features can be employed to generate robust image hashes against a large class of distortions and attacks. Another example is in Table 2.5 and Table 2.6, where the distortions

and attacks evaluated in related references are listed. Most works investigate the robustness against Gaussian noise, Gaussian filter, compression and geometric attacks. It suggests that these types of distortions and attacks are the most common ones, to which the proposed scheme should be robust. Based on the robustness against these preliminary distortions, researchers should extend their algorithms to deal with other attacks. Hence, this comprehensive review reveals the prospective research directions in the area of digital image hashing and content-based fingerprinting.

Chapter 3 presents a digital image hashing algorithm based on a recent dimension reduction technique, the Fast Johnson-Lindenstrauss Transform (FJLT). FJLT is essentially a random projection method that could preserve the local similarity of data in a high dimensional space into a lower dimensional space. Images are divided into overlapped sub-images by random sampling and treated as high-dimension features, which could be further projected into a lower dimensional space by FJLT for generating compact image hashes. From the robustness of the proposed FJLT hashing demonstrated in experiments, we can see that image pixel values and statistics are robust features against classic image processing attacks such as additive noise, blurring, and compression etc., but sensitive to geometric attacks and brightness changes. Hence, the popular Fourier-Mellin transform is incorporated into the proposed FJLT hashing (FJLTH) to improve its performances under rotation attacks and the content-based fingerprinting concept by combining FJLTH and rotation-invariant FJLTH (RI-FJLTH) is further presented and demonstrated to yield superior robustness against various distortions and attacks. Obviously, content-based fingerprinting is an extended concept from image hashing and essentially a feature fusion scheme. The underlying motivation for introducing the content-based fingerprinting concept is that generating a single type of image hash based on certain kinds of features to resist all types of manipulations is highly unlikely, while it is relatively easier to find a specific feature to be robust against certain distortions. Fusion at the final hash level could improve the overall robustness. However, in content-based fingerprinting, the tradeoff between robustness and compactness should be taken into consideration carefully.

Chapter 4 presents a novel image hashing algorithm based on robust SIFT-Harris feature point detection and shape context descriptors. Local feature patterns such as SIFT have been well studied in the computer vision area and applied to many research topics and areas. The major benefit of employing local feature patterns is the robustness against geometric attacks. However, its sensitivity to classic image processing attacks such as noise addition, compression and especially blurring restricts its practical applications in image hashing. Based on our preliminary study, Harris criterion is incorporated to select the most stable SIFT key points under various distortions. Radial shape context hashing (RSCH) and angular shape contest hashing (ASCH) schemes are proposed by embedding the SIFT-Harris feature points into shape context descriptors in radial and angular directions respectively. The proposed SCH schemes are demonstrated to be more robust than FJLTH, RI-FJLTH, and NMF hashing under rotation attacks and illumination changes. The combination of RSCH and ASCH is proposed to capture the distribution of local feature points in both radial and angular directions to further explore the feature characteristics and improve the robustness against noise addition, blurring, and compression. Also the proposed SCH schemes could be used for image tampering detection. We note that the shape contexts based image hashing inherits the robustness of local feature patterns against geometric attacks, but the performances under classical image processing attacks can be sacrificed, even for the combined RSCH and ASCH approach, due to the intrinsic weakness of local feature patterns.

Chapter 5 presents a novel binary image hashing compression algorithm using semi-supervised spectral embedding (SSE). The significance of the proposed SSE lies in two aspects: First, it is the first time to incorporate machine learning methods into image hashing. Due to the infinite cluster problem of image hashing, the advanced learning methods suffer from the expensive computational cost requirement for dealing with large-scale image databases for generating digital image hashes. The motivation of the proposed SSE is to learn embeddings to map the real-valued feature space into the binary space, while preserving the local similarity. By applying

out-of-sample extension, the learned embedding could be used for mapping new features into binary signatures without requiring re-training. Therefore it paves the way of applying advanced learning methods to image hashing. Secondly, since the proposed SSE is inherently a kernel based method, the learned embedding mainly relies on the feature similarity rather than feature dimensions. Therefore the proposed SSE scheme could be easily extended to combine multiple real-valued intermediate image hashes and embed them into fixed-length binary hashes. Recalling the tradeoff between robustness and compactness of content-based fingerprinting, when combining different types of image hashes, we can see that the proposed SSE leads to a more sophisticate fusion way to generate robust and compact binary content-based image fingerprints.

