Registration of copyright: D/2004/9964/19

ISBN : 2-930344-58-X
Cover : Luc Letellier
Printed in Belgium

Distribution: www.i6doc.com, on-line university publishers
Available on order from bookshops or at
CIACO University Distributors
Grand-Place, 7
1348 Louvain-la-Neuve, Belgium
Tel. 32 10 47 33 78
Fax 32 10 45 73 50
duc@ciaco.com

Université catholique de Louvain
Faculté des Sciences Appliquées
LABORATOIRE DE TÉLÉCOMMUNICATIONS
ET
TÉLÉDÉTECTION

B - 1348 Louvain-la-Neuve Belgique

Message digests for photographic images and video contents

Frédéric Lefèbvre

Thèse présentée en vue de l'obtention du grade de
Docteur en Sciences Appliquées

Examination Committee:

Benoit MACQ (UCL/TELE) - *Supervisor*
Jean-Didier LEGAT (UCL/DICE) - *Supervisor*
Jean-Jacques QUISQUATER (UCL/DICE-Crypto Group)
Caroline FONTAINE (CR CNRS, France)
Jean-Luc DUGELAY (EURECOM, France)
Luc VANDENDORPE (UCL/TELE) - *President*

Juin 2004

Remerciements

La vie est un long fleuve tranquille semé d'embûche qui réserve de nombreuses surprises.
La vie est un bac à sable, on creuse, on construit, on démolit, on découvre.
La thèse, c'est un peu l'école de la vie, pleine de recherche, de remise en cause, d'égoïsme et de bonheur. On croit tenir le bon bout et tout d'un coup le château de sable s'effondre. Tout est à reconstruire.
Dans toute thèse, il faut un guide, un guide capable de vous indiquer comment construire un bon château. Je remercie le Professeur Benoït Macq d'avoir été ce guide pendant quatre ans en m'accordant beaucoup de confiance et de liberté.
Dans tout choix professionnel, il faut un guide prêt à vous indiquer les bons choix à prendre. Je remercie David Samyde, cet ami qui m'a encouragé à entreprendre une thèse à l'UCL dans l'équipe de Benoit Macq.
Dans la vie, il faut des gardiens du temple, des personnes prêtes à vous soutenir, à recevoir, à donner, à vous remettre dans le droit chemin. Je remercie ma femme Isabelle, mes parents, Francoise et Philippe, ma soeur Caroline, mes amis Marie Thérèse, David et les autres, mes amis et collègues de Louvain la Neuve de m'avoir supporté et encouragé.
Un grand merci à mes amis Gianfranco et Jacek pour leur aide précieuse en latex.

Contents

Introduction

Intellectual property has a growing influence in common usage and it divides users and productive art. Can I extract and copy a picture from an illustrated magazine? Can I (store and) copy a DVDRom movie in my hard disk or a volatil storage device? Due to video compression improvements and high network bandwidth, most of end users say yes. The digital right management and the digital right protection are not under control. It is easier to copy an artwork than to create it. Regarding internet specialist traders, the internet death is expected if communication and storage devices connected to internet are not secured and intellectual properties respected.

A new technique applied in multimedia right management, called watermarking, introduces a new solution to overcome copy piracy. Associated with other technologies, it reveals to be very helpful to provide copyright protection, monitoring . A watermark algorithm embeds information into a multimedia bit stream. This information such as a label or a copyright is persistent, robust against natural and voluntary attacks, and not perceptible. The TELE laboratory of the Universite catholique de Louvain, a main actor in this domain shares and develops new techniques in accordance with European projects.

One of them, called ASPIS, brings forth tools for multimedia copyright protection. According to ASPIS objectives detailed in IST-12554, the aim of this proposal is the development of an innovative software protection system, which will protect the use of DVD-ROM executable and data files and will enable secure software or data access through the Internet. Due to illegal copying of applications and data files, the main outcome of the project is expected to be the restriction of illegal software and data copying and unauthorized use in Europe as well as worldwide. The authentication and protection in multimedia applications are based on watermarking tools.

During ASPIS project, UCL has developed a private watermarking algorithm for securely hiding information in images. This hiding information method combines an additive watermarking algorithm in the spatial domain providing resistance against cropping and exhaustive search and a synchronization template in the Fourier domain providing resistance against geometrical deformations. The additive watermark in the spatial domain is based on an original generalized 2-D cyclic pattern for secret message embedding. The cyclic property and pattern redundancy facilitate detection and synchronization against cropping and image processing basic attacks (like compression, filtering, blurring). This algorithm is complemented by a template insertion for getting resistance against rotation and scaling which are caused by print and scan processes. We generate the template in the Fourier domain inserting some points locally. The watermark and the template are weighted by a Human Visual masking function. The global scheme, though very classical in its global concept, provides a very efficient protection to digital image which could be delivered both in a digital high quality format and in a printed form.

During Paris bookfair from march 16th to march 21st, France Telecom Research and Development, Flammarion-Casterman and Université catholique de Louvain presented a new process to ensure a secure image delivery over Internet network fig.1.

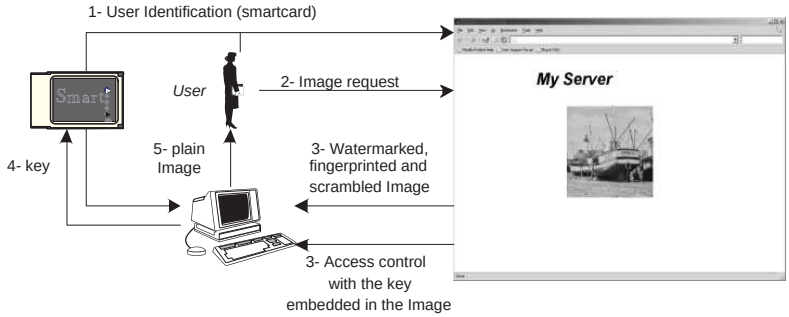

Figure 1: *User interface for a secure image delivery over Internet, ASPIS project*

These technologies are based on a cryptographic system called VIACCESS (access control) and watermarking method. User identification and safe transaction are built on access control. The user is identified by a smart card (a VIACCESS component) which contains his identity and private in-

formation (special rights and options). After transaction, images are watermarked by an author label and fingerprinted by the transaction label. The transaction label can be the date of the transaction between the user and the server. Secure web server and protocol ensure data confidentiality between Client and Server. Watermarking method allows a copyright protection after image delivery.

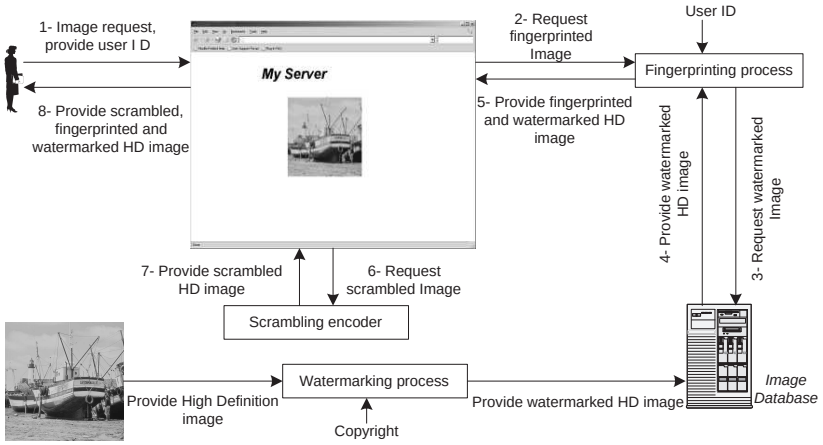

Figure 2: *Server design for a secure image delivery over Internet, ASPIS project*

A smart card with special rights is given to a public Web site user. This public Web site contains images in low resolutions called stamps. When a stamp is selected, the user is connected to a secure France Telecom Web site fig.2. If this smart card has the correct rights associated to the image category, the High Definition image is watermarked with a copyright, fingerprinted with private information, scrambled and provided to the visitor. If his smart card is present in smart reader, the visitor can view and print the unscrambled HD image. To test watermark robustness, we printed HD Image and we scanned the printed image.

The watermarking algorithm has been successfully processed for print and scan operations, a recovered watermark rate of 100 percent. It puts in prominent position its resistance against image processing and geometrical deformations. This algorithm, also tested in a projection room for digital movie protection, suffers from some deficiencies. Due to geometrical deformations, the distortions alter too much information embedded into the picture. But it is efficient against digital compression. Thanks to syn-

chronization block, the global watermarking scheme is resistant against geometrical deformations. But this Fourier domain template is not a low cost process and it is protected by international patents.

A new research project of the Wallon region named TACTILS highlights new security and real time requirements for digital cinema. Due to high data flows inherent to digital cinema, an advanced architecture for movie digital right management based on efficient algorithms and hardware integration is proposed. The global watermarking scheme used in ASPIS project and including an embedding message scheme and a synchronization block suffers from many troubles in real time application. A pattent-free method resistant against geometrical distortions is expected.

The classical method is based on a fingerprint extraction process with the help of the unmodified content. Before trying to read the embedded watermark message, we compensate the deformation applied to the analysed image. The synchronization block is removed from the initial watermarking architecture. The new design of the watermarking scheme called light watermarking algorithm is focussed on the pixel domain data hiding method. Due to the large number of authentication tests between original images and analysed images, this proposal is well adapted for small image database but not for digital cinema applications. Another inconvenient is the necessary access to original content.

Watermarking is largely used in multimedia protection as a copyright tool but it is not the only one. Digital signature is also widely used in digital communication as an authentication process and provides interesting properties such as easy to compute. The aim of the method presented in TACTILS project is a new one way function for images and video contents. This hash function for image and video content keeps most of cryprographic requirements. Regarding image constraints, some output hash function bit stream (called message digest) properties are modified. Two different images must have two different message digests. Two images are different if and only if image contents are different. The message digest must be resistant and robust, so remaining the same before and after attacks, if these attacks do not alter visual contents. In fig.3, the new visual hash function properties for image content lead to $MessageDigest1 \cong MessageDigest2 \neq MessageDigest4 \neq MessageDigest5$. The design of this hash algorithm is focussed on natural and voluntary attacks.

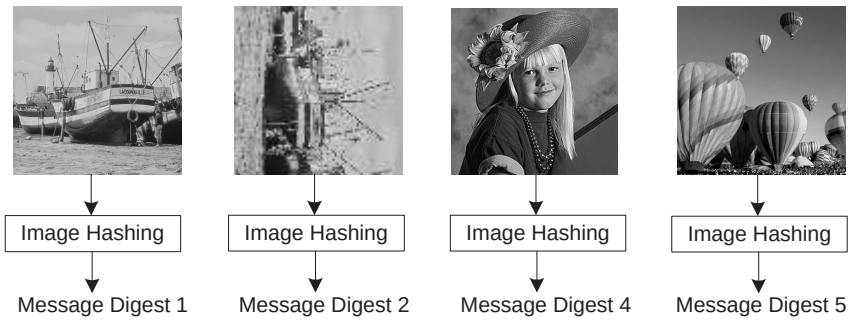

Figure 3: *Visual hash for images*

A light watermarking algorithm based on ASPIS algorithm, combined with a one-way function for video respects most of TACTILS requirements: security and quickness.

Overall, the objectives of this thesis are fourfold.
Chapter 1 presents a state of the art of image and content security.
Chapter 2 explores a new one-way function for digital images.
Chapter 3 describes a digital image signature extended to video.
Chapter 4 highlights a new advanced architecture for movie right management.

1

Image and content security: Overview

1.1 Digital Signature Standard

As described in [1] Digital Signature Standard appeared and was proposed in 1991 by National Security Agency and published in 1994 by Federal Register. Digital signature schemes can be used in data integrity (to be sure that data has not been altered), data origin authentication, and non-repudiation (to garanty that a person cannot deny previous actions). As defined in FIPS PUB 186-2 [2] explanation about DSS, "An algorithm provides the capability to generate and verify signatures. Signature generation makes use of a private key to generate a digital signature. Signature verification makes use of a public key which corresponds to, but is not the same as, the private key. Each user possesses a private and public key pair. Public keys are assumed to be known to the public in general. Private keys are never shared. Anyone can verify the signature of a user by employing that user's public key. Signature generation can be performed only by the possessor of the user's private key. A hash function is used in the signature generation process to obtain a condensed version of data, called a message digest (fig.1.1). The message digest is then input to the digital signature (ds) algorithm to generate the digital signature. The digital signature is sent to the intended verifier along with the signed data (often called the message). The verifier of the message and signature verifies the signature by using the sender's public key. The same hash function must also be used in the verification process."

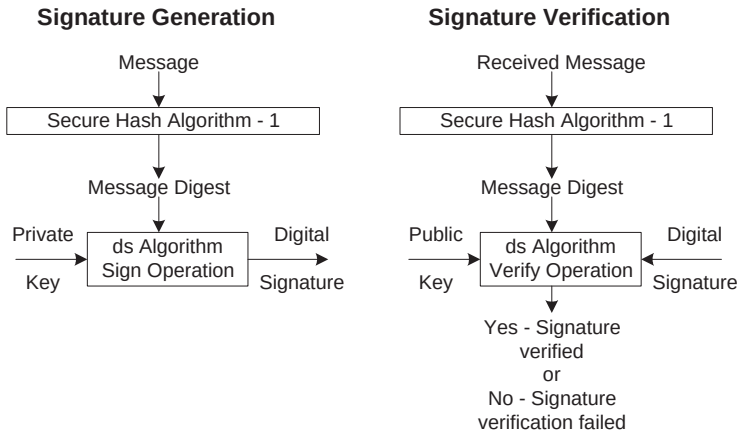

Figure 1.1: *Signature generation and signature verification operation*

Digital Signature Algorithm (DSA), Rivest-Shamir-Adleman (RSA) digital signature [ANSI X9.31], or Elliptic Curve Digital Signature Algorithm (ECDSA) [ANSI X9.62] are DSS algorithm and are full compliant with National Institute of Standards and Technology (NIST) requirements.

1.1.1 DSA-ECDSA

Digital Signature Standard or U.S. Government Federal Information Processing Standard (FIPS) of the NIST specified in August 1991 the Digital Signature Algorithm (DSA). DSA an evolution of ElGamal signature. Due to ElGamal design, based on Discrete Logarithm Problem (DLP), a bit length signature of 512 (or 1024 regarding some experts) bits is necessary to ensure a strong cryptography security. But for some applications this bits length signature is too big (smart card), and a new process needs to use only 160-bit length message to compute a 320-bit length signature (signature is twice bigger than message to sign).

DSA uses two prime numbers: the first called p with 160 bits (at least) and the second called q with 1024 bits (at least).

ECDSA is the elliptic curve analogue of the DSA. ECSDA parameters are defined in Appendice 6 of FIPS PUB 186-2[2]. For an analogue security constraints, the key used in ECDSA are less large than DSA. This type of Digital Signature algorithm are largely used in smard card crypto-processor.

A complete explanation and proposals of these three Digital Signature algorithm are described in FIPS PUB 186-2 [2].

1.1.2 Hash function, one-way function

In fact, a basic solution to sign a message is to divide the plain text in a several blocks of 160 bits, and then to cipher each block. Due to slow progress of the usual cipher design and size of a such signature, this technical proposal is not hold. The technical solution retained is the hashing function or one-way function. As detailed in the FIPS 186-2 , the hashing process provides a condensed version of the data, called also a fingerprint of the data. Stinson figure [1] described this process as below fig.1.2.

The message digest bit length depends on applications.

MD5 [3]proposed by Rivest, SHA [4] (and its evolution) designed by National Service Agency (NSA), are popular one way functions and largely

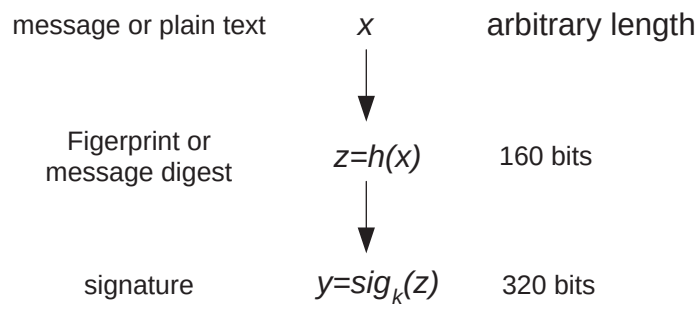

Figure 1.2: *Fingerprint signature*

used in crypto-systems. Based on MD4 considerations, the Secure Hash Standard (SHS) has been published in 1992 by the Federal Register.

Due to the use of a message digest instead of plain text for digital signature standard and in order to be secure in crypto-systems, the one-way function shares some properties. J. Borst thesis detailed these requirements.

- **Property 1**
  A one-way function or cryptographic hash function f has the property ease of computation: for every input x (from domain of f) f(x) is 'easy' to compute.

- **Property 2**
  A hash function f maps an input x of arbitrary bit length to an output h(x) of fixed bit length.

- **Definition 1: Preimage resistant**
  Given any image y, for which there exists an x with f(x)=y, it is computationally infeasible to compute any preimage x' with f(x')=y.

- **Definition 2: weakly collision-free**
  Given any preimage x it is computationally infeasible to find a 2nd preimage $x' \neq x$ with f(x)=f(x').

- **Definition 3: strongly collision-free**
  It is computaionally infeasible to find any two distinct inputs x,x' such that f(x)=f(x').

Due to technical improvements in image processing compression, some of these definitions and/or properties are not available for multimedia bit streams.

1.2 Image hashing

Accessing, organizing, and managing visual contents present technical challenges due to the large and always growing amount of contents to deal with, and to the lack of normalized and reliable way to describe the image attributes. Such description methods should be independent of the compression algorithm used to represent the content, and should reflect the similarity existing between "near-duplicate" contents.

In recent writting about visual content authentication, the term "image hashing" has been introduced to refer to the computation of a content-based image digest [16, 17, 18, 19, 22]. Following this terminology, we also call "hashing" the extraction of a content-based image or video digest, but we make the distinction between cryptographic hashing and robust hashing. Hash functions are well-known in cryptography and are generally used for digital signatures. In essence, they summarize a message in a short and constant bit length digest, which uniquely identifies the original message. Cryptographic hashing has to be resistant to collision, and computationally non-invertible [23] (i.e it should be computationally impossible to construct a different file producing the same hash value) . In cryptography, the output message digest dramatically changes when a single bit of the input message changes [23]. One says that cryptographic digests are discontinuous. Discontinuous hashes are useful to guarantee strong integrity and authenticity. However, in visual content management applications, one prefers *continuous hash* functions. A continuous hash function, also called robust hash function, alters the output message (or media) digest in proportion to the changes in the input message. When applied to image or video signals, such functions are designed to capture the essence of the visual content.

The purpose of *robust image hashing* is thus to define an image digest that satisfies two properties. In one hand, similar to cryptographic message digest, the robust image digest characterizes the image in the sense that it uniquely identifies its content, i.e. the digests derived from a pair of visually distinct inputs have a low probability to be identical. In the other hand, the hashing process is robust in the sense that the digest is only

slightly affected when the image changes due to compression or minor processing, i.e. visually indistinguishable images generate equal or similar digests. Conversely to cryptographic hashing, robust hashing is thus able to deal with visually non-significant changes of the content, and supports common manipulations like compression or reformating (e.g. spatial or temporal subsampling).

Because it defines a vector that identifies the image contents, robust hashing is an obvious solution for content identification and indexing. When used in combination with conventional cryptographic digital signature methods, robust hashing can also be used for integrity and authentication purposes [16, 24]. In watermarking, hashing enables the creation of payloads that depend on the media content, and which are thus resistant to the "copy attack" reported by Fridrich and al. in [25, 26].

1.2.1 Overview of our contribution

In [15], we present a technique for copyright protection and video recognition (RAdon Soft Hash algorithm). It is a new one-way function for images, based on the Radon transform, and adapted to the particularities of image and video signals. A soft hash process computes an invariant output bit stream for each image. According to [15], two images are different if and only if image contents are different. The computed message digest is resistant and robust, and thus remains the same before and after attacks, if these attacks do not alter visual contents fig.1.3. From fig.1.3, $MessageDigest1 \cong MessageDigest2 \cong MessageDigest3$ for same contents.

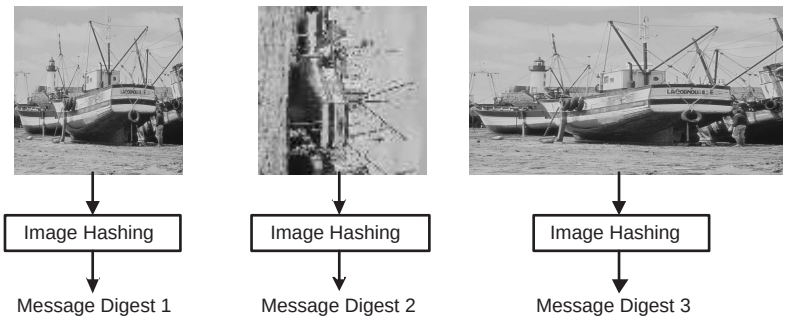

Figure 1.3: *Computation of message digest for images*

In [14], we improve its authentication properties and define a new hash function (RADIal Sof Hash algorithm) based on variance computation along radial projection. Presented in [14], an extension to video application allow message digest usage as an alternative and/or a complement to watermarking process due to the geometrical properties of the message digest.

1.2.2 Related works

There has been a number of proposals to provide visual robust hashing.

Traditional image retrieval systems are based on a large set of different image features, including color histograms [1], textural features [18], edge density and edge direction [19], or even textual semantic features [20]. These features share an important property with the features considered for robust image hashing: both are robust towards slight geometrical or processing distortions. However, the features considered by a robust image hashing algorithm differ from retrieval systems features in the sense that, for hashing purposes, the features have to provide a strong resistance to collision. This property is relaxed in retrieval systems.

In [25] and [26], Fridrich and Goljan propose to extract an image feature vector based on a decomposition of the image into blocks. Each block defines a binary symbol of the feature vector from the analysis of the block luminance distribution on a random spatial pattern. This method relies on a good synchronization between the image and the spatial pattern used to extract the binary symbols. So, it is not expected to be robust to geometrical distortions, e.g. due to slight image geometric deformation, or to motion in a video.

In [18, 24, 21], the extracted feature is based on the invariance of the relationships between DCT coefficients at the same position in separate blocks of an image, when the DCT coefficients are quantized by the compression engine. This approach is robust to compression and low-pass filtering, but does not survive to geometrical transformations like scaling or rotation.

In [16], Bhattacharjee and Kutter proposed an authentication method based on the extraction of "salient" image feature points. This approach is interesting in the sense that it captures fundamental structures in the image. In [16], the authors are interested in checking whether two sets of structures are identical or not. On the contrary, in the context of our video hashing algorithm, we are interested in measuring the similarity between two set of structures. Indeed, we have observed in our experiments

that the representative frames corresponding to visually similar sequences may slightly differ. As a consequence, defining a continuous measure of the distance between two sets of structures is a key issue to extend the feature extraction proposed in [16] to our video hashing problem.

In [19], the average values of a random and secret rectangular tilings of image wavelet subbands are used as a feature vector. Whilst being computationally more complex, this method appears to be a valid alternative to our proposed RADISH algorithm from a functional point of view. An additional interesting contribution of [19] is the use of Reed-Muller error correcting codes to build a robust binary hash value from the quantized image digest. This approach can directly complement our proposed RADISH digest method.

In [22], the authors propose a video hashing method based on spatial and temporal subtraction of the average luminances computed on a set of blocks that are defined within a group of 30 pictures. This approach performs well, but lacks of robustness towards temporal subsampling or shifting.

In [22], Johnson and Ramchandran assume the availability of a robust digest extracted from the image, and propose to exploit distributed compression principles after the digest extraction to guarantee its information-theoretic security. This contribution complements our RADISH image digest algorithm, and makes it useful for authentication purposes.

For completeness, it is worth noting that robust hashing has also been considered for audio content [27, 28].

1.3 Watermarking

Watermark a signal (audio or video) is to embed a robust but not perceptible information in the signal. Watermarking is described as a digital signature technic for multimedia stream. Watermarking scheme are largely used to ensure most of multimedia stream. We can find following applications:

- copyright protection.
- monitoring: copyright verification.
- multimedia streaming tracking, called also fingerprinting.
- copy attack protection, e.g DVD copy.

- document authentication.
- labelling or indexing tool in a database.

Some attacks can be applied to remove watermark such as sharpening, cropping, blur effects, jpeg compression. But all of these attacks must respect image quality. If the attacks brought too many visual distortion, *watermarking is a nonsens* [20]. In [20], Herley compared watermarking and cryptography methods and concluded that watermarking is a nonsens. In [21], the author brings number of arguments such detectors computational complexity in watermarking and valuable sets. Cryptography and watermarking requirements are different, and their usage scenario are also different.

Specifications of watermarking schemes depend on their usage scenario. Often, the design of a watermarking algorithm focuses on specific attacks leads to requirements which are quite different than requirements for getting resistance against geometric attacks such as rotations, stretching, cropping... It is very difficult to hold out against both kinds of attacks (e.g : stirmark [7]).

In a global watermarking embedding scheme fig.1.4, the psychovisual mask, the watermark pattern and the synchronized block are in a prominent position.

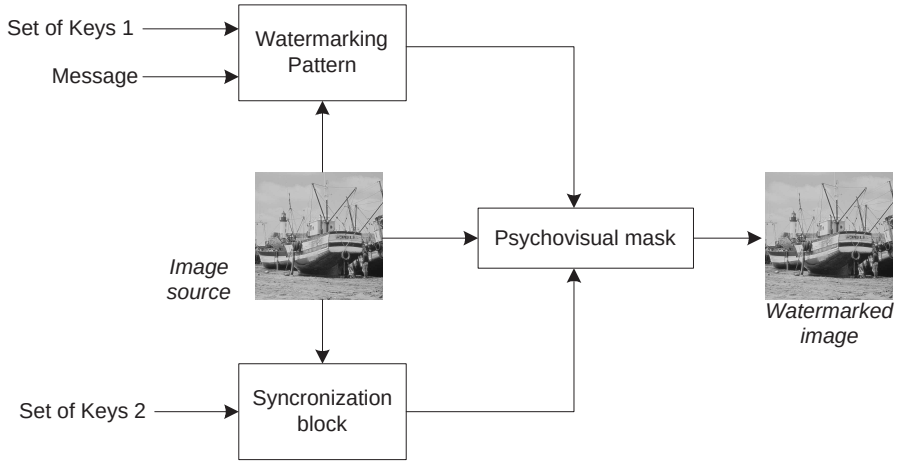

Figure 1.4: *Global watermarking scheme.*

1.3.1 The psychovisual mask

A perform watermarking algorithm is to be not visible and robust [13]. Robustness is guaranted by the quality of the insertion scheme and invisibility by a psychovisual mask. The purpose of this mask is to give more or less influence on the watermark to embed. Hence, mask weights the bits (or intensity of pixels) to modify in a image.

1.3.2 The watermarking pattern

The watermarking pattern is the aim of a global watermarking scheme. The working domain, used to embed the information or pseudo-mark into the multimedia stream, depends on domain compression of the media. Zao in [10] prefers DCT domain for JPEG compression, Barni in [9] inserts in wavelet domain to be full compliant with JPEG2000 standard whereas Kalker in [11] embeds in spatial domain for uncompressed multimedia stream.

For any type of watermarking scheme, the global architecture respects the following insertion-extraction scheme fig.1.5

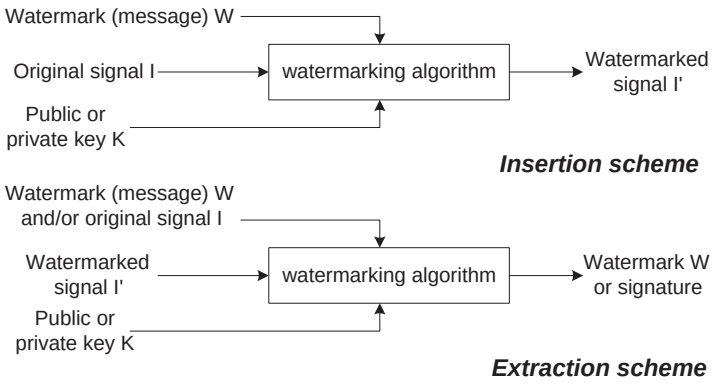

Figure 1.5: *insertion/extraction scheme.*

According to the previous figure fig.1.5, watermarking scheme allows:

- Public or blind watermarking : It requires neither signal source I nor the mark embedded. To extract message from signal I' or to insert message in signal I, the same keys are used.
- Asymmetric watermarking : Introduced by T. Furon in [6], this method is very difficult and needs two distinct watermarking block

for insertion and extraction process. The extraction process is based on theoretical detection method and it does not depend on embedding watermark scheme. With this watermarking scheme, the mark cannot be removed.

- Private Watermarking : In private watermarking scheme, the image source I is used to extract embedded information W from watermarked image I. Usually two types of information can be extracted from a private watermarking scheme: whole of the embedded bits or a bit of signature (presence or not of the mark) in the candidate watermark picture. This kind of watermarking scheme is more robust than the other one.

1.3.3 The synchronized block

Domain insertion and resistance have an important relationship. In [5], some points are embedded in Fourier transform domain. The aim of this algorithm is to resist against print and scan operations. In fact, print and scan hightlight a lot of attacks: scaling, rotation with blur effect and some MIRE default produced by a basic printer. Domain transform such as Fourier transform provides some interesting properties. A scaling transformation in the spatial domain corresponds to a scaling with an inverse factor in the Fourier domain.

$$\begin{aligned} TF(f \circ S(S_x, S_y))(u,v) &= \alpha \int_{\mathbb{R}^2} f(S_x.x, S_y.y) e^{-(ux+vy)} d_x d_y \\ &= \alpha \int_{\mathbb{R}^2} f(X,Y) e^{-(\frac{ux}{S_x}+\frac{vx}{S_y})} d_x d_y \\ &= TF(f).S\left(\frac{1}{S_x}, \frac{1}{S_y}\right)(u,v) \end{aligned}$$

A rotation in the spatial domain has the same effect in the Fourier domain.

$$\begin{aligned} TF\left(f\right)(u,v) &= \alpha \int_{\mathbb{R}^2} f(x,y) e^{-(ux+vy)} dxdy \\ &= \alpha \int_{\mathbb{R}^2} f(R_\theta\left(x,y\right)) e^{-(ux+vy)} dxdy \\ &= \alpha' \int_{\mathbb{R}^2} f(X,Y) e^{-\left((u,v).R_{(-\theta)}(X,Y)\right)} dXdY \end{aligned}$$

$$\begin{aligned} TF\left(f \circ R_\theta\right)(u,v) &= \alpha' \int_{\mathbb{R}^2} f(X,Y).e^{-R_\theta(u,v).(X,Y)} dXdY \\ &= \alpha'' TF\left(f\right).R_\theta\left(u,v\right) \end{aligned}$$

Using both spatial and Fourier domain specifications, a global image embedding method is presented in [5] and detailed in the last chapter. This approach is followed by several research groups (including Digimarc, University of Geneve, University of Firenze ...) and suffers from weaknesses due to the additive structure of the watermark (the so-called template and transposition attacks).

Many watermarking scheme are designed for typical applications and for typical working domain. Such as our contribution in hashing image, few of watermarking design are content-based. A content-based watermarking scheme reaches a relevant rate of watermark recovery after geometrical deformations.

1.3.4 Related work in content-based watermarking design

As described in [29], the design of this type of watermarking uses feature points of the image as content descriptor. The algorithm build its reference sytem on content characteristics. In [29], P. Bas proposed an embedding and detection scheme based on salient points to define a content descriptor. Salient points are located in corners and closed to image edges. Many detectors are tested in [29], and Harris detector is relevant for robust detector. After features points extraction, a Delaunay tesselation is performed to produce a image partitioning into disjoint triangles. The signature is inserted in each triangle of the tessellation fig.1.6.

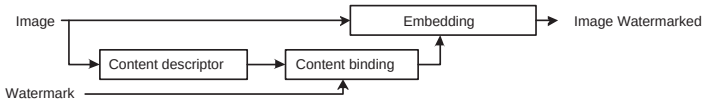

Figure 1.6: *Content based watermarking scheme by P.Bas*

The Delaunay tessallation is sensitive against feature points. Due to fragile salient points extraction, this approach suffers from image manipulations.

Bibliography

[1] D. Stinson, "Cryptography Theory and Practice", CRC Press, Inc.,1995

[2] National Institute of Standards and Technology,"Digital Siganture Standard (DSS)", FIPS PUB 186-2

[3] R. RIVEST "RFC 1321: The MD5 Message-Digest Algorithm." RSA Data Security Inc., April 1992.

[4] National Institute of Standards and Technology (NIST), "Announcement of Weakness in the Secure Hash Standard", 1994.

[5] F. LEFEBVRE, D. GUELUY, D. DELANNAY and B. MACQ,"A print and scan optimized watermarking scheme", Multimedia Signal Processing, 2001, p511-516, Cannes, France.

[6] T. FURON and P. DUHAMEL, "An Asymmetric Public Detection Watermarking Technique", Information Hiding 1999,pp. 88-100.

[7] F.A.P. PETITCOLAS, R.J. ANDERSON and M.G. KUHN, "Attacks on copyright marking systems", in Information Hiding:2nd Workshop, vol.1525, D. Aucsmith, Ed. Berlin, Germany:Springer-Verlag, 1998.

[8] I.J. COX, J. KILIAN,T. LEIGHTON, and T. HAMMON, "A Secure, robust watermark for multimedia", in Proc. Workshop on Information Hiding, vol.1, pp. 244-250, April 1992.

[9] M. BARNI, F. BARTOLINI, V. CAPELLINI, A. LIPPI and A. PIVA, "A DWT-based technique for spatio-frequency masking of digital signatures", Proceeding of SPIE vol. 3657, Electonic Imaging '99, San Jose, CA, January 1999.

[10] J. ZHAO, and E. KOCH, "Embedding robust labels into images for copyright protection",In: Proc. of the Int. Congress on Intellectual Property Rights for Specialized Information, Knowledge and New Technologies, Vienna, August 1995.

[11] T. KALKER, G. DEPOVERE, J. HAITSMA and M. MAES, "A video watermarking system for broadcast monitoring", in Proc. SPIE

IS&T/SPIE's 11th Annu. Symp., Electronic Imaging '99: Security and Watermarking of Multimedia Contents, vol. 3657, Jan. 1999.

[12] T. FURON and P. DUHAMEL, "An Asymmetric Public Detection Watermarking Technique", Information Hiding 1999,pp. 88-100.

[13] J.-F. DELAIGLE, C. DE VLEESCHOUVER, and B. MACQ, "Watermarking algorithm based on a human visual model", Signal Processing, Vol. 66, n3, May 1998, pp. 319-336.

[14] F. Lefebvre, C. DeRoover, C. De Vleeschouwer and B. Macq, "An invariant soft video digest", Manuscript submitted on IEEE International Conference on Image processing 2004

[15] F. Lefebvre, B. Macq and J.-D. Legat, "RASH : RAdon Soft Hash algorithm", EUropeean SIgnal Processing COnference, 2002, Toulouse

[16] S. Bhattacharjee, and M. Kutter, "Compression tolerant image authentication", in *Proceedings of the 5th IEEE International Conference on Image Processing*, vol. 1, pp. 435-439, Chicago, USA, October 4-7, 1998.

[17] G.L. Friedman, "The trustworthy digital camera: restoring credibility to the photographic image," *IEEE Transactions on Consumer Electronics*, vol. 39, No. 4, pp. 905-910, November 1993.

[18] C.-Y. Lin, and S.-F. Chang, "Generating robust digital signature for image/video authentication", in *Proceedings of Multimedia and Security Workshop, ACM Multimedia*, Bristol, UK, September 1998.

[19] R. Venkatesan, S.M. Koon, M.H. Jakubowski, and P. Moulin, "Robust image hashing," *Proceedings of the International Conference on Image Processing*, September 2000.

[20] C. Herley, Why watermarking is nonsense, IEEE Signal Processing Mag., vol. 19, no. 5, pp. 1011, Sept. 2002.

[21] P. Moulin, " Comments on Why Watermarking is Nonsense",IEEE Signal Processing Magazine, November 2003, pp.58-59.

[22] J. Oostveen, T. Kalker, and J. Haitsma, "Visual hashing of digital video: applications and techniques," *SPIE applications of digital image processing XXIV*, July/August 2001, San Diego, USA.