6.2 Future Works

6.2.1 Learning Optimal Fusion on Hash Spaces Based on Semi-Supervised Spectral Embedding

Content-based fingerprinting by combining different robust image hashes has been shown to be a more advanced way to improve the perceptual robustness of image hashing against a large class of image processing attacks and distortions. Therefore the optimal fusion of different robust image hashes can be a promising future direction to generate more robust content-based image hashes, instead of seeking universal robust features that are highly infeasible in practice. However, heuristic ways by simply concatenating robust image hashes might benefit the overall robustness at the cost of the compactness. Hence, advanced machine learning schemes should be considered for learning the optimal fusion on the hash spaces but still preserving the compactness of the generated final image hashes.

The proposed SSE scheme in Chapter 5 provides a learning scheme to optimally map the real-valued intermediate hashes into binary signatures, given the cost function that preserves the local similarity. It also facilities heuristic ways to combine different robust hashes to generate binary

signatures without losing the compactness based on the chosen inter-class similarity and intra-class similarity. However, the heuristic way to control inter-class and intra-class similarity based on VPAHS is not optimal. We suggest that the procedure could be further optimized by introducing cost functions that minimize the hash distances between the original images and their distorted copies.

6.2.2 Measurable Robustness and Sensitivity Toward Image Quality Changes

Conventionally, the perceptual robustness of image hashing is measured by the hash distances between original images and their distorted copies based on certain thresholds. There still lacks of a measure to quantify the robustness of an image hashing scheme and monitor its sensitivity to image quality changes. One of our preliminary studies [61] applied FJLT hashing on the DCT domain images and obtained image hashes as side information, which could be used for estimating the image quality of the received image. The hash distances are shown to be related with certain objective measurements of image quality such as PSNR, and it reveals how sensitive the image hash is to image quality changes. It is clear that an image hashing scheme with less sensitivity to image quality changes is more robust than the one that changes drastically for large quality degradation. In other words, by investigating the sensitivity of an image hashing scheme under the image quality changes, it is promising to quantify its robustness and evaluate the robustness in an objective way.

6.2.3 Hashing at Semantic Feature Levels

By going through the image hashing algorithms in literatures, we feel that the most important issue of image hashing is robust feature extraction. However, the features adopted for image hashing in the current literature are still mainly limited to a low feature level, such as image pixel statistics, without any semantic meaning. Currently, the state-of-art content-based image retrieval (CBIR) in computer vision has more focused on retrieving images

at the semantic feature level, such as objects, scenes and so on, by taking advantages of some advanced semantic feature descriptions such as bag-of-words model. Therefore it might be promising to extend image hashing and content-based fingerprinting ideas to be based on the semantic feature level and to be applied to wider applications such as preventing images with certain people, faces, and objects from unauthorized access.

6.2.4 Universal Security Measurements

Aside from the compactness, security is another key issue that makes image hashing and content-based fingerprinting stand out compared with the conventional CBIR area. The pseudo-randomization techniques controlled by a "secret key" could be generally incorporated into any step of the image hashing framework for enhancing the security. However, there still lacks of universal measurements for quantifying and evaluating the security of image hashing. Unpredictability measured by differential entropy or mutual information only reveals necessary properties of secure image hashes, and it is still inadequate. We feel that other analysis such as key space and fragility to sophisticated attacks are also important issues that should be taken into consideration in security study.

Bibliography

[1] Casia, http://forensics.idealtest.org.

[2] Object and concept recognition for content-based image retrieval, http://www.cs.washington.edu/research/imagedatabase/.

[3] M. Abdel-Mottaleb, G. Vaithilingam, and S. Krishnamachari. Signature-based image identification. In *Proc. of SPIE on Multimedia Systems and Applications II*, volume 3845, pages 22–28, 1999.

[4] F. Ahmed, MY Siyal, and V. Uddin Abbas. A secure and robust hash-based scheme for image authentication. *Signal Processing*, 90(5):1456–1470, 2010.

[5] N. Ailon and B. Chazelle. Approximate nearest neighbors and the fast johnson-lindenstrauss transform. In *Proceedings of the 38 annual ACM symposium on Theory of computing*, pages 557–563, 2006.

[6] M. Alghoniemy and A.H. Tewfik. Geometric invariance in image watermarking. *IEEE Transactions on Image Processing*, 13(2):145–153, 2004.

[7] Y. Avrithis, G. Tolias, and Y. Kalantidis. Feature map hashing: sub-linear indexing of appearance and global geometry. In *Proceedings of the ACM International Conference on Multimedia*, pages 231–240, 2010.

[8] H. Bay, T. Tuytelaars, and L. Van Gool. Surf: Speeded up robust features. *Computer Vision and Image Understanding*, 110(3):346–359, 2008.

[9] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 509–522, 2002.

[10] Y. Bengio, J.F. Paiement, P. Vincent, O. Delalleau, N. Le Roux, and M. Ouimet. Out-of-sample extensions for lle, isomap, mds, eigenmaps, and spectral clustering. *Advances in neural information processing systems*, 16:177–184, 2004.

[11] S. Bhattacharjee and M. Kutter. Compression tolerant image authentication. In *IEEE International Conference on Image Processing (ICIP)*, volume 1, pages 435–439, 1998.

[12] C.M. Bishop et al. *Pattern recognition and machine learning*. springer New York, 2006.

[13] A.W. Bowman and A. Azzalini. *Applied smoothing techniques for data analysis*. Oxford University Press, USA, 1997.

[14] P. Brasnett and M. Bober. Fast and robust image identification. In *IEEE 19th International Conference on Pattern Recognition (ICPR)*, pages 1–5, 2008.

[15] Y. Cao, C. Wang, Z. Li, L. Zhang, and L. Zhang. Spatial-bag-of-features. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3352–3359, 2010.

[16] E. Chang, J. Wang, C. Li, and G. Wiederhold. Rime: A replicated image detector for the world-wide web. In *Proc. of SPIE Symposium of Voice, Video, and Data Communications*, volume 3527, pages 58–67, 1998.

[17] M.S. Charikar. Similarity estimation techniques from rounding algorithms. In *Proceedings of the 34 annual ACM symposium on Theory of computing*, pages 380–388, 2002.

[18] H. Chi Wong, M. Bern, and D. Goldberg. An image signature for any kind of image. In *IEEE International Conference on Image Processing (ICIP))*, volume 1, pages I–409, 2002.

[19] S.C. Chu, H.C. Huang, Y. Shi, S.Y. Wu, and C.S. Shieh. Genetic watermarking for zerotree-based applications. *Circuits, Systems, and Signal Processing*, 27(2):171–182, 2008.

[20] O. Chum, J. Philbin, and A. Zisserman. Near duplicate image detection: min-hash and tf-idf weighting. In *Proceedings of the British Machine Vision Conference*, volume 3, pages 493–502, 2008.

[21] T.M. Cover and J.A. Thomas. *Elements of information theory*. Wiley-interscience, 2006.

[22] S. Dasgupta and A. Gupta. An elementary proof of the johnson-lindenstrauss lemma. Technical report, Citeseer, 1999.

[23] M. Datar, N. Immorlica, P. Indyk, and V.S. Mirrokni. Locality-sensitive hashing scheme based on p-stable distributions. In *Proceedings of the 22 ACM annual symposium on Computational geometry*, pages 253–262, 2004.

[24] J. Dittmann, A. Steinmetz, and R. Steinmetz. Content-based digital signature for motion pictures authentication and content-fragile watermarking. In *IEEE International Conference on Multimedia Computing and Systems*, volume 2, pages 209–213, 1999.

[25] M. Fatourechi, X. Lv, Z.J. Wang, and R.K. Ward. Towards automated image hashing based on the fast johnson-lindenstrauss transform (fjlt). In *IEEE International Workshop on Information Forensics and Security (WIFS)*, pages 121–125, 2009.