[23] A. Menezes, V. Oorschot, and S. Vanstone, "Handbook of applied cryptography", CRC Press, 1998.

[24] 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*, February 2001.

[25] J. Fridrich, and M. Goljan, "Robust hash functions for digital watermarking," *ITTC 2000*, Las Vegas, USA, 2000.

[26] J. Fridrich, "Visual hash for oblivious watermarking," *Proc. SPIE Photonic West Electronic Imaging, Security and Watermarking of Multimedia Contents,* pp. 286-294, San Jose, CA, USA, 2000.

[27] J. Haitsma, T. Kalker, and J. Oostveen, "Robust audio hashing for content identification", *Proceedings of the International Workshop on Content-Based Multimedia Indexing*, Brescia, Italy, September 2001.

[28] M. Kivanç Mihçak, and R. Venkatesan, "A tool for robust audio information hiding: a perceptual audio hashing algorithm," *Proceedings of the Information Hiding Workshop*, Pittsburgh, USA, 2001.

[29] P. Bas, J-M Chassery and B. Macq, "Geometrically Invariant Watermarking Using Feature Point", IEEE Transactions on Image Processing , Vol 11, 9, pp 1014-1028, 2002.

2

Authentication and Geometrical Attacks Detection for Image Signature

2.1 Introduction

The authentication scheme based on Digital signature, asserts that an adversary can not compute a fake message exhibiting the same signature than an original one. Classical cryptography provides tools to insure data integrity, data origin authentication and non repudiation. Classical cryptography does only deal with bits, not with contents: two documents are considered different if and only if their bit streams are different. This non collision property is given by hash function property. According to the first chapter, to be cryptographically secure, the two important hash functions properties are:

- A short, constant and unique bit length digest.
  This hash function provides a unique output called message digest for each input. In other word, if some bits of the input are modified, the digital signature provided by the hash function will differ from the original signature. It is desirable to have as few collisions as possible.

- Non-invertible.
  It must be computationally infeasible to reverse the process, i.e it must be impossible to find a fake message exhibiting the same digital signature.

For image applications [6, 7], one could want to provide signatures for image contents instead of signatures for image particular binary representations. Several binary representations can be found for the same image content. In this case, image hashing functions have to ensure unicity of digests related to image content : two different image contents must lead to two different message digests. The message digest must be resistant and robust to binary manipulations which do not alter the image contents, so remaining the same before and after attacks [8], if these attacks do not modify visual contents. The design of this hash algorithm focused on such imaging attacks : blur, sharpening, compression, noise insertion, rotation, scaling and stirmark [8], leads to requirements which are quite different from those that are required for text document.

The main idea behind our design was to extract features derived from angular projections of the image, in order to obtain some resilience to rotation and scaling. Two additional requirements have guided our design. Firstly, the extracted features should provide good image characterization

capabilities. Secondly, they should be robust towards classic image processing operations that do not dramatically change the visual appearance of the image, e.g. blurring and compression. A simple way to achieve robustness towards filtering-kind of operations is to compute the features based on a large spatial support. Obviously the size of the support trades off robustness for complexity. But it also trades off robustness for content characterization capability. Indeed, intuitively, a too large support is unable to capture sharp and local transitions that characterize the image.

The Radon transformation largely used in medical image processing [9] provides a good basis for our design.

The amount of elements in transform domain is almost the same as the pixel domain when perfect reconstruction is required. It is however possible to further reduce the amount of coefficients to realize a real soft hash function. From Radon transform, some robust and almost invariant elements can be extracted.

The Radon transformation [9] provides some mathematical properties. From Radon transform, some robust, relevant and almost invariant elements can be extracted figure 2.1.

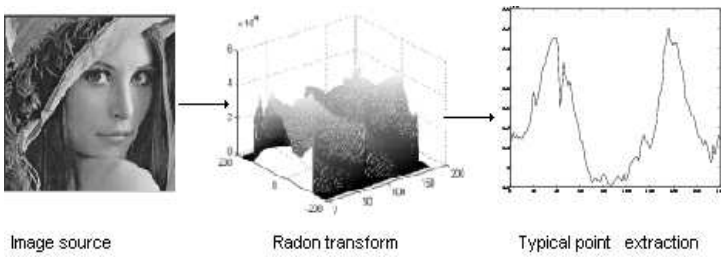

Figure 2.1: *Typical point extraction*

In the following sections, we describe our robust and invariant hash function for images.

2.2 Radon Transform

The Radon transform is largely used in medical image processing. In tomography, when a bundle of X-Rays goes through an organ, its attenua-

tion depends on content of organ, distance, and direction or angle of this projection.
This set of projections is called Radon transform.

2.2.1 Continuous Radon transform

In two dimensions, we can illustrate it by the figure fig.2.2

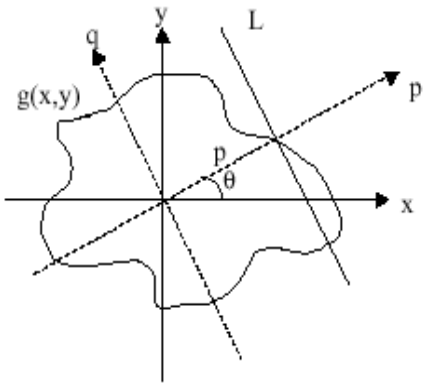

Figure 2.2: *Projections*

The Radon Transform 2.2 is given by the integrals :

$$\mathcal{R}\mathrm{g(x,y)} = \int_{\mathrm{L}} \mathrm{g(x,y)\,dl} \tag{2.1}$$

Where L is given by:

$$p = x.cos\theta + y.sin\theta \tag{2.2}$$

So each projection is a line integral of g(x,y) along the $p - axis$ and with the θ direction. To express this integral in another way, we can simply use a variable's change:

$$x = p.cos\theta - q.sin\theta \tag{2.3}$$
$$y = p.sin\theta + q.cos\theta \tag{2.4}$$

This new representation is given by the fig. 2.3.

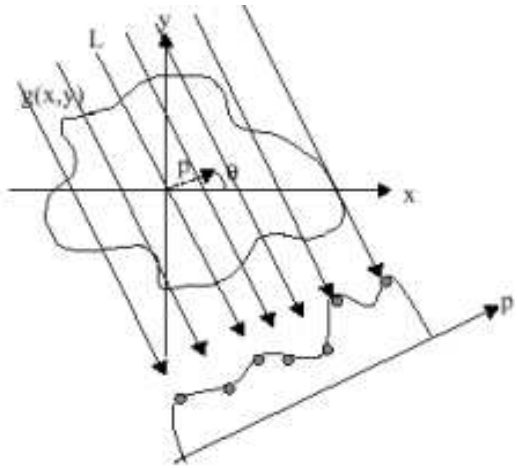

Figure 2.3: *line integral of Radon*

Mathematical equation of this transformation becomes:

$$\mathcal{R}\mathrm{g}(p,\theta) = \int_{-\infty}^{\infty} g(p.cos\theta - q.sin\theta, p.sin\theta + q.cos\theta)\, dq \tag{2.5}$$

The mathematical expression of Radon transform leads to some very useful properties.

- If a set (images) g is shifted by (x_0, y_0), the Radon transform is

$$g(x - x_0, y - y_0) \longleftrightarrow \mathcal{R}\mathrm{g}(p - x_0.cos\theta - y_0.sin\theta, \theta) \tag{2.6}$$

- If a set (images) g is rotated by ϕ, the Radon transform is

$$g(x.cos\phi - y.sin\phi, x.sin\phi + y.cos\phi) \longleftrightarrow \mathcal{R}\mathrm{g}(p, \theta + \phi) \tag{2.7}$$

- If a set (images) g is scaled by a factor α, the Radon transform is

$$g(\alpha.x, \alpha.y) \longleftrightarrow \frac{1}{|\alpha|}.\mathcal{R}\mathrm{g}(\alpha.p, \theta) \tag{2.8}$$

- There is energy conservation in the Radon transform and in the space domain

$$E^2 = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} |g(x,y)|^2\, dx\, dy \longleftrightarrow \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} |\mathcal{R}g(p,\theta)|^2\, dp \quad (2.9)$$

The sinograms (projections taken along the angular direction) of Lena and Lena rotated show us the rotation property of Radon Transform

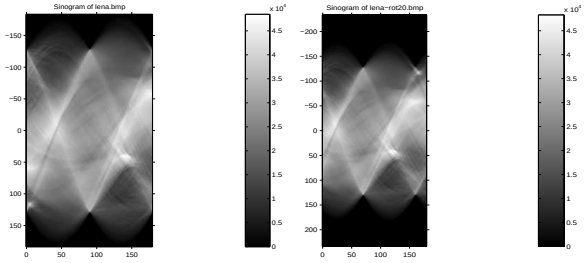

Figure 2.4: *Sinograms of Lena and Lena rotation of* 20^o

Basically, the second property states that the Radon transform of a rotated image is simply translated by the corresponding angle. The third property shows that when an image is scaled, its Radon transform is scaled by the same factor and the magnitude is simply divided by the scale factor.

Thanks to Radon Transform invariance properties, the robustness against image rotation and scaling is intrinsically achieved. Furthermore, by extracting some invariant features of the Radon transform of the image, the message digest is sensitive to the image content but not to minor pixel modifications that arise from blurring and compression operations. All previous properties are avalaible in continuous domain. An efficient discrete representation has to be used for our application.

2.2.2 Discrete Radon transform

To explain the soft hash algorithm for images, the image will be NxN square and converted in gray level. The Lena picture will be taken as test image.

The line integral along $x.\cos\phi + y.\sin\phi = d$ is approximated by a summa-

tion of the pixels lying in the one-pixel-wide strip

$$d - \frac{1}{2} \leq x.\cos\phi + y.\sin\phi < d + \frac{1}{2} \tag{2.10}$$

Since strips have unit width, d can be restricted to integer format values, and for a given angle, the number of strips needed in the addition is limited by

$$N \leq n(d, \phi) \leq \sqrt{2}.N$$

This method gives a quick and a good approximation of discrete Radon transform (fig.2.5). A better implementation is described in [10].

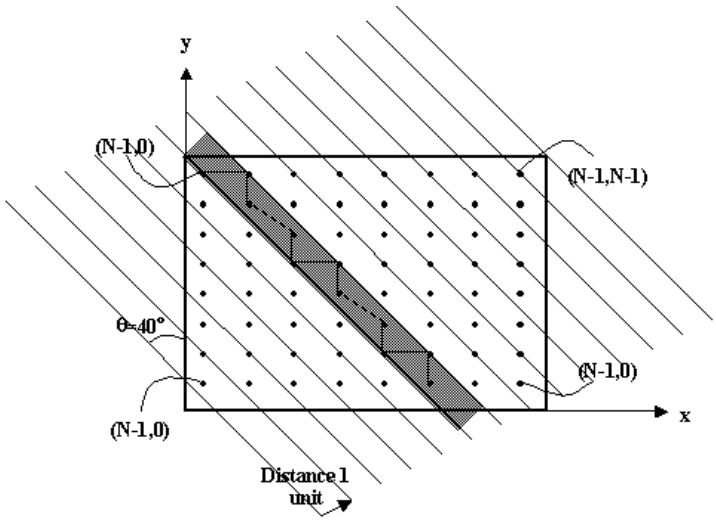

Figure 2.5: *Discrete Radon transform*

These projections give a unique representation for each image. But this set has the same cardinality as the image in space domain. Rotation and scaling spread the signal. In the Radon transform domain, we need to find some invariant points included in a set of fixed-length element.

The next section develops the hash function and explains how to find these invariant and robust points.

2.3 Radon Soft Hash Function

2.3.1 Points extraction

Due to mathematical properties, rotation and scaling spread the signal. If we extract some points from each projection for each angle, it is very difficult to retrieve these points as shown in figure (2.6), figure (2.7) and figure(2.8).

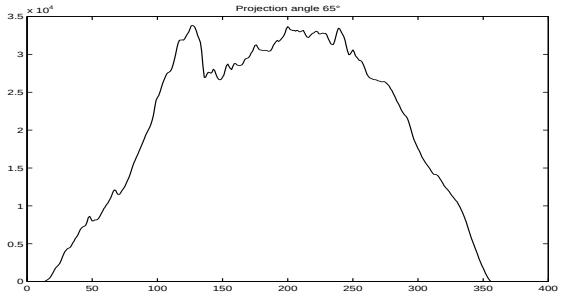

Figure 2.6: *Projection with $\theta = 65^o$ for original image of Lena*

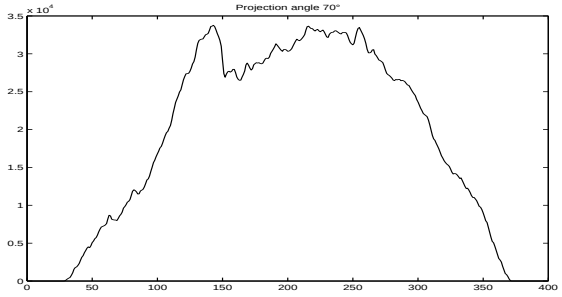

Figure 2.7: *Projection with $\theta = 70^o$ for Lena rotation of 5^o*

The main aim of RAdon Soft Hash algorithm [1] is to be resistant against common attacks applied to images (scaling, rotation, blurring...). The figure of sinogram (fig.2.9) representation focuses on spread projections $p(\phi)$. The axis range $p(\phi)$ depends on the size of the image. If Lena is rotated or scaled by a factor bigger than one, the axis range $p(\phi)$ is also spread. If

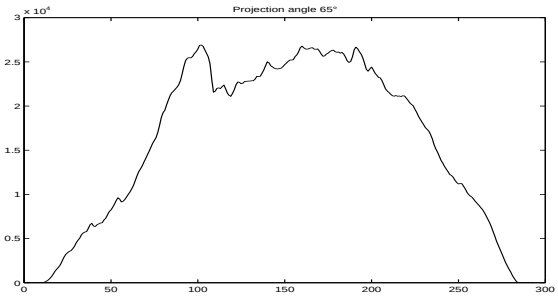

Figure 2.8: *Projection with $\theta = 65^o$ for Lena scaling of* 0.8

$p_{min} = 0$, and N the size of a square image,

$$p_{max} = \sqrt{2}.N \tag{2.11}$$

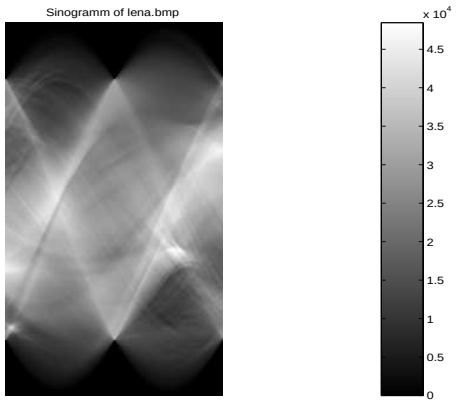

Figure 2.9: *Sinogram of Radon transform*

The $p(\phi)$ axis range is never the same, it depends on the rotation angle and the scaling factor. Invariant points have to be insensitive to spread range. Only one kind of points are invariant: the medium points of each projection for each angle fig.2.10. These medium points keep all Radon transform properties.
So the RASH algorithm computes a kind of projections that goes through one selected medium point. For a NxN image, computation of medium

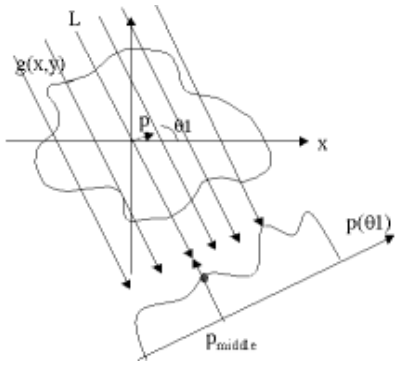

Figure 2.10: *Typical Points extraction*

points is based on the following middle point fig.2.11:

$$p_{middle} = \frac{N}{2} \tag{2.12}$$

$$\phi_{middle} = 45^0 \tag{2.13}$$

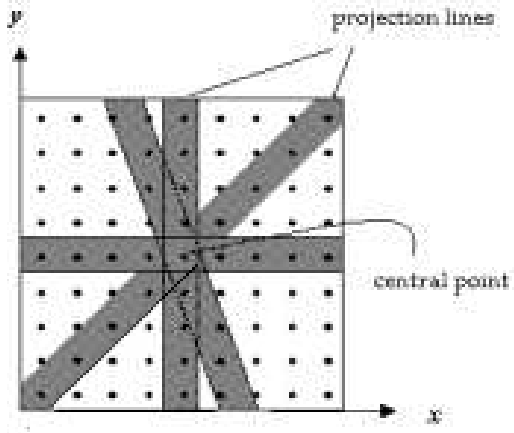

Figure 2.11: p_{middle} *and strip of discret Radon transform*

2.3.2 Features extraction algorithm

We will now further describe our RASH algorithm applied on a NxN image.

STEP 1.
The middle point is selected by

$$x_m = \lfloor p_{middle}. \cos \phi_{middle} \rfloor \tag{2.14}$$
$$y_m = \lfloor p_{middle}. \sin \phi_{middle} \rfloor \tag{2.15}$$

STEP 2.
For a uniformly distributed set of 180 angles ϕ, with $0 \leq \phi \leq 180$ discretized with 1^0 sampling angle, each pixel can be projected regarding the medium line path, given by the coordinates of the medium point :

$$d_m = x_m. \cos \phi + y_m. \sin \phi \tag{2.16}$$

The line path is called also the strip.
Considering $A(\phi)$ is a set of (x, y) for a given angle, ϕ belongs to the same strip that the middle point. Following the expression of the discrete Radon transform (2.10) we can express this relationship by:

$$(x, y) \in A(\phi) \quad if and only if \quad d_m - \frac{1}{2} \leq x. \cos \phi + y. \sin \phi < d_m + \frac{1}{2} \tag{2.17}$$

Finally, we add to the integral along the strip the value of all the pixels that meet this condition. $R[\phi]$ is the result of the addition for a given angle. The expression of our operation can be written as:

$$R[\phi] = \sum_{(x,y) \in A(\phi)} I(x, y) \quad \forall (x, y) \tag{2.18}$$

Applying this feature extraction on all angles discretized with 1^0 sampling angle, the output streaming contains 180 elements, i.e one element per angle. This discretization of Radon transform is π symetric, $R[\phi + \pi] = R[\phi]$, so we only need the first 180 projections to compute our message digest. An exemple of such a digest is shown in fig.2.12.
The summation in the discrete Radon domain along several directions provide some resistance against image processing attacks. So, any attacks in space domain such as blurring, sharpenning, does not modify completly these typical points but give a smooth signal of the 180 extracted points.

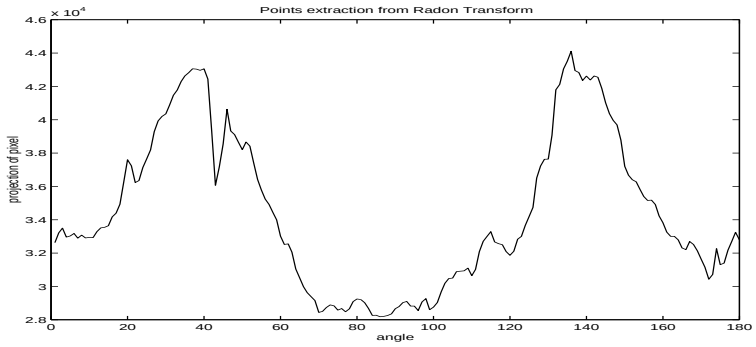

Figure 2.12: *Feature extraction from Radon transform*

2.3.3 Geometrical deformation detection

Due to mathematical properties described in eq.2.7 and eq.2.8, geometrical attacks are easy to detect.

Scaling detection

Following the scalling property eq.2.8, we intend to detect the scale factor of the signature. If we consider an original image $I(x, y)$ with its signature $S(\phi)$, and its scale image $I'(x, y)$ with its $S'(\phi)$, the factor of scaling a can be recovered by an energy relation. If the energy of the original RASH signature is E_s and E'_s the energy of the scaled image, then:

$$E'_s = \frac{E_s}{a^2} \Rightarrow a = \sqrt{\frac{E_s}{E'_s}} \tag{2.19}$$

with:

$$E_s = \sum_{k=0}^{N-1} S(k)S(N-1-k) \quad where \quad N = 180 \tag{2.20}$$

$$E'_s = \sum_{k=0}^{N-1} S'(k)S'(N-1-k) \quad where \quad N = 180 \tag{2.21}$$

And the factor of scaling is perfectly recovered.

Rotation detection

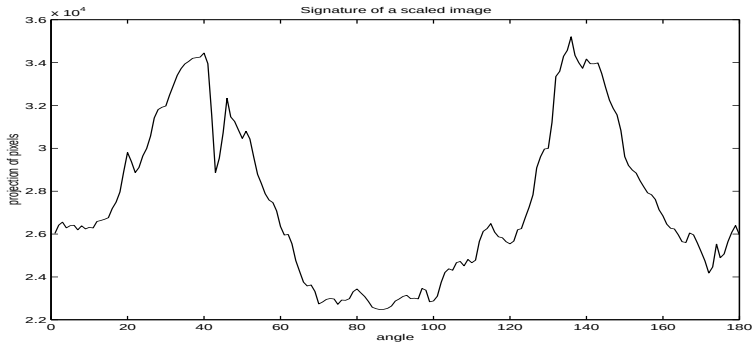

Figure 2.13: *Signature of lena scaled*

To implement the metric of distance between the signatures of two images we use a classic normalized cross-correlation algorithm:

$$R_{S'S}(n) = \sum_{\phi=0}^{N-n-1} \left(S_\phi - \frac{1}{N}. \sum_{i=0}^{N-1} S_i \right) . \left(a.S'_{n+\phi} - \frac{a}{N}. \sum_{i=0}^{N-1} S_i \right) \quad , N = 180 \tag{2.22}$$

where a is the energy normalizing factor given by the previous step. If the two signatures are computed on the same image with or without attacks, the peak of $R_{S'S}(n)$ ($0 \leq n \leq 359$) is closed to 1 and the rotation ϕ_0 is detected by:

$$\phi_0 = 180 - argmax_n(R_{S'S}(n)) \tag{2.23}$$

In the exemple presented in the fig. 2.14, $\phi_0 = 180 - 170 = 10$ and the peak of (normalized) cross-covariance is 1.

2.3.4 Detection and experiments

Cross correlation is an efficient metric to estimate the likehood between two signatures. The Mean Square Error (MSE) gives us an other metric to evaluate the distance between the two signatures:

$$MSE = \frac{\sum_{i=0}^{N-1} (S_i - S'_i)^2}{N} \tag{2.24}$$

If the candidate image content (baboon, barbara,...) does not match with

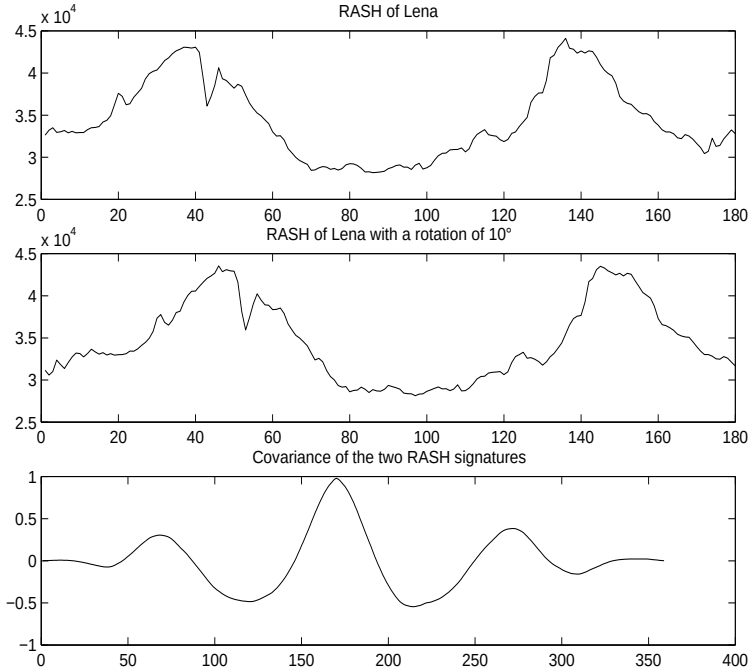

Figure 2.14: *Signature of lena rotated*

Ref:Lena	$max(R_{xy})$	$argmax(R_{xy})$	MSE
Lena-scale(0.8)	0.99	180	$7.4.10^{-4}$
Lena-sharpen*2	0.99	180	$1.9.10^{-3}$
Lena-blur*2	0.99	180	$7.8.10^{-4}$
Lena-stirmark	0.99	180	$4.4.10^{-3}$
baboon	0.6	181	0.8
barbara	0.65	187	0.82
fishingboat	0.45	265	1.19
houses	0.7	183	0.4
peppers	0.6	181	0.8

Figure 2.15: *collision and detection tests*

the image reference content (Lena), the message digest from database's images leads to a different candidate visual hash.

This algorithm generate an array of 180 elements, each element corresponding to one angle's projection. This basic message digest of the RASH algorithm allows us to authenticate and recognize an image even if a geometrical attack is applied.

2.4 Message digest for digital signature

Applying this extraction on all angles discretized with 1^0 sampling angle, the output set contains 180 elements, i.e one element per angle. Some tests and experiments [1] gave efficient results to detect and recognize two images with two signatures based on RASH algorithm. To realize our soft hash function with a short (1024-bit-length or 160-bit-length) output according to Secure Hash Standard, we need to compress these typical points provided from RASH. The next section explains how to reduce the cardinality and how to decrease the risk of collision attacks.

2.4.1 Normalized RASH

The amount of points in a strip depends on the projection's angle. Considering a square image, for angles closed to 45^0 and 135^0, the summation of intensity pixel projection is $\sqrt{2}$ higher than the summation of intensity pixel projection for angles closed to 0^0 and 90^0. A pattern can be modelized in the RASH message digest. The values in the middle tend to follow a trigonometric pattern between the edges. The experimental results confirm that the cross-correlations tend to be high, even for different images.

Therefore, a pattern-breaking effort is necessary in order to avoid a great number of collisions. By normalizing the summation of luminance projection for each angle by the amount of pixels added, the visual hash takes care of image size and energy in certain direction. The message digest is now computed by the mean luminance of the projection for each angle.

So, the equation 2.18 must be redefined as following: where $N(\phi)$ is the number of pixels that belong to $A(\phi)$ with $A(\phi)$ the set of pixel (x, y) for a direction ϕ:

$$R[\phi] = \sum_{(x,y)\in A(\phi)} \frac{I(x,y)}{N(\phi)} \quad \forall (x,y) \tag{2.25}$$

The normalization keeps all previous properities, excepted scale detection fig.(2.16).

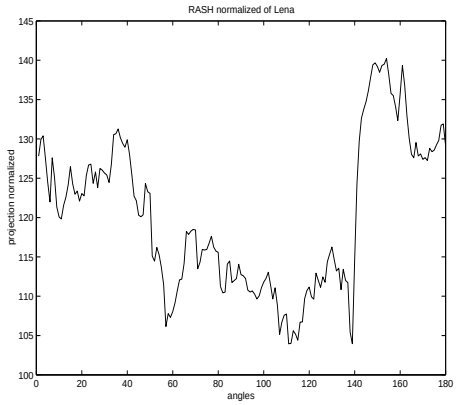

Figure 2.16: *RASH normalized*

Due to normalized projection by the amount of pixels number along this projection, the scaling factor estimation is lost.

2.4.2 Final message digest

Using normalized RASH, we have an output of 180-float stream representing the mean luminance of each projection and another float stream representing the energy. This length of output bit stream is too big compared to the usual hash message digest. So we need to develop a compression algorithm for the RASH output.

Discrete Cosinus Transform seems to be a correct tool to separate high frequencies to low frequencies. Due this property, The DCT should also be an efficient line projection decorrelation tool. An other way to decorrelate is described later by using Principal Component Analysis on line projection.

Given an $x(n)$ array, the DCT transformation is:

$$X(k) = w(k).\sum_{n=0}^{N-1}\left(x(n).\cos\frac{\pi.(2n+1).k}{2N}\right) \tag{2.26}$$

with

$$w(k) = \begin{cases} \frac{1}{\sqrt{N}} & k = 0 \\ \sqrt{\frac{2}{N}} & 1 \leq k \leq N \end{cases} \tag{2.27}$$

Most of the signal energy is gathered in the low frequencies. The 25 first coefficients provide 65 percent of the energy. We can use this particularity to compute an efficient way to compress and detect with the help of the other coefficient provided with the candidate RASH digest. We store only a part of the whole signal.
The complete visual soft hash algorithm based on middle point projection is divided in five steps:

Step 1 : Energy
The square root (for dynamic range considerations) of the mean energy is computed and quantified to reduce the dynamic range of the signal. During this step, we can fix and store the scale factor given by the signal energy.

Step 2 : Mean reduction
The first DCT coeficient is not necessary, only the signal shape is interesting to recognize and detect a correct signature.

Step 3 : Discret Cosinus Transform
We compute the DCT coefficients 1 to n+1 from the normalized rash array, where n is a parameter depending on the length desired of the message digest.

Step 4 : Normalization
If the minus value of the transformed coefficients is negative we add its absolute value to all the coefficients in order to have a non-negative array and work with unsigned values, saving space. The absolute value must be kept for reconstruction. To keep the DCT array in the desired dynamic range for quantifying with one byte per coefficient, we normalize all the array values by the maximum DCT coefficient.

Step 5 : Quantization
In order to minimize the lossy quantization, casting the float to integer, we multiply all values by a quantization step depending on all dynamic range. This value is also stored to reverse the process for reconstruction. All float values are converted into integers.

The following figure fig.2.17 shows the different steps to compute a visual hash for Lena.

2.4.3 Detection

The authentication and detection are divided in two steps.

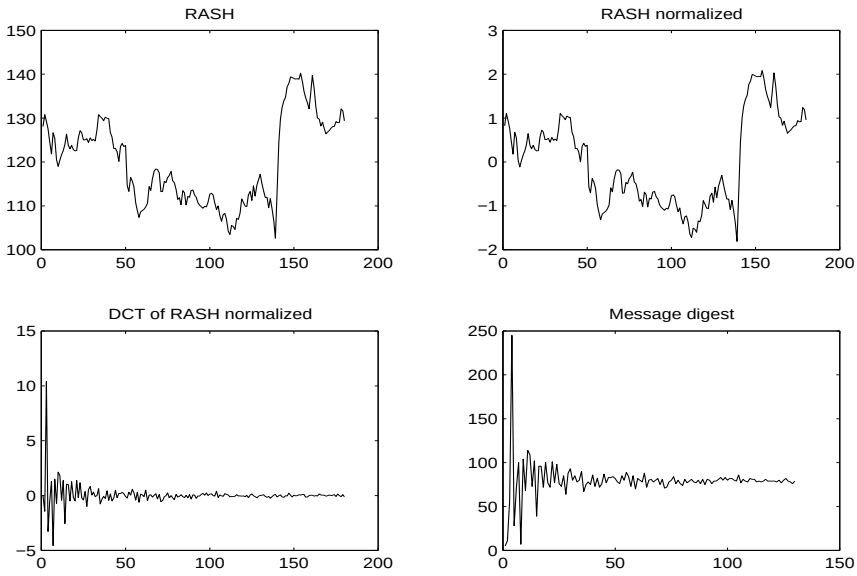

Figure 2.17: *RASH compression steps for a message digest of 1024 bits*

Firstly, we determine if the 25 first coefficients of the original DCT signature $X(k)$ and 25 first coefficients of the candidate DCT signature $X'(k)$ are computed from the same image. During this first step, called detection, we pad the candidate DCT coefficients by zero.

Secondly, if the cross correlation between the two inverse DCT sequences gives a peak over 0.95 (found experimentaly), we synchronize the two signatures by shifting the candidate signature sequence, and we compute the second step, called authentication.

Step 1: Detection
The goal of the first step is to decrease the false alarm rate by correlation estimation: we evaluate if the candidate DCT sequence matches with the original signature. The final detection result is given by the step 2. A false alarm or a false detection can occur during this first step. We compute the message digest algorithm on the candidate image until the step 3. We keep the first 25 DCT coefficient and padd the two sequences (candidate and original) by zeros to constitute a 180-element array for each sequence. Based on those few coefficients sequence, we inverse the Discrete Cosinus Transform. Those two new sequences, $x_e(n)$ and $x'_e(n)$, give a smooth

curve of the signature:

$$x_e(n) = \sum_{k=0}^{N} \left(w(k).X(k). \cos \frac{\pi.(2n+1).k}{2N} \right) \tag{2.28}$$

$$x'_e(n) = \sum_{k=0}^{N} \left(w(k).X'(k). \cos \frac{\pi.(2n+1).k}{2N} \right) \tag{2.29}$$

with

$$X(k) = \begin{cases} X(k) & 1 \leq k \leq 25 \\ 0 & k = 0 \; and \; 26 \leq k \leq N-1 \end{cases} \tag{2.30}$$

and

$$X'(k) = \begin{cases} X'(k) & 0 \leq k \leq 25 \\ 0 & k = 0 \;\; 26 \leq k \leq N-1 \end{cases} \tag{2.31}$$

The peak of cross correlation between $x_e(n)$ and $x'_e(n)$ gives the degree of the rotation and the likelihood level of the two sequences, $x_e(n)$ and $x'_e(n)$. If a peak is detected over 0.95, the candidate sequence is shifted until a perfect synchronization with the original signature and the step 2 is computed.

Step 2 : Authentication
We padd the 25 first DCT of message digest from the original signature with the 154 last DCT coefficients of the candidate signature. Now, we have:

$$x_e(n) = \sum_{k=0}^{N} \left(w(k).X(k). \cos \frac{\pi.(2n+1).k}{2N} \right) \tag{2.32}$$

$$x'_e(n) = \sum_{k=0}^{N} \left(w(k).X'(k). \cos \frac{\pi.(2n+1).k}{2N} \right) \tag{2.33}$$

with

$$X(k) = \begin{cases} X(k) & 1 \leq k \leq 25 \\ X'(k) & k = 0 \;\; 26 \leq k \leq N-1 \end{cases} \tag{2.34}$$

The Discret Cosinus Transform is a linear transform and each coefficient depends on each element of the input signal. And the inverse transform

respects this last property. If an inverse transform is computed using two different sequences, the output signal will be different to an output built with the same sequence. If signature and DCT coefficients padded do not come from the same signal, the reconstruction will be wrong, and too many differences will exist between the iDCT (inverse Discrete Cosinus transform) from the original message digest and the candidate signature. The table (2.18) gives us the detection result for a small image database.

	without synch		with synch	
Ref:Lena	$max(R_{xy})$	MSE	$max(R_{xy})$	MSE
Lena-rot(-5)	0.99	0.21	0.99	$2.3.10^{-3}$
Lena-scale(0.8)	0.99	10^{-3}	0.99	10^{-3}
Lena-sharpen*2	0.99	$1.76.10^{-3}$	0.99	$1.76.10^{-3}$
Lena-blur*2	0.99	$7.1.10^{-4}$	0.99	$7.1.10^{-4}$
Lena-stirmark	0.99	$4.3.10^{-3}$	0.99	$4.4.10^{-3}$
baboon	0.6	0.76	0.6	0.74
barbara	0.67	0.81	0.68	0.62
cat	0.61	0.96	0.59	0.82
fishingboat	0.45	1.17	0.84	0.39
fruits	0.53	0.93	0.53	0.93
girl	0.58	3.21	0.5	1.04
goldhill	0.86	0.28	0.86	0.27
houses	0.82	0.38	0.82	0.35
peppers	0.61	0.77	0.61	0.77
pool	0.54	2.97	0.55	0.97
watch	0.44	2.56	0.54	1.37

Figure 2.18: *collision and authentication tests*

The results confirm results from previous work. Over 0.95 peak cross-correlation, the candidate signature and the original signature contain the same image contents. The detection is efficient for all attacked Lena images.

2.5 The theoretical threshold computation

The goal of this study is to evaluate the statistical rate of collision. It means that we want to calculate the false detection rate, the number of images identified as an original image of the database instead of identified as a

corrupted image. In the first part, the main idea is to obtain the lowest collsion rate. In the second section, we calculate the best theoretical threshold based on cross-correlation between two image signatures.