[26] T. Fawcett. An introduction to roc analysis. *Pattern Recognition Letters*, 27(8):861–874, 2006.

[27] J. Fridrich. Visual hash for oblivious watermarking. In *Proc. SPIE on Security and Watermarking of Multimedia Contents II*, volume 3971, pages 286–294, 2000.

[28] M.A. Gavrielides, E. Sikudova, and I. Pitas. Color-based descriptors for image fingerprinting. *IEEE Transactions on Multimedia*, 8(4):740–748, 2006.

[29] L. Gerold and U. Andreas. Key-dependent jpeg2000-based robust hashing for secure image authentication. *EURASIP Journal on Information Security*, 2008, 2008.

[30] D. Guillamet, B. Schiele, and J. Vitria. Analyzing non-negative matrix factorization for image classification. In *Proceedings of IEEE 16th International Conference on Pattern Recognition (ICPR)*, volume 2, pages 116–119, 2002.

[31] X. Guo and D. Hatzinakos. Content based image hashing via wavelet and radon transform. *Advances in Multimedia Information Processing*, pages 755–764, 2007.

[32] O. Harmanci, V. Monga, and MK Mihcak. Geometrically invariant image watermarking via robust perceptual hashes. In *IEEE International Conference on Image Processing (ICIP)*, pages 1397–1400, 2006.

[33] C. Harris and M. Stephens. A combined corner and edge detector. In *Alvey vision conference*, volume 15, pages 147–151. Manchester, UK, 1988.

[34] J.H. Hsiao, C.S. Chen, L.F. Chien, and M.S. Chen. A new approach to image copy detection based on extended feature sets. *IEEE Transactions on Image Processing*, 16(8):2069–2079, 2007.

[35] A. Jain, K. Nandakumar, and A. Ross. Score normalization in multimodal biometric systems. *Pattern recognition*, 38(12):2270–2285, 2005.

[36] H. Jégou, M. Douze, and C. Schmid. Product quantization for nearest neighbor search. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(1):117–128, 2011.

[37] M. Johnson and K. Ramchandran. Dither-based secure image hashing using distributed coding. In *IEEE International Conference on Image Processing (ICIP)*, volume 2, pages II–751. IEEE, 2003.

[38] A. Joly and O. Buisson. Random maximum margin hashing. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 873–880, 2011.

[39] C. Kailasanathan, R.S. Naini, et al. Image authentication surviving acceptable modifications using statistical measures and k-mean segmentation. *IEEE-EURASIP Work. Nonlinear Sig. and Image Processing*, 1, 2001.

[40] C. Kailasanathan, R.S. Naini, and P. Ogunbona. Compression tolerant dct based image hash. In *Proceedings of the 23rd International Conference on Distributed Computing Systems*, pages 562–567, 2003.

[41] Y. Ke, R. Sukthankar, and L. Huston. An efficient parts-based near-duplicate and sub-image retrieval system. In *Proceedings of the 12th annual ACM international conference on Multimedia*, pages 869–876, 2004.

[42] F. Khelifi and J. Jiang. Perceptual image hashing based on virtual watermark detection. *IEEE Transactions on Image Processing*, 19(4):981–994, 2010.

[43] C. Kim. Content-based image copy detection. *Signal Processing: Image Communication*, 18(3):169–184, 2003.

[44] V. Kitanovski, D. Taskovski, and S. Bogdanova. Combined hashing/watermarking method for image authentication. *International Journal of Signal Processing*, 3(3):223–229, 2007.

[45] S.S. Kozat, R. Venkatesan, and M.K. Mihçak. Robust perceptual image hashing via matrix invariants. In *IEEE International Conference on Image Processing (ICIP)*, volume 5, pages 3443–3446, 2004.

[46] B. Kulis and K. Grauman. Kernelized locality-sensitive hashing for scalable image search. In *IEEE 12th International Conference on Computer Vision (ICCV)*, pages 2130–2137, 2009.