2.5.1 Working hypothesis

If there is only one image in the database, we called X_i the random variable (RV) such as:

$$X_i = \begin{cases} 1 & \text{if there is a collision} \\ 0 & \text{otherwise} \end{cases} .$$

Regarding Bernouilli law [11] (success or failure of an event A in n independant trials): We are given an experiment *S* and an event *A* with:

$$\begin{aligned} P(A) &= p \\ P(\overline{A}) &= q \\ p+q &= 1 \end{aligned}$$

We repeat the experiments n times with independant trials, we shall determine the probability $p_n(k)$ that the event *A* occurs exactly k times and the fundamental theorem is:

$$p_n(k) = P\left\{A \text{ occurs } k \text{ times in any order}\right\} \tag{2.35}$$

$$p_n(k) = \binom{k}{n}.p^k.q^{n-k} \tag{2.36}$$

In our case, if we call $p = P(X_i = 1)$ and $q = P(X_i = 0) = 1 - p$, X_i is according to Bernouilli law with the parameter p. If we call M, the collision mean over a sample of n images is:

$$M = \frac{1}{n}\sum_i x_i \tag{2.37}$$

2.5.2 Estimator model

We need to estimate and find the probability p - defined as the probability to have a collision between a signature test with original signature among the database - thanks to N independant observations:

$$\begin{aligned} X &= x_1 \text{ at the } first \text{ observation} \\ &= x_2 \text{ at the } second \text{ observation} \\ &= x_i \text{ at the } ith \text{ observation} \\ &= x_N \text{ at the } last \text{ observation} \end{aligned}$$

Regarding probability distribution, the X RV law is also called *Binomial Law* .

For a such law, the mean of the N RV X is Mb:

$$Mb = \frac{1}{N}\sum_{1}^{N} X_i \tag{2.38}$$

The mean $E(X)$ of the RV X is:

$$\begin{aligned} E(X) &= E(X_1 + X_2 + \cdots + X_n) \\ &= E(X_1) + E(X_2) + \cdots + E(X_n) \\ &= n.E(X_i) = n.p \end{aligned} \tag{2.39}$$

The variance $Var(X)$ of the RV X is:

$$E(X) = n.Var(X_i) = n.p.q = n.p.(1-p) \tag{2.40}$$

If $T = Mb/n$ is the estimator with (X_i) the random sequence, the RV are independant and follow the same law. The estimator T tends to p and $converges$.

To estimate correctly if $T = Mb/n$ is a correct estimator, we need to evaluate if it is biased or not. The bias is computed by $b(T) = E(T) - p$:

$$\begin{aligned} E(T) &= E(Mb/n) \\ &= \frac{1}{n}.E(\frac{1}{N}.\sum_{1}^{N} X_i) \\ &= \frac{1}{n}.\frac{1}{N}.E(\sum_{1}^{N} X_i) \\ &= \frac{1}{n}.E(X) \\ &= p \end{aligned} \tag{2.41}$$

So, $b(T) = E(T) - p = 0$ and the estimator $T = Mb/n$ is unbiased.

2.5.3 Estimator Efficiency

As defined in [11], we have an RV **x** with density $f(x,\theta)$, and we wish to estimate θ in terms of a single observation of RV **x**. To do so, we plot the density $f(x,\theta)$ as a function of θ, assigning to x the observed value of **x**, and we determine the value $\hat{\theta} = \theta_{max}$ of θ that maximizes $f(x,\theta)$. We shall call the curve $f(x,\theta)$ so plotted the $likelihood\ function$ of **x** and the number $\hat{\theta}$ the $maximum\ likelihood$ (ML) estimate of θ. This estimate is the value of θ for which the probability $f(x,\theta)dx$ that the RV **x** is in the interval $(x, x+dx)$ is maximum.

We shall now determine the ML estimate of θ in terms of n observations x_i of **x**. To do so, we form the joint density:

$$f(X,\theta) = f(x_1,\theta)\cdots f(x_n,\theta) \tag{2.42}$$

of n samples x_i of **x**. This density considered as a function of θ is called the $likelihood\ function$ of **X**. the value $\hat{\theta}$ of θ that maximizes $f(X|\theta)$is the ML estimate of θ. The logarithm

$$L(X,\theta) = ln\ f(X,\theta) = \sum_{i=1}^{n} ln\ f(x_i,\theta) \tag{2.43}$$

is the $log-likelihood$ function of **X**.

If the density $f(x,\theta)$ of **x** is differentiable with respect to θ and the boundary of the domain of x does not depend on θ, according to $Cramer-Rao$, an estimator is the best or that it is closed to the best if the variance $E\{[\hat{\theta}-\theta]^2\}$of any unbiased estimator $\hat{\theta}$ cannot be smaller than $1/nI$.

$$\sigma_{\hat{\theta}}^2 \geq \frac{1}{n.I} \tag{2.44}$$

$I(\theta)$ is called Fisher information and its value is:

$$I(\theta) = E_\theta[\frac{\partial}{\partial\theta}.ln\ f(x,\theta)]^2 \tag{2.45}$$

$$I(\theta) = -E_\theta[\frac{\partial^2}{\partial\theta^2}.ln\ f(x,\theta)] \tag{2.46}$$

If $X_1,...,X_n$ are independant then $\frac{\partial}{\partial\theta}.ln\ f(x,\theta$ are also independant and:

$$Var(\sum_{i=1}^{n}\frac{\partial}{\partial\theta}.ln\ f(X,\theta) = \sum_{i=1}^{n} Var(\frac{\partial}{\partial\theta}.ln\ f(X_i,\theta) \tag{2.47}$$

$I(\theta)$ is the information for one observation and $I_n(\theta)$ is the information for n observations. $1/I_n(\theta)$ is the $Cramer - Rao$ bound. $I_n(\theta)$ is the Fisher information for n observations.
So, $I_n(\theta)$ can be expressed as fellow:

$$I_n(\theta) = Var[\frac{\partial}{\partial\theta}.ln\ f(X,\theta)] \tag{2.48}$$

In our case , the law is following the Binomial law and its density function is:

$$f(x|p) = \binom{x}{n}.p^x.q^{n-x} \tag{2.49}$$

Now, we compute the Fisher information for this binomial law:

$$\begin{aligned} \frac{\partial}{\partial p}log\ f(x|p) &= \frac{x}{p} - \frac{n-x}{1-p} \\ &= \frac{x(1-p)-(n-x)p}{p(1-p)} \\ &= \frac{x-np}{p(1-p)} \end{aligned} \tag{2.50}$$

$$\begin{aligned} I(p) &= Var(\frac{x-np}{p(1-p)}) \\ &= \frac{np(1-p}{p^2(1-p)^2} \\ &= \frac{n}{p(1-p)} \end{aligned} \tag{2.51}$$

An other demonstration can be done. In our case the experiments are discret and the Fisher information can calculated by:

$$I(p) = -E(\frac{\partial^2}{\partial p^2}ln\ P(X=x)) \tag{2.52}$$

So,

$$\frac{\partial}{\partial p}ln\ P(X=x) = \frac{X}{p} + \frac{n-X}{1-p} \tag{2.53}$$

$$\frac{\partial^2}{\partial p^2}ln\ P(X=x) = -\frac{X}{p^2} - \frac{n-X}{(1-p)^2} \tag{2.54}$$

$$\begin{aligned} E(\frac{\partial^2}{\partial p^2} ln\ P(X=x)) &= E(-\frac{X}{p^2}) - E(\frac{n-X}{(1-p)^2}) \\ &= -\frac{n}{p} - \frac{np}{(1-p)^2} + \frac{n}{(1-p)^2} \\ &= -\frac{n}{p(1-p)} \end{aligned} \tag{2.55}$$

$$\begin{aligned} I(p) &= -E(\frac{\partial^2}{\partial p^2} ln\ P(X=x)) \\ &= \frac{n}{p(1-p)} \end{aligned} \tag{2.56}$$

For N observations, the $Cramer - Rao$ bound is:

$$\frac{1}{N.I(p)} = \frac{p(1-p)}{nN} \tag{2.57}$$

Our estimator $T = Mb/n$ is efficient if

$$Var(T) \geq \frac{1}{N.I(p)} \tag{2.58}$$

The variance estimator is:

$$Var(T) = Var(\frac{Mb}{n}) \tag{2.59}$$

$$= \frac{1}{n^2} Var(X) \tag{2.60}$$

$$= \frac{p(1-p)}{nN} \tag{2.61}$$

So, $Var(T) = \frac{1}{N.I}$ and T is the best unbiased estimator and it is efficient.

2.5.4 Confidence interval

We found the best estimator, $T = Mb/n$ with $Mb = \frac{1}{N}\sum X_i$. We want to obtain the lowest collision rate. In this part, we study the rate of signature collision based on estimator model $T = Mb/n$.

If we called $x_1, x_2, ...x_N$ the mesurements and $X_1, X_2, ..., X_N$ the random variables, p can be estimated by $mb = \frac{1}{N}\sum x_i$.

If $Mb = \frac{1}{N}\sum X_i$ due to Central Limit Theorem (CLT), the RV $N.Mb$ can be approximated by a $Normal\ law\ N(Np, Np(1-p))$. According to [11], the statistical test can be approximated using the RV $\frac{N.Mb-N.p}{\sqrt{N.p.q}}$ and this RV is following the $Normal\ law\ N(0,1)$.

We search a confidence interval for p with the risk α (0.05 for exemple). We search an interval for RV U following $Normal\ law$ such as:

$$P(-u_\alpha \leq \frac{N.Mb - N.p}{\sqrt{N.p.q}} \leq u_\alpha) \approx 1-\alpha \tag{2.62}$$

$$P(p - u_\alpha.\sqrt{\frac{pq}{N}} \leq Mb \leq p + u_\alpha.\sqrt{\frac{pq}{N}}) \approx 1-\alpha \tag{2.63}$$

Applying this previous result on mesurements, we have:

$$P(p - u_\alpha.\sqrt{\frac{pq}{N}} \leq mb \leq p + u_\alpha.\sqrt{\frac{pq}{N}}) \approx 1-\alpha \tag{2.64}$$

With $mb = \frac{1}{N}\sum x_i$. x_i are Bernouilli RV, we can approximate $\sqrt{pq} = \sqrt{p(1-p)}$ by $\sqrt{mb(1-mb)}$. Hence, we obtain:

$$p \in [mb - u_\alpha.\sqrt{\frac{mb(1-mb)}{N}}, mb + u_\alpha.\sqrt{\frac{mb(1-mb)}{N}})] \tag{2.65}$$

With $mb = \frac{1}{N}\sum x_i$ and for a error risk α, u_α come from $Normal\ law\ table$ [11].

EXPERIMENTS

Regarding image size, and for a risk $\alpha = 0.05$, the experiments are as fellows. The database come from some frames extracted from TV video bit stream. The algorithm used to select frames in a video will be explained later in next chapter.

For all above tests, the $bound\ min$ is 0 and the risk $\alpha = 5\%$ for a mistake, we have:

- for image size=320x240, less than 1.4 false detection for 1 million of images
- for image size=480x360, less than 6.3 false detection for 1 million of images

Image size	#images	#collisions	bound max (10E-5)
320x240	181332	1	1.327
480x360	229761	10	6.289
640x480	125145	3	4.345

Figure 2.19: *Interval confidence applied on TV frames*

- for image size=640x480, less than 4.4 false detection for 1 million of images

The results are efficient, and the image size seems to be not sensitive for signature detection. Appling this test in an other database from image database and not from video database, the model is less efficient. This result is due to in case of texture. A signature from database provide 9 collisions between signature from image database and signature from TV capture. The signature computation does not take in consideration the image texture. The summation along a certain direction affects image representation and does not take enought information to caracterize the picture. The next chapter provides a solution to image description and image recognition.

2.5.5 Theoretical optimal threshold

Firstly, we calculate the empirical threshold using a training USC-SIPI database [13]. Secondly, we determine the theoretical threshold for a correct detection between two signatures.

EMPIRICAL THRESHOLD
For each image, we performed a series of 8 image processing attacks:

- **Filtering**: 3x3 Gaussian filtering with standard deviation of 0.5 and 3x3 averaging filtering.
- **Compression**: JPEG compression 25%, JPEG compression 15%,
- **Geometric**: scaling with scaling factor α = 1.2 and 0.8, 2 degree rotation and 1 degree rotation with cropping.

Cross-correlation is the metric used for matching detection between two message digests. Hence 320 matchings are done between images with

the same content, we call them intra-image matchings. For comparisons between images with different contents (inter-image matchings), we matched each image signature from the database against the 39 remaining image signatures.

Different statistical detection rate are analysed to evaluate the empirical threshold:

- The *false alarm* detection: two signatures are detected as differents signatures although they are identical.
- The *miss* detection: two signatures are detected as identical signatures although they are differents
- The *hit* detection: two identical signatures are detected as identical signatures although they are identical.
- The *correct rejection* detection: two different signatures are detected as differents.

For a given threshold, each statistical detection rate is computed fig.2.20. The Equal Error Rate (EER) is usually choosen as empirical threshold, its value is 0.95. A best threshold given by the same $falsealarm$ rate with 0 $missdetection$ is 0.89.

THEORETICAL THRESHOLD

Previous experiments bring to the force an empirical threshold of 0.95. The context and experiences higlight three types of detection and probability:

- p_e = probability of false detection
- p_1 = probability of no collision
- p_0 = probability of correct and justified collision

The data and sample are large, we can estimate that the density probablity functions are $normal$.

If, f_0 is the density function of the probability p_0 with a mean μ_0 and the variance σ_0:

$$f_0(y) = \frac{1}{\sqrt{2\pi\sigma_0^2}} \exp - \frac{(y-\mu_0)^2}{2\sigma_0^2} \tag{2.66}$$

If, f_1 is the density function of the probability p_1 with a mean μ_1 and the variance σ_1:

$$f_1(y) = \frac{1}{\sqrt{2\pi\sigma_1^2}} \exp - \frac{(y-\mu_1)^2}{2\sigma_1^2} \tag{2.67}$$

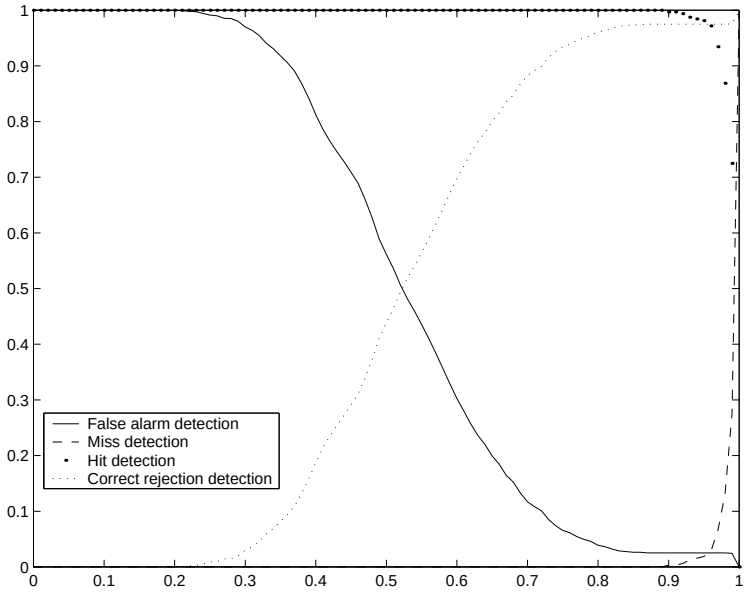

Figure 2.20: $Falsealarm$ *detection and* $miss$ *detection regarding the threshold*

If a is the correct value of the RV X and $\hat{a}$ the measurement value of the RV X at the moment k, the false detection probability p_e is:

$$p_e = p_1.P(\hat{a} = 0/a = 1) + p_0.P(\hat{a} = 1/a = 0) \tag{2.68}$$

The main idea is to minimize the probability p_e.
If T is the optimal threshold The conditionnal probabilities are :

$$P(\hat{a} = 0/a = 1) = \int_{-\infty}^{T} f_1(y)dy \tag{2.69}$$

$$P(\hat{a} = 1/a = 0) = \int_{T}^{\infty} f_0(y)dy \tag{2.70}$$

So, we have:

$$p_e = p_1.\int_{-\infty}^{T} \frac{1}{\sqrt{2\pi\sigma_1^2}} \exp -\frac{(y-\mu_1)^2}{2\sigma_1^2}dy + p_0.\int_{T}^{\infty} \frac{1}{\sqrt{2\pi\sigma_0^2}} \exp -\frac{(y-\mu_0)^2}{2\sigma_0^2}dy \tag{2.71}$$

Using $u = \frac{y-\mu_0}{\sqrt{2\sigma_0^2}}$, $v = \frac{y-\mu_1}{\sqrt{2\sigma_1^2}}$ and $erfc = \frac{2}{\sqrt{\pi}} \int \exp -u^2 du$, we obtain:

$$p_e = \frac{p_0}{2} erfc(\frac{\mu_0 + T}{\sqrt{2\pi\sigma_0^2}}) + \frac{p_1}{2} erfc(\frac{\mu_1 - T}{\sqrt{2\pi\sigma_1^2}}) \tag{2.72}$$

Find the optimal threshold is to minimize the probability of false detection p_e. Minimize the probability of false detection p_e is to solve $\frac{dPe}{dT} = 0$:

$$\frac{\partial Pe}{\partial T} = p_1 . \exp(-\frac{(\mu_1 - T)^2}{2\sigma_1^2}) - p_0 . \exp(-\frac{(\mu_0 + T)^2}{2\sigma_0^2}) \tag{2.73}$$

$$\frac{\partial Pe}{\partial T} = 0 \Leftrightarrow ln(\frac{p_1}{p_0}) = -\frac{(\mu_0 + T)^2}{2\sigma_0^2} + \frac{(\mu_1 - T)^2}{2\sigma_1^2} \tag{2.74}$$

$$\frac{\partial Pe}{\partial T} = 0 \Leftrightarrow T^2 . a + T . b + c = 0 \tag{2.75}$$

with

$$\begin{cases} a = 2\sigma_1^2 - 2\sigma_0^2 \\ b = 4\sigma_0^2\mu_1 + 4\sigma_1^2\mu_0 \\ c = ln(\frac{p_1}{p_0}).(4\sigma_0^2\sigma_1^2) - 2\sigma_0^2\mu_1^2 + 2\sigma_1^2\mu_0^2 \end{cases} .$$

So, with $T \geq 0$, the optimal threshold is $T = \frac{-b+\sqrt{b^2-4ac}}{2a}$.

EXPERIMENTS

As before, the database come from some frames extracted from TV video bit stream. The algorithm used to select frames in a video will be explained later in next chapter. The experiment is tested in 72 hours of TV programs (6480000 frames), and only 229761 are selected and compared to a database of 7 images. 13 image signatures are correctly detected.