[47] F. Lefebvre, J. Czyz, and B. Macq. A robust soft hash algorithm for digital image signature. In *IEEE International Conference on Image Processing (ICIP)*, volume 2, pages 495–498, 2003.

[48] F. Lefebvre, B. Macq, and J.D. Legat. Rash: Radon soft hash algorithm. In *European Signal Processing Conference*, pages 299–302, 2002.

[49] C.Y. Lin and S.F. Chang. A robust image authentication method distinguishing jpeg compression from malicious manipulation. *IEEE Transactions on Circuits and Systems for Video Technology*, 11(2):153–168, 2001.

[50] C.Y. Lin and S.F. Chang. A robust image authentication method distinguishing jpeg compression from malicious manipulation. *IEEE Transactions on Circuits and Systems for Video Technology*, 11(2):153–168, 2002.

[51] C.Y. Lin, M. Wu, J.A. Bloom, I.J. Cox, M.L. Miller, and Y.M. Lui. Rotation, scale, and translation resilient watermarking for images. *IEEE Transactions on Image Processing*, 10(5):767–782, 2001.

[52] Y.C. Lin, D. Varodayan, and B. Girod. Image authentication based on distributed source coding. In *IEEE International Conference on Image Processing (ICIP)*, volume 3, pages 5–8, 2007.

[53] Y.C. Lin, D. Varodayan, and B. Girod. Spatial models for localization of image tampering using distributed source codes. In *Picture Coding Symposium, Lisbon, Portugal*, 2007.

[54] D.C. Lou and J.L. Liu. Fault resilient and compression tolerant digital signature for image authentication. *IEEE Transactions on Consumer Electronics*, 46(1):31–39, 2000.

[55] D.G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[56] C.S. Lu and C.Y. Hsu. Geometric distortion-resilient image hashing scheme and its applications on copy detection and authentication. *Multimedia Systems*, 11(2):159–173, 2005.

[57] C.S. Lu and H.Y.M. Liao. Structural digital signature for image authentication: an incidental distortion resistant scheme. *IEEE Transactions on Multimedia*, 5(2):161–173, 2003.

[58] W. Lu and M. Wu. Multimedia forensic hash based on visual words. In *IEEE International Conference on Image Processing (ICIP)*, pages 989–992, 2010.

[59] Xudong Lv and Z. Jane Wang. Fast johnson-lindenstrauss transform for robust and secure image hashing. In *IEEE 10th Workshop on Multimedia Signal Processing (MMSP)*, pages 725–729, 2008.

[60] Xudong Lv and Z. Jane Wang. An extended image hashing concept: content-based fingerprinting using fjlt. *EURASIP Journal on Information Security*, 2009:1–17, 2009.

[61] Xudong Lv and Z. Jane Wang. Reduced-reference image quality assessment based on perceptual image hashing. In *IEEE International Conference on Image Processing (ICIP)*, pages 4361–4364, 2009.

[62] Xudong Lv and Z. Jane Wang. Shape contexts based image hashing using local feature points. In *IEEE International Conference on Image Processing (ICIP)*, pages 2541–2544, 2011.

[63] Xudong Lv and Z. Jane Wang. Perceptual image hashing based on shape contexts and local feature points. *IEEE Transactions on Information Forensics and Security*, 7(3):1081–1093, June 2012.

[64] Y. Mao and M. Wu. Unicity distance of robust image hashing. *IEEE Transactions on Information Forensics and Security*, 2(3):462–467, 2007.

[65] Y. Maret, GN Garcia, and T. Ebrahimi. Identification of image variations based on equivalence classes. In *Proc. SPIE on Visual Communications and Image Processing*, volume 5960, pages 584–595, 2005.

[66] A. Meixner and A. Uhl. Security enhancement of visual hashes through key dependent wavelet transformations. In *Proceedings of the 13th international conference on Image Analysis and Processing (ICIAP)*, pages 543–550. Springer, 2005.

[67] A. Meixner and A. Uhl. Robustness and security of a wavelet-based cbir hashing algorithm. In *Proceedings of the 8th ACM Workshop on Multimedia and Security*, pages 140–145, 2006.