If $IdCorr_i$ is the set of correlation between two identical signatures (images with the same content):

$$p_1 = \text{probability of correct detection for two identical signatures}$$

$$\mu_1 \approx m_1 = \frac{1}{n} \sum IdCorr_i \tag{2.76}$$

$$\sigma_1^2 \approx s_1^2 = \frac{1}{n} \sum (IdCorr_i - \mu_1)^2 \tag{2.77}$$

If $DiffCorr_i$ is the set of correlation between two different signatures (images with different content):

$$\begin{aligned} p_0 &= \text{probability of correct detection for two different signatures} \\ \mu_0 &\approx m_0 = \frac{1}{n}\sum DiffCorr_i \quad (2.78) \\ \sigma_0^2 &\approx s_0^2 = \frac{1}{n}\sum (DiffCorr_i - \mu_0)^2 \quad (2.79) \end{aligned}$$

With n the number of correlations ($n =' Number\ of\ images'$ x $'Number\ of\ signatures\ in\ the\ database'$).

m_0	s_0^2	p_0	m_1	s_1^2	p_1
0.67964	0.008062	0.999943	0.99482	5.1E-0.5	5.66E-0.5

Figure 2.21: *Optimal threshold T computation*

The theoretical optimal threshold is 0.87 and it is closed to empirical threshold 0.96. The difference between the two thresholds is perhaps due to the number of samples in the reference database. With more samples the result should be better.

2.6 Radon Transform and Principal Component Analysis

The previous work [1, 14] detailed in the previous section describes a new hash function based on summation of radial projection. In this section, we use a new features extraction based on the Radon transform and Principal Component Analysis to increase the robustness against geometrical transformation (rotation and scaling) and image processing attacks (compression, filtering, blurring).

2.6.1 Description

In the previous section, we applied the Radon transform principle to digital images. Given an image, the luminance of image pixels $g(i, j)$ are summed up along a set of directions fig.(2.3). This operation is repeated for 180 directions uniformally distributed on a half circle and defines 180 projections of the image. Formally, for $\theta = 0, 1, ...179$, we compute 180

projections

$$p_i(\theta) = \frac{1}{N_{i\theta}} \sum_j g(i\cos\theta - j\sin\theta, i\sin\theta + j\cos\theta)$$

where $N_{i\theta}$ is the pixel number along direction θ. The projection $p_\theta(i)$ is therefore the average luminance of the image in direction θ. The purpose of the normalisation is to keep the magnitude of p_θ between 0 and 255. The image content is better described by the *variation* of the projections rather than the projection themselves, which depend on the average luminance value of the image. To achieve robustness against average luminance changes, we use the projection angular increment $w_\theta(i) = p_\theta(i) - p_{\theta-1}(i)$ to generate the image signature. We introduce a set of N 180-dimensional vectors $v_i(\theta)$, that we call Radon vectors, by taking the ith value of of the angular increment $w_\theta(i)$ for the 180 directions. The number N depends on the size of the image. For a square image with size n, we have $N = \lfloor\sqrt{2}n\rfloor$. Although the two properties cited above are not valid for discrete functions, a good approximation of the Radon transform of rotated and scaled images can be found using a discrete version of equations (2.7) and (2.8). Let v_i^ϕ and v_i^α correspond to Radon vectors of an image rotated by ϕ and scaled by a factor α respectively. It can be shown that

$$v_i^\phi(\theta) \approx v_i(\theta + \phi) \text{ for } \theta + \phi \leq N \tag{2.80}$$

$$v_i^\phi(\theta) \approx v_i(N + \theta - \phi) \text{ for } \theta + \phi > N. \tag{2.81}$$

In other words, the Radon vectors undergo a cyclic shift during a rotation.

In order to fulfill the digital signature requirements cited in previous section, we must extract a short and fixed length bit string from the Radon vectors of an image, that characterises as well as possible this image. To achieve this, we extract two vectors from the Radon vectors using a method inspired by Principal Component Analysis (PCA) [12]. PCA tools were developed in collaboration with Jacek Czyz.

Principal Component Analysis (PCA)

This PCA explanations come from Jacek Czyz thesis [15].

PCA is a standard technique in data analysis which is used for dimensionality reduction or equivalently for feature extraction for signal representation. Given a set of l data vectors $\mathbf{x}_i \in \mathbb{R}^n$ which are instances of a random vector $\mathbf{x}$, PCA looks for $m \leq n$ orthonormal vectors $\{\boldsymbol{\phi}_j\} \in \mathbb{R}^n$ which

form an orthonormal basis of the subspace that captures maximal variance of the $\mathbf{x}_i$'s. It can be shown [12] that the $\{\boldsymbol{\phi}_j\}$'s are the eigenvectors of the sample covariance matrix Σ of the $\mathbf{x}_i$'s

$$\Sigma = \frac{1}{l}\sum_{i=1}^{L}(\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})^T, \tag{2.82}$$

where $\bar{\mathbf{x}}$ is the sample mean of the $\mathbf{x}_i$'s. Let λ_j be the eigenvalue associated with the eigenvector $\boldsymbol{\phi}_j$ we have

$$\Sigma\boldsymbol{\phi}_j = \lambda_j\boldsymbol{\phi}_j.$$

The approximation or *reconstruction* $\hat{\mathbf{x}}$ of the random vector $\mathbf{x}$ is the component of $\mathbf{x}$ that lies in the subspace expressed in $\mathbb{R}^n$ i.e.

$$\hat{\mathbf{x}} = \bar{\mathbf{x}} + \sum_{j=1}^{m}(y_j - \bar{y}_j)\boldsymbol{\phi}_j,$$

where

$$y_i = \boldsymbol{\phi}_i^T\mathbf{x}$$

is called the ith principal component of $\mathbf{x}$ and $\bar{y}_i$ is the ith principal component of the mean $\bar{\mathbf{x}}$, i.e.

$$\bar{y}_i = \boldsymbol{\phi}_i^T\bar{\mathbf{x}}.$$

By defining the $n \times m$ matrix Φ whose columns are m eigenvectors $\boldsymbol{\phi}_j$ of Σ we can write the principal component vector $\mathbf{y} \in \mathbb{R}^m$

$$\mathbf{y} = \Phi^T\mathbf{x},$$

and because the columns of Φ are orthonormal, we have

$$\hat{\mathbf{x}} = \bar{\mathbf{x}} + \Phi(\mathbf{y} - \bar{\mathbf{y}}),$$

where $\bar{\mathbf{y}} = \Phi^T\bar{\mathbf{x}}$. A very nice property of PCA is that the mean square error ϵ^2 between $\mathbf{x}$ and its reconstruction, which is

$$\epsilon^2 = E[\|\mathbf{x} - \hat{\mathbf{x}}\|^2],$$

can be written as

$$\epsilon^2 = \sum_{j=m+1}^{n}\tilde{\lambda}_j,$$

where $\tilde{\lambda}_j$ are the eigenvalues of the true (unknown) covariance matrix generating the $\mathbf{x}_i$'s. This last equation suggests that the mean square error between $\mathbf{x}_i$ and its reconstruction $\hat{\mathbf{x}}_i$ is minimised if the subspace basis contains the m eigenvectors $\boldsymbol{\phi}_j$ with the m highest eigenvalues. One must keep in mind that the transform Φ is the optimal *linear* transform under a mean square error criterion. This means that there may exist other non-linear transforms which leads to a better representation of the signal.
The PCA decorrelates the $\mathbf{x}_i$'s as the covariance expressed in the principal subspace

$$\Sigma_{\mathbf{y}} = \Phi^T \Sigma \Phi \tag{2.83}$$

is a $m \times m$ diagonal matrix whose diagonal elements are the λ_j's.

PCA and Radon projections

Here, we have N vectors that we want to characterise using a small set of numbers. From the N Radon vectors v_i corresponding to an image, we estimate the covariance matrix of the v_i by

$$\Sigma = \sum_{i}^{N} (v_i - \mu)(v_i - \mu)^T \tag{2.84}$$

where μ is the v_i mean, i.e. $\mu = \frac{1}{N}\sum_i^N v_i$. Due the small cardiality expected for the message digest, we extract the eigenvectors of Σ corresponding to the only two largest eigenvalues, and these vectors form the digital signature of the image fig. 2.22.
This process has a geometrical interpretation: the set of N Radon vectors that characterises a given image can be seen as a set of N points, forming a cloud in a 180-dimensional space. The eigenvector with maximum eigenvalue of the covariance matrix corresponds to the direction where the cloud has maximum variance. This direction is therefore a global statistical property of the points that will be little affected by small changes in the points resulting from small changes in the image. In contrast, the global configuration of the points will change when the image content is different, hence the direction with largest variance will change as well.

If the image is rotated, its Radon vectors are cyclically shifted. It can be shown that the eigenvectors of the Radon vector covariance matrix are shifted in the same way.

If the image is scaled, the same signal is resampled more densely or more sparsely depending on whether the size increases or decreases. The same

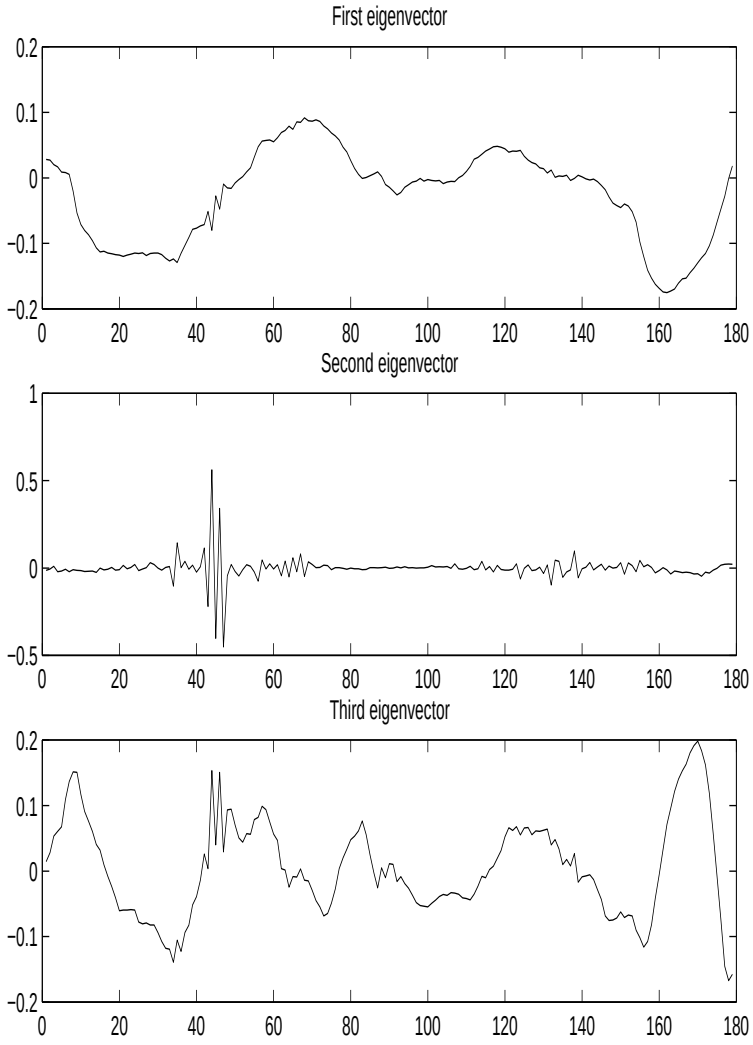

Figure 2.22: *The first eigenvectors of PCA applyed on Radon Transform*

resampling happens for the Radon vectors. In fact, the amount of points forming the cloud changes but its global configuration remains the same, leading hence to the same direction with largest variance as the original image.
When two signatures x and y of two images have to be matched in order to determine whether the two images have the same or different contents,

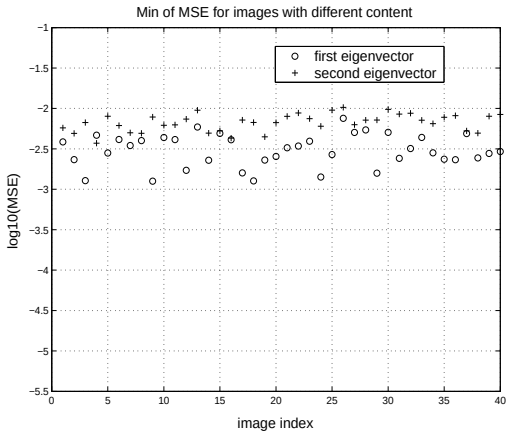

Figure 2.23: *Minimum MSE for signature matching between images with different content.*

we compute the cross-correlation R_{xy} between the two signatures

$$R_{xy}(m) = \sum_{n=0}^{d-m-1} \left(x_n - \frac{1}{d}.\sum_{i=0}^{d-1} x_i \right). \left(y_{n+m} - \frac{1}{d}.\sum_{i=0}^{d-1} y_i \right)$$

where d is the length of the signature. Since cross-correlation compares the two signals at different values of shifting, when the two signatures come from images with the same content, $R_{xy}(m)$ will be close to 1 for a certain m^*. In fact, m^* corresponds to the angle between the images in the case of two rotated version of the same images. In our implementation, the two signatures are re-synchronised using m^* and the Mean Square Error (MSE) between them is computed using

$$MSE = \frac{\sum_{i=0}^{d-1} (x_i - y_i)^2}{d}. \tag{2.85}$$

The MSE determines if the signatures come from images with the same content.

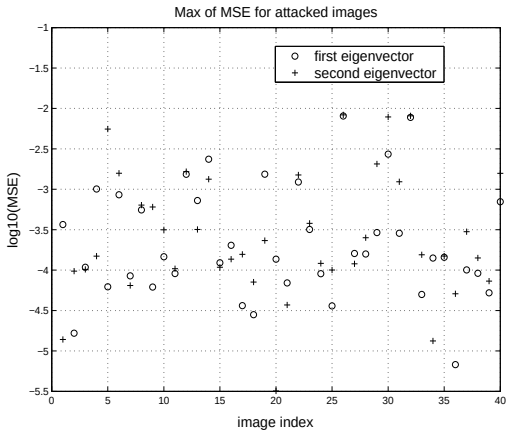

Figure 2.24: *Maximum MSE for signature matchings between images with the same content.*

2.6.2 Experiments

In order to evaluate the robustness and the collision resistance of the proposed algorithm, we performed experiments on real images taken from the USC-SIPI database [13]. The USC-SIPI image database is a collection of digitised images which are free of copyrights if used in image processing research. The miscellaneous sub-set consists of 40 images like baboon, Lena and peppers, of various sizes such as 256x256 pixels, 512x512 pixels, or 1024x1024 pixels. All colour images are transformed into 8 bits/pixel gray level images.
For each image, we performed a series of 8 image processing attacks:

- **Filtering**: 3x3 Gaussian filtering with standard deviation of 0.5 and 3x3 averaging filtering.
- **Compression**: JPEG compression 25%, JPEG compression 15%,
- **Geometric**: scaling with scaling factor α = 1.2 and 0.8, 2 degree rotation and 1 degree rotation with cropping.

Cross-correlation and MSE are computed (after re-synchro-nisation) between the original and the modified image signatures. Hence 320 matchings are done between images with the same content, we call them intra-image matchings. For comparisons between images with different contents (inter-image matchings), we matched each image signature from the

database against the 39 remaining image signatures. This gives 40x39/2 = 780 inter-image matchings. For the 40 images in the database, figure 2.23 shows the minimum MSE from these 39 matchings, that is, the MSE between the two closest signatures when the image contents are different. Circles are plotted when the first eigenvector is used as signature, while crosses are plotted when the second eigenvector is used. Figure 2.24 shows the maximum MSE for each image signature matched with the 8 corresponding attacked image signatures. Hence only $\frac{1}{8}$ of the intra-image matchings is shown. From this figure, it appears most of the MSE's are below 10^{-3} for intra-image matchings, only 8 attacked images (on 320) lead to MSE's greater than 10^{-3}. For inter-image matchings there is no MSE under 10^{-3}. Using the MSE, we can therefore determine with a certain confidence if two signatures come from the same image or from two different images. It appears that the images leading to high intra-image MSE contain a lot of high frequency textures. This suggests that the signature is not well adapted to these kind of images. It is likely that the first eigenvector characterising the Radon vectors is not well defined in case of texture.

2.6.3 Conclusions

In the algorithm development, care has been taken for robustness to rotation and scaling by designing a method based on Radon transform and Principal Component Analysis. Our experimental results show that the digital signature is quite robust to popular image processing attacks, such as JPEG compression. Future work will be devoted to study resistance to other attacks, like stirmark [8] and to increase robustness for texture images.

Bibliography

[1] F. LEFEBVRE, B. MACQ and J.-D. LEGAT,"RASH:RAdon Soft Hash algorithm",EUSIPCO2002

[2] O. GOLDREICH "Two remarks concerning the Goldwasser-Micali-Rivest signature scheme." Technical Report MIT/LCS/ TM-315, MIT Laboratory for Computer Science, September 1986.

[3] R. RIVEST "RFC 1321: The MD5 Message-Digest Algorithm." RSA Data Security Inc., April 1992.

[4] National Institute of Standards and Technology (NIST), "Announcement of Weakness in the Secure Hash Standard", 1994.

[5] National Institute of Standards and Technology,"Digital Siganture Standard (DSS)", FIPS PUB 186-2

[6] J. FRIDRICH and M. GOLJAN, "Robust Hash Functions for Digital Watermarking", ITCC 2000, Las Vegas, March 27-29, 2000, Nevada, USA

[7] C.L. SABHARWAL and S.K. BHATIA. "Perfect Hash Table Algorithm for Image Databases Using Negative Associated Values." Pattern Recognition. 28:7. pp. 1091-1101. July 1995.

[8] F.A.P. PETITCOLAS, R.J. ANDERSON and M.G. KUHN, "Attacks on copyright marking systems", in Information Hiding:2nd Workshop, vol.1525, D. Aucsmith, Ed. Berlin, Germany:Springer-Verlag, 1998.

[9] Z.-P. LIANG and P.C. LAUTERBUR, "Principles of Magnetic Resonance Imaging, A Signal Processing Perspective", IEEE Press Series in Biomedical Engineering, Metin Akay, Series Editor.

[10] M.L. Brady, W. Yong, "Fast Parallel Discrete Approximation Algorithms for the Radon Transform." *SPAA*, 1992,pp.91-99.

[11] A. Papoulis "Probability, Random Variables, and Stochastic Process", Third edition, Mc Graw Hill.

[12] K. Fukunaga,"An Introduction to Statistical Pattern Recognition", Academic Press, 1990

[13] USC-SIPI image database, available at http://sipi.usc.edu/services/database/Database.html

[14] F. Lefebvre, B. Macq, "AGADDIS: Authentication and Geometrical Attacks Detection for Digital Image Signature", Information Theory 2002 Benelux,p171-178, Louvain La Neuve, Belgium

[15] J. Czyz, "Decision fusion in identity verification using facial images", thesis, december 2003, Universite catholique de Louvain.

3

A video digest based on the robust hashing of representative frames

3.1 Introduction

The purpose of *robust image hashing* is thus to define an image digest that satisfies two properties. First, similar to cryptographic message digest, the robust image digest characterizes the image in the sense that it uniquely identifies its content, i.e. the digests derived from a pair of visually distinct inputs have a low probability to be identical. Second, the hashing process is robust in the sense that the digest is only slightly affected when the image changes due to compression or minor processing, i.e. visually indistinguishable images generate equal or similar digests. Conversely to cryptographic hashing, robust hashing is thus able to deal with visually non-significant changes of the content, and supports common manipulations like compression or reformating (e.g. spatial or temporal subsampling).

Because it defines a vector that identifies the image content, robust hashing is an obvious solution for content identification and indexing. When used in combination with conventional cryptographic digital signature methods, robust hashing can also be used for integrity and authentication purposes [16, 24]. In watermarking, hashing enables the creation of payloads that depend on the media content, and which are thus resistant to the "copy attack" reported by Fridrich and al. in [25, 26].

The primary purpose of this chapter is the design of a robust image and video hashing algorithm. In Section 3.2 we propose to extract an image feature vector based on radial luminance projections, and validate our approach in terms of robustness, and discriminating nature of the extracted feature. In Section 3.3, the proposed image hashing technique is extended to video sequences. Representative frames are selected in the video sequence. The sequence digest corresponds to the set of image digests extracted from the set of representative frames.

A part of this chapter has be written in collaboration of Cedric De Roover and Christophe De Vleeschouwer.

3.2 Image robust hashing

This section defines and validates our proposed robust image hashing algorithm.

The proposed algorithm is defined in Section 3.2.1. It ensures different outputs for visually distinct input images, while providing similar outputs for visually equivalent contents. The design of our algorithm has

been focused on providing robustness towards specific geometric image transformations, i.e. rotation and scaling, and towards image processing attacks like blurring, sharpening, and compression.
Section 3.2.2 validates our method. It demonstrates that our proposed image digest performs better than a standard histogram-based digest, both in terms of robustness, and discriminating capabilities.

3.2.1 Robust image digest based on radial projections

The continuous Radon transform, and its discrete approximation are presented in previous section. There, we note that the continuous Radon transform is respectively dilated or translated due to input signal scaling or rotation. These properties result from the fact that the Radon transform is based on continuous angular projections of the image. They have inspired the design of our image hashing algorithm. Specifically, our algorithm computes the variance of the pixels luminance values along image lines projections. The projection lines go through the image center and are charaterized by their angular orientation. For this reason, we refer our algorithm as the Radial HASHing (RADISH) algorithm.

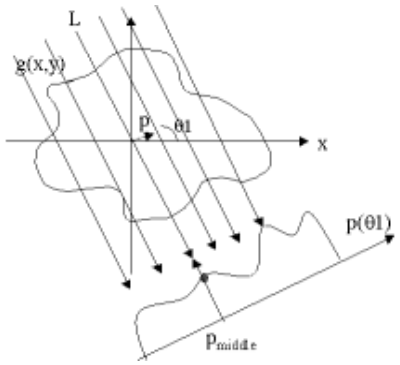

Figure 3.1: *Central point, and the corresponding lines of pixels used to compute the radial projections.*

Based on these guidelines, the spatial features extracted by the
RADISH algorithm have been defined on a 1-D support, as illustrated in Figure 3.1. According to RASH design, the components of the feature vector are computed on a set of lines articulated around the center of the image.

In practice, we have chosen to use a 180-sample feature vector, which corresponds to a uniformly distributed set of 180 angles ϕ, with $0 \leq \phi \leq 180$. For each projection angle, the feature sample is defined as the variance (instead of summation or PCA) of the pixels luminance values along the corresponding line.

The intuition behind the choice of the variance is the following. As a second order moment, the variance efficiently captures luminance discontinuities along the line. In the image, these discontinuities correspond to edges that are orthogonal to the projection direction. So, in final, the variance is expected to capture relevant information about the distribution of edges on the image, which in turns characterizes the visual content of an image.

Formally, we define the RADIal haSHing (RADISH) feature vector as follows. Let $\Gamma(\phi)$ denote the set of pixels (x, y) on the projection line corresponding to a given angle ϕ. Letting (x', y') denote the coordinates of the central pixel, $(x, y) \in \Gamma(\phi)$ if and only if

$$-\frac{1}{2} \leq (x - x').\cos\phi + (y - y').\sin\phi \leq \frac{1}{2} \tag{3.1}$$

Letting I(x,y) denote the luminance value of pixel (x, y), the RADISH feature vector $R[\phi]$, $0 \leq \phi \leq 180$, is then defined by

$$R[\phi] = \frac{\sum_{(x,y)\in\Gamma(\phi)} I^2(x, y)}{\#\Gamma(\phi)} - \left(\frac{\sum_{(x,y)\in\Gamma(\phi)} I(x, y)}{\#\Gamma(\phi)}\right)^2 \tag{3.2}$$

Figure 3.2 present the RADISH feature vector obtained for Lena.

Obviously, the 180-samples feature vector contains partly redundant information. We now explain how to derive a compact image digest from the redundant feature vector. We use the Discrete Cosine Transform (DCT) to decorrelate the RADISH feature sample.

Figure 3.3 represents the variance of the RADISH DCT coefficients derived from a 40 images subset of the USC-SIPI database. We observe that most of the RADISH feature vector energy is compacted on few low frequency DCT coefficients. We conclude that the DCT efficiently decorrelates the feature vector samples. In addition, Figure 3.4 presents, for each RADISH DCT coefficient, an estimation of the noise to signal energy ratio due to JPEG compression. We observe that the ratio remains almost constant for the 40 first (low frequency) coefficients, but progressively increases beyond that point.

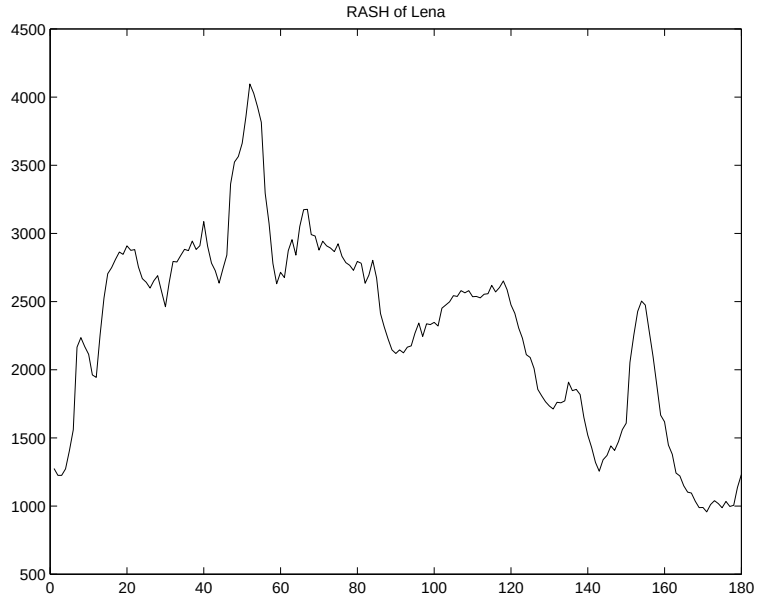

Figure 3.2: *Example of RADISH feature vector for Lena image.*

Based on these observations, we define the RADISH image digest as the 40 low frequency DCT coefficients of the RADISH feature vector. Formally, given the RADISH feature vector $R(\phi)$, with $0 \leq \phi \leq 180$, the RADISH image digest coefficients $D(n)$, are defined by

$$D(n) = \sqrt{\frac{2}{N}} . \sum_{\phi=0}^{N-1} \left(R(\phi) . \cos \frac{\pi . (2\phi + 1) . n}{2N} \right), \quad 1 \leq n \leq N = 40 \qquad (3.3)$$

In practice, each coefficient is quantized on 8 bits, so that the quantization noise remains negligible in front of the noise due to common image processing manipulations. This results in a 320 bits image digest. When the size of the digest becomes an issue, the quantizer has to be optimized in accordance with the decision engine, i.e. the module in charge of deciding whether the digest extracted from a candidate image corresponds to a precomputed digest or not. Figure 3.4 shows that the energy of the noise produced by image compression on the RADISH digest is more or less proportional to the signal energy. This suggests that the noise is more important in low frequencies than in high frequencies, which in turns means that the optimal decision engine should "amplify" the high frequency compo-

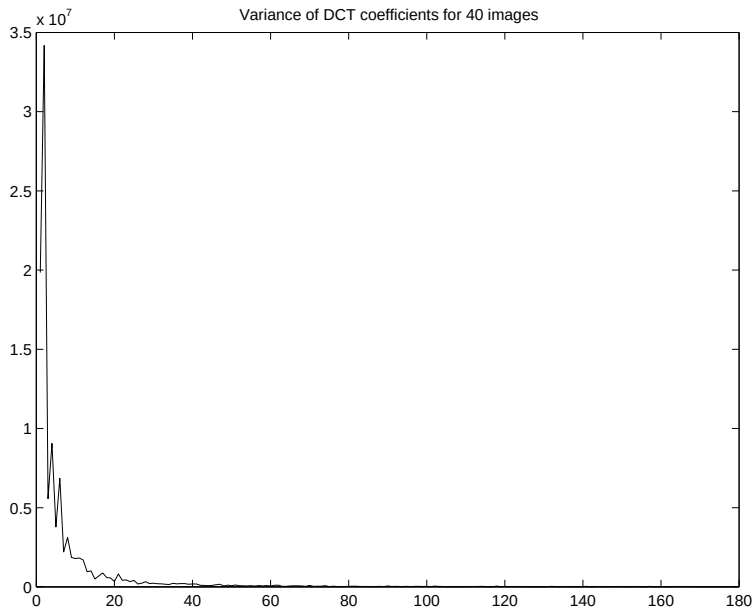

Figure 3.3: *Expected energy of the 180 RADISH feature vector DCT coefficients. In this figure, the expected energy corresponds the variance of the DCT coefficients computed on 40 images of the USC-SIPI dataset.*

nents of the digest signal before convolution. However, the characterization of the noise resulting from any kind of image manipulations is beyond the scope of this paper. So, in the following, the decision module assumes a white noise, and is based on digest cross-correlation computations, without initial "amplification" of the high frequency components of the signal. In this case, the optimal quantizer is uniform.

3.2.2 Visual hash experimental validation

In this section, we evaluate the robustness and collision resistance of our proposed hashing method. This is done through experiments on 40 real-world images taken from the USC-SIPI database [13]. For comparison purpose, we also provide results based on the 64-bin luminance histogram, which is one of the most commonly used image feature in content-based retrieval system [1].

For each of the 40 images of the dataset, we consider 8 image processing

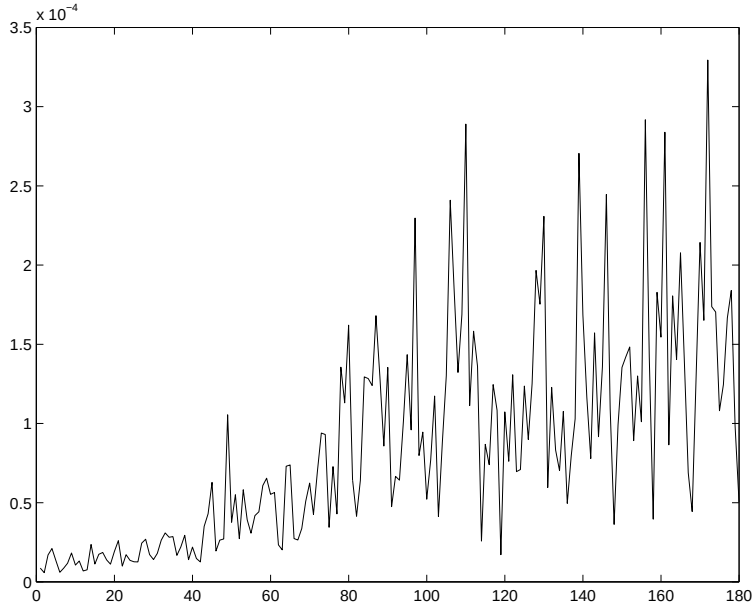

Figure 3.4: *Expected noise to signal ratio for the 180 RADISH DCT coefficients. The signal energy is the variance computed on the RADISH DCT coefficients of the database images. The noise energy is the variance of the RADISH DCT coefficients computed on the error obtained when encoding the database images with JPEG-60%.*

attacks, generating 320 images, named processed images in the following. The 8 image processing manipulations envisioned in the experiments are:

- **Filtering**: 3x3 Gaussian filtering with standard deviation of 0.5 and 3x3 average filtering.
- **Compression**: JPEG compression with 80 and 60% quality factor.
- **Geometric**: scaling (factors = 1.2, 0.8), rotation (2^0), and rotation (1^0) followed by "inside-box" cropping.

Figure 3.5 and Figure 3.6 depicts the peaks of cross correlation (PCCs) measured between each pair of original and corresponding processed frame. Both the RADISH and histogram-based digests are considered. Figure 3.5 is devoted to filtering processes, while Figure 3.6 focuses on geometrical distortion. By comparing the right and left columns of each

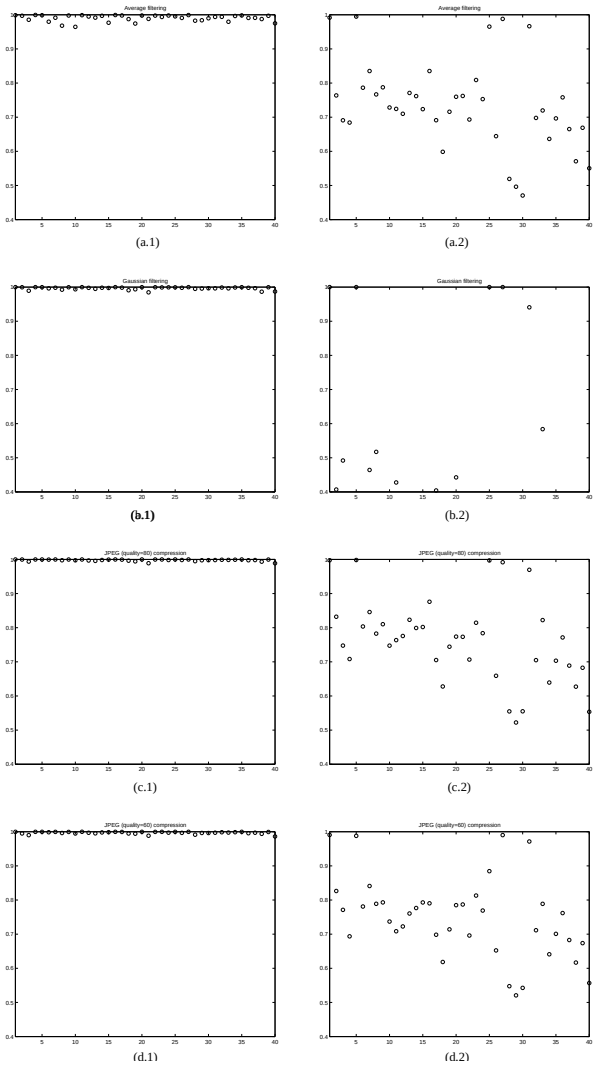

Figure 3.5: *Peak of cross-correlation between the image digests computed for each one of the 40 pairs of original and corresponding processed frame. Left column: RADISH image digest; right column: histogram-based feature vector. Each line corresponds to a specific filtering: (a) 3x3 average filtering, (b) gausian filtering, (c) JPEG-80% quality factor, (d) JPEG-60% quality factor.*

figure, we observe that most RADISH PCCs remain close to one, while numerous histogram-based PCCs dramatically fall down due to image pro-

cessing. We conclude that the RADISH digest is significantly more robust to image processing than the histogram. In both figures, we also observe that all RADISH PCCs are larger than 0.85. We conclude that this value is a good candidate threshold to decide whether two images are visually similar or not. This is confirmed in the following.

To evaluate the risk of collision, in Figure 3.7, we compare, for each original image from the dataset, the worst Intra matching with the best Inter matching. Intra and Inter matching are defined as follows. Given a reference original image, we classify the 320 processed images in Intra and Inter processed images, depending on whether they have been derived from the reference image or not. Then, we compute the peaks of cross-correlation (PCCs) between the feature vector reconstructed from the stored reference original image digest, and each one of the feature vectors extracted from the processed image. We refer to an Intra (Inter) matching to denote the PCC computation with an Intra (Inter) processed images.

Figure 3.7 presents the worst Intra PCCs and the best Inter PCCs for each image from the dataset. From this figure, we observe that all Intra PCC's are larger than 0.85, and that no Inter PCC lies under 0.85. We conclude that cross correlation is an efficient way to compare two RADISH digests, and that 0.85 is a good threshold to decide whether two images are visually similar or not.

Our tests also revealed that the images leading to low Intra PCC contain a lot of high frequency "random" textures. This suggests that the proposed digest is not able to capture complex textural information. This also suggests that the proposed RADISH algorithm could benefit from a preprocessing of the input image. Specifically, low-pass filtering of the input image would result in an image that is better suited to our proposed digest method. This idea is left to future investigation, and has not been considered in the rest of the paper.

For comparison purposes, Figure 3.8 provides the worst Intra and best Inter PCCs computed based on the histogram digest. We observe that in nearly all cases, the worst Intra matching is lower than the best Inter matching, which indicates that after processing, it is not possible to partition Intra and Inter images based on histogram digest comparisons.