[68] A.J. Menezes, P.C. Van Oorschot, and S.A. Vanstone. *Handbook of applied cryptography*. CRC, 1997.

[69] Y. Meng and E.Y. Chang. Image copy detection using dynamic partial function. In *Proc. SPIE on Storage and Retrieval for Media Databases*, volume 5021, pages 176–186, 2003.

[70] K. Mihcak and R. Venkatesan. New iterative geometric techniques for robust image hashing. In *Proc. ACM Workshop on Security and Privacy in Digital Rights Management Workshop*, pages 13–21, 2001.

[71] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *IEEE Transactions on pattern analysis and machine intelligence*, pages 1615–1630, 2005.

[72] V. Monga, A. Banerjee, and B.L. Evans. A clustering based approach to perceptual image hashing. *IEEE Transactions on Information Forensics and Security*, 1(1):68–79, 2006.

[73] V. Monga and B.L. Evans. Perceptual image hashing via feature points: performance evaluation and tradeoffs. *IEEE Transactions on Image Processing*, 15(11):3452–3465, 2006.

[74] V. Monga and MK Mhcak. Robust and secure image hashing via non-negative matrix factorizations. *IEEE Transactions on Information Forensics and Security*, 2(3):376–390, 2007.

[75] V. Monga, D. Vats, and B.L. Evans. Image authentication under geometric attacks via structure matching. In *IEEE International Conference on Multimedia and Expo (ICME)*, pages 229–232, 2005.

[76] B. Nadler, S. Lafon, R.R. Coifman, and I.G. Kevrekidis. Diffusion maps, spectral clustering and reaction coordinates of dynamical systems. *Applied and Computational Harmonic Analysis*, 21(1):113–127, 2006.

[77] R. Norcen and A. Uhl. Robust visual hashing using jpeg 2000. In *Communications and Multimedia Security*, pages 223–235. Springer, 2005.

[78] M.P. Queluz. Content-based integrity protection of digital images. In *Proc. SPIE on Security and Watermarking of Multimedia Contents*, volume 3657, pages 85–93, 1999.

[79] M. Raginsky and S. Lazebnik. Locality-sensitive binary codes from shift-invariant kernels. *The Neural Information Processing Systems*, 22, 2009.

[80] S. Roy and Q. Sun. Robust hash for detecting and localizing image tampering. In *IEEE International Conference on Image Processing (ICIP)*, volume 6, pages 117–120, 2007.

[81] S. Roy, X. Zhu, J. Yuan, and EC Chang. On preserving robustness false alarm tradeoff in media hashing. *Proc. SPIE Visual Communication and Image Processing*, 2007.

[82] R. Salakhutdinov and G. Hinton. Semantic hashing. *International Journal of Approximate Reasoning*, 50(7):969–978, 2009.

[83] M. Schneider and S.F. Chang. A robust content based digital signature for image authentication. In *IEEE International Conference on Image Processing (ICIP)*, volume 3, pages 227–230, 1996.

[84] J.S. Seo, J. Haitsma, T. Kalker, and C.D. Yoo. A robust image fingerprinting system using the radon transform. *Signal Processing: Image Communication*, 19(4):325–339, 2004.

[85] D. Seung and L. Lee. Algorithms for non-negative matrix factorization. *Advances in Neural Information Processing Systems*, 13:556–562, 2001.

[86] C.S. Shieh, H.C. Huang, F.H. Wang, and J.S. Pan. Genetic watermarking based on transform-domain techniques. *Pattern Recognition*, 37(3):555–565, 2004.

[87] F.Y. Shih and Y.T. Wu. Enhancement of image watermark retrieval based on genetic algorithms. *Journal of Visual Communication and Image Representation*, 16(2):115–133, 2005.

[88] C. Skrepth and A. Uhl. Robust hash functions for visual data: An experimental comparison. *Pattern Recognition and Image Analysis*, pages 986–993, 2003.

[89] A.W.M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain. Content-based image retrieval at the end of the early years. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(12):1349–1380, 2000.