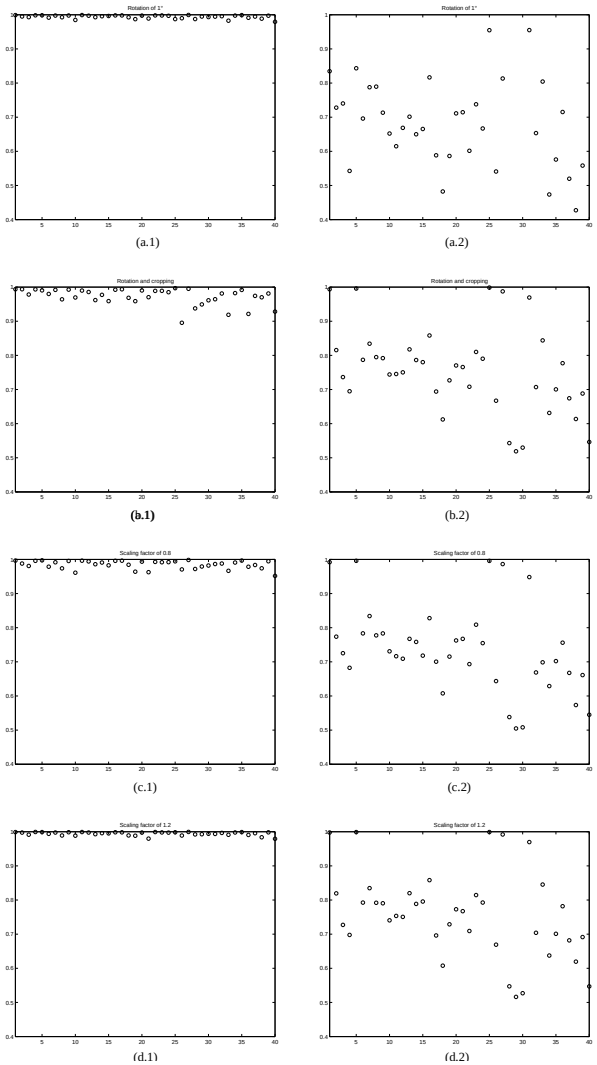

Figure 3.6: *Peak of cross-correlation between the image digests computed for each one of the 40 pairs of original and corresponding processed frame. Left column: RADISH image digest; right column: histogram-based feature vector. Each line corresponds to a specific geometric distortion: (a) 2 rotation, (b) 1 rotation followed by an inside-box cropping, (c) scaling by a 0.8 factor, (d) scaling by a 1.2 factor.*

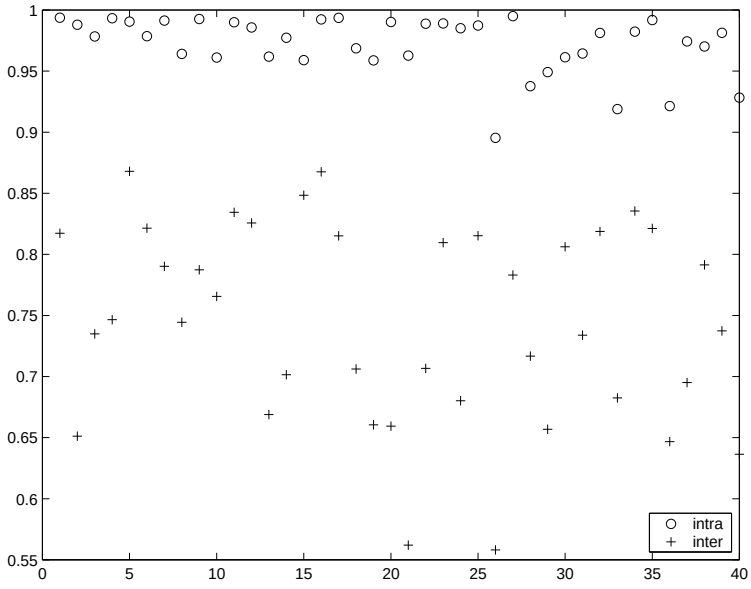

Figure 3.7: *Peak of cross correlation based on RADISH image digests. For each one of the 40 tested images, each graph compares the worst INTRA matching with the best INTER matching. INTRA matching is computed between the RADISH digests extracted from an original frame and any of its corresponding processed frames. On the contrary, INTER matching is computed between the digests extracted from an original frame, and from any other frame that is not derived from the original frame.*

3.3 Extension to video hashing

In Section 3.2, we have defined the robust RADISH image digests. In this section, our purpose is to extend the notion of "image hash" to "video hash". In Section 3.3.1, we introduce the philosophy of our method. In short, our approach consists in selecting *representative frames* to characterize the video sequence, and to extract the image digests corresponding to these frames to build the video sequence digest. A representative frame is defined to be associated to a video shot, and to characterize the video shot visual content. By definition, there is a one-to-one mapping between representative frames and sequence video shots. Based on this definition, Section 3.3.2 explains how to select representative frames based on video shot detection algorithms. In the literature, *key-frames* are defined to be

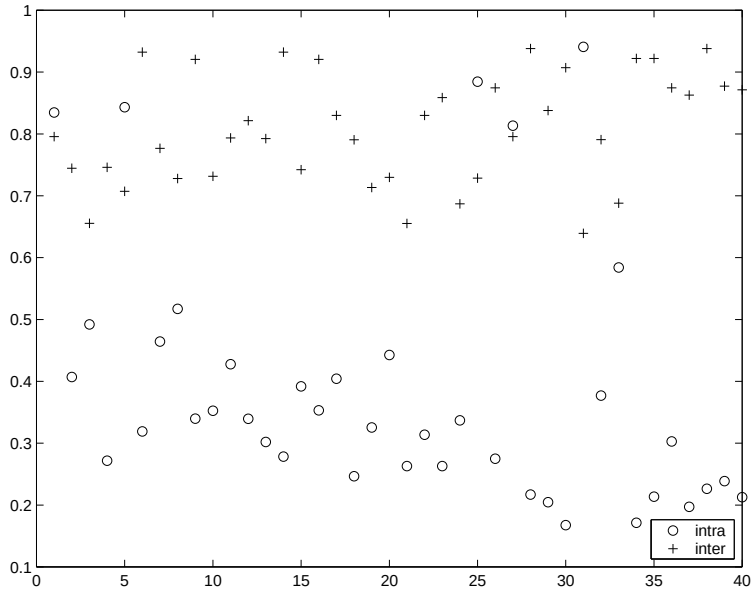

Figure 3.8: *Peak of cross correlation based on the histogram feature vector. For each one of the 40 tested images, each graph compares the worst INTRA matching with the best INTER matching. INTRA matching is computed between the histograms extracted from an original frame and any of its corresponding processed frames. On the contrary, INTER matching is computed between the histograms extracted from an original frame, and from any other frame that is not derived from the original frame.*

the boundaries of video shots. As a consequence, in Section 3.3.2, several approaches are considered to detect key-frames. Eventually, Section 3.3.3 validates the performance of an approach that combines pseudo-global and local criteria with respect to the representativeness and robustness of the selection process. It demonstrates that the proposed video digest is able to uniquely identify the video sequence visual content.

3.3.1 From ”image hash” to ”video hash”: the notion of representative frames

A naive way to extend ”image hash” to ”video hash” is to compute an image digest for each frame of the video sequence. However, this approach suffers from numerous drawbacks. First, it is computationally expensive.

Second, it results in a large sequence digest. Third, temporal axis processing, e.g. subsampling, strongly affects the sequence digest.

To circumvent these drawbacks, we notice that most real-life video sequences can be temporally separated into video shots, within which the frames are visually similar. Therefore, we decide to model the video sequence as a collection of video shots. Widely used in the area of content-based information retrieval [1], this approach describes the video sequence as a set of feature vector, with one vector per video shot. In our case, feature vectors are the image digests of carefully selected *representative frames*. To select the representative frames, we first detect some easily detectable frames, called *key-frames*. Key-frames are defined as the ones that delimit groups of similar frames, i.e. they are the video-shot boundaries. Once they have been located, one representative frame is selected between each pair of consecutive key-frames. The image digests that are computed on these representative frames are expected to form the video sequence digest. So, the frames that are selected between pairs of key-frames have to be representative of the video shot visual content. This is the reason why we call them *representative frames*.

So doing, even after a change of frame rate, or a frame drop, the key-frame detection algorithm hopefully identifies the same video shots in the original sequence, and in the processed sequence. Hence, the representative frames corresponding to the video shots in each sequence, are expected to be visually similar, if not identical. As a consequence, the robustness of the image digest computation algorithm allows for correct similarity measurements between the original and the processed video sequences. These statements are validated through experiments in Section 3.3.3.

3.3.2 Representative frames

In this section, we explain how to detect sharp visual changes in a video sequence, i.e. how to detect video key-frames. We also define how to select a representative frame between a pair of consecutive key-frames. Remember that only representative frames are intended to be hashed.

Key-frames detection overview

Shot boundaries detection has been the subject of many researches, both for video indexing and content based retrieval applications. The first section presents a number of related works in this area. In summary, most

video shots boundaries detection algorithms are based on the fact that the visual content of the images changes between two shots. The goal of the key-frame detection algorithm is therefore to find significant disparities between consecutive frames of the sequences. This approach is especially suited to our application, for which the detection of other frames than actual boundaries between semantically distinct video shots, e.g. frames that correspond to flashes in a shot, is not an issue. The most important for us is to detect the same key-frames in the original and in a processed, visually similar, sequence.

Most key-frames detection algorithms are based on a three steps model:

- First, extract a feature from each frame of the video sequence.
- Second, use a metric $d(t, t-\tau)$ to measure the distance between the features extracted at time t and $t-\tau$. The distance $d(t, t-\tau)$ is expected to measure the disparity between frames at time t and $t-\tau$.
- Third, compare the distance values $d(t, t-\tau)$ to a threshold T. If $d(t, t-\tau) > T$, the frame at time t is marked as being a key-frame. In general, $\tau = 1$.

We now consider each of these steps in more details.

Video shots detection literature overview

There exists an extensive literature about video shots detection methods. Most of them are based on a measure of disparity between successive frames. Some measure frame disparities in the spatial luminance domain, while others work in a transformed [7], compressed domain [8]), or feature-based domain [9]. Some approaches are based on simple disparity measurements, e.g. the distance between histograms, while others exploit more complex information, like an estimation of the motion between successive frames [10, 11]. Once the disparity measurements are available, a decision is taken about the position of video shot boundaries. For this purpose, some methods define heuristic or automatic thresholds, while other exploit more sophisticated statistical models [2, 12, 13] or *a priori* knowledge about the shots lengths distribution [14].

In the context of our proposed video digest system, the aim of the shot detection algorithm is to identify key-frames that can still be detected in a visually similar processed video sequence. As a consequence, the goal of

our key-frame detection algorithm is basically reduced to hard cuts identification, i.e. to the detection of sharp scene changes in the video sequence. Our purpose is certainly not to detect fades [15], dissolves [11, 16], or wipes [17] transitions. For this reason, in Section 3.3.2, we have intentionally limited our investigations to simple detection algorithms based on luminance histogram disparity measurements, and on the use of automatic thresholds.

Extracted feature and distance measurement for key-frames detection

Common features to evaluate the disparity between video frames are the frame pixels color intensities and their histograms. For each of them, a number of distance measures have been proposed in the literature. According to [2] and [3], the ℓ_1 norm histogram difference provides the best performances w.r.t. key-frames detection. In this section, we envision the use of the 40-samples RADISH feature vector instead of the 64-bins histogram to detect key-frames. To compare the two approaches, we estimate their capability to aggregate the frames of a video sequence into compact clusters. We conclude that there is no benefit to use the RADISH digest. As a consequence, in the rest of the paper, we use the ℓ_1 norm between 64-bins luminance histograms as the reference frame distance measurement method. We now detail and motivate this decision.

To compare the RADISH digest and the histogram feature vector with respect to their ability to capture representative shot boundaries in the video sequence, we estimate their capability to aggregate the frames of a video sequence into compact clusters. Based on the temporal structure of the video, we adopt the following definition of a video cluster. Let j and $(j+1)$ denote the indices of two successive frames in the video sequence. Let x_j and x_{j+1} be their respective extracted features, and $d(x_j, x_{j+1})$ denote the distance measured between x_j and x_{j+1}. Hence, j and $(j+1)$ belong to the same cluster if and only if $d(x_j, x_{j+1}) < \epsilon$. In general, this definition is likely to produce chain-like clusters where one end of a cluster is very far from the other end. However, it has been found in [1] that most clusters defined according to the above condition in real video sequences are compact, i.e. all frames in a cluster are such that the distance between their feature vectors is smaller than ϵ.

Given an ϵ value, the clustering structure of a video can be computed as follows. We scan the frames in chronological order. Given two consecutive frames j and $(j+1)$, if $d(x_j, x_{j+1}) \leq \epsilon$, frame $(j+1)$ belongs

to the same cluster as frame j. If $d(x_j, x_{j+1}) > \epsilon$, a new cluster is created. Different ϵ values result in different clustering structure. The smallest the ϵ, the more clusters. Figure 3.3.2 presents the normalized size of clusters as a function of the number of clusters, when clusters are computed based on the RADISH or 64-bins histogram l_1 norms. Given a partition $\mathcal{C} = \{C_1, C_2, ..., C_M\}$ of the video sequence into M clusters $C_i, 0 < i \leq M$, the normalized size of clusters is defined as follows. Let $\Theta_{\mathcal{C}}$ defines the pairs of indices that belong to distinct clusters, i.e. $\Theta_{\mathcal{C}} = \{(k,l)|k \in C_i,\ l \in C_j,\ i \neq j\}$. Given a cluster $C_i \in \mathcal{C}$, let Ω_{C_i} denote the pairs of distinct indices in C_i, i.e. $\Omega_{C_i} = \{(k,l)|k \in C_i,\ l \in C_i,\ k \neq l\}$. Hence, the normalized size $\chi_{\mathcal{C}}$ of the clusters in partition $\mathcal{C}$ is defined as follows

$$\chi_{\mathcal{C}} = \frac{\overline{d_{l_1,INTRA}}(\mathcal{C})}{\overline{d_{l_1,INTER}}(\mathcal{C})} \tag{3.4}$$

with

$$\overline{d_{l_1,INTRA}}(\mathcal{C}) = \frac{\sum_{C_i \in \mathcal{C}} \sum_{(k,l) \in \Omega_{C_i}} d_{l_1}(k,l)}{\sum_{C_i \in \mathcal{C}} (\#\Omega_{C_i})} \tag{3.5}$$

and

$$\overline{d_{l_1,INTER}}(\mathcal{C}) = \frac{\sum_{(k,l) \in \Theta_{\mathcal{C}}} d_{l_1}(k,l)}{\#\Theta_{\mathcal{C}}} \tag{3.6}$$

Based on this definition, we understand that, for a given number of clusters, the lower the normalized cluster size, the more compact the clusters. In Figure 3.3.2, we observe that the RADISH and histogram features result in equivalent compactness of the video clustering structure. So, because of its computational simplicity, in the rest of the paper, we use the l_1 norm between 64-bins luminance histogram as the frame distance measurement method.

Threshold definition for key-frames detection

Once the distance $d(t, t-1)$ between the features of frames t and $(t-1)$ has been computed, it is compared to a threshold to decide whether t is a key-frame or not. Next to heuristic thresholds, automatic thresholds, either global or adaptive, have been proposed in the literature. This section presents them, and proposes to combine them to circumvent their respective drawbacks.

A global automatic threshold was introduced in [4]. Letting μ and σ denote the mean and the standard deviation of the distance measure, the threshold is set to $\mu + \alpha\sigma$. [4] assumes that within a shot, all changes are

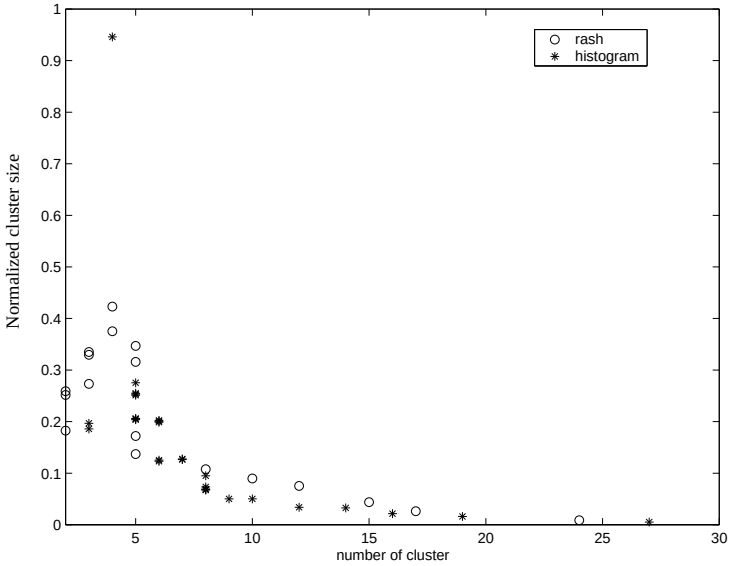

Figure 3.9: *Normalized cluster size as a function of the number of clusters. Given an ϵ value, clusters are defined in chronological order, such that frame $j+1$ belongs to the cluster of frame j if and only if $d(x_j, x_{j+1}) < \epsilon$.*

due to a white gaussian noise. As a consequence, with $\alpha = 3$, 99.9% of the discontinuity values within a shot are under the threshold, and upper values can be considered as key-frames. The main disadvantage of this method is that the threshold can only be computed *a posteriori*, after an initial estimation of the μ and σ parameters.

On the contrary, adaptive methods compute the threshold based on a temporal sliding window [5]. Letting $2L_1 + 1$ denotes the size of the sliding window, and t be the index of the window center, a key-frame is detected in t if both following conditions are true:

- First, the distance measured at window center is maximal on the window, i.e.

$$d(t, t-1) > d(t+i, t+i-1) \ \forall i \in [-L_1, L_1] \backslash \{0\}. \tag{3.7}$$

  This avoids having too close key-frames.

- Second, the distance measured at window center is higher than α_1

times the second higher distance measured in the window :

$$d(t, t-1) \geq \alpha_1 d_{max2}. \tag{3.8}$$

The α_1 parameter has to be chosen experimentally. Moreover, when the discontinuity values are low and do not change much, a key-frame might be detected within a shot despite a very low discontinuity value.

From the above definitions, a global threshold appears to be unpractical, while an adaptive threshold sometimes detects irrelevant key-frames, e.g. when the distance