[90] S.H. Srinivasan and N. Sawant. Finding near-duplicate images on the web using fingerprints. In *Proceedings of the 16th ACM International Conference on Multimedia*, pages 881–884, 2008.

[91] C. Strecha, A. Bronstein, M. Bronstein, and P. Fua. Ldahash: Improved matching with smaller descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(1):66–78, 2010.

[92] A. Swaminathan, Y. Mao, and M. Wu. Robust and secure image hashing. *IEEE Transactions on Information Forensics and Security*, 1(2):215–230, 2006.

[93] M. Tagliasacchi, G. Valenzise, and S. Tubaro. Hash-based identification of sparse image tampering. *IEEE Transactions on Image Processing*, 18(11):2491–2504, 2009.

[94] Z. Tang, S. Wang, X. Zhang, W. Wei, and S. Su. Robust image hashing for tamper detection using non-negative matrix factorization. *Journal of Ubiquitous Convergence Technology*, 2(1):18–26, 2008.

[95] T. Tuytelaars and K. Mikolajczyk. Local invariant feature detectors: A survey. *Foundations and Trends in Computer Graphics and Vision*, 3(3):177–280, 2008.

[96] R. Venkatesan, SM Koon, M.H. Jakubowski, and P. Moulin. Robust image hashing. In *IEEE International Conference on Image Processing (ICIP)*, volume 3, pages 664–666, 2000.

[97] J. Wang, S. Kumar, and S.F. Chang. Sequential projection learning for hashing with compact codes. In *Proceedings of International Conference on Machine Learning*, 2010.

[98] Y. Weiss, A. Torralba, and R. Fergus. Spectral hashing. In *Advances in Nerual Information Processing Systems*, 2008.

[99] L. Weng and B. Preneel. Attacking some perceptual image hash algorithms. In *IEEE International Conference on Multimedia and Expo (ICME)*, pages 879–882, 2007.

[100] D. Wu, X. Zhou, and X. Niu. A novel image hash algorithm resistant to print–scan. *Signal processing*, 89(12):2415–2424, 2009.

[101] M. Wu, Y. Mao, and A. Swaminathan. A signal processing and randomization perspective of robust and secure image hashing. In *IEEE 14th Workshop on Statistical Signal Processing*, pages 166–170, 2007.

[102] M.N. Wu, C.C. Lin, and C.C. Chang. Novel image copy detection with rotating tolerance. *Journal of Systems and Software*, 80(7):1057–1069, 2007.

[103] S. Xiang, H.J. Kim, and J. Huang. Histogram-based image hashing scheme robust against geometric deformations. In *Proceedings of the 9th ACM workshop on Multimedia & security*, pages 121–128, 2007.

[104] L. Xie, G.R. Arce, and R.F. Graveman. Approximate image message authentication codes. *IEEE Transactions on Multimedia*, 3(2):242–252, 2001.

[105] Z. Xu, H. Ling, F. Zou, Z. Lu, and P. Li. Robust image copy detection using multi-resolution histogram. In *Proceedings of the ACM International Conference on Multimedia information retrieval*, pages 129–136, 2010.

[106] Z. Xu, H. Ling, F. Zou, Z. Lu, and P. Li. A novel image copy detection scheme based on the local multi-resolution histogram descriptor. *Multimedia Tools and Applications*, 52(2):445–463, 2011.

[107] K. Yan and R. Sukthankar. Pca-sift: A more distinctive representation for local image descriptors. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 506–513, 2004.

[108] S.H. Yang and C.F. Chen. Robust image hashing based on spiht. In *IEEE 3rd International Conference on Information Technology: Research and Education (ITRE)*, pages 110–114, 2005.

[109] L. Yu, M. Schmucker, C. Busch, and S. Sun. Cumulant-based image fingerprints. In *Proc. SPIE on Security, Steganography, and Watermarking of Multimedia Contents VII*, volume 5681, pages 68–75, 2005.

[110] F. Zou, H. Ling, X. Li, Z. Xu, and P. Li. Robust image copy detection using local invariant feature. In *IEEE International Conference on Multimedia Information Networking and Security (MINES)*, volume 1, pages 57–61, 2009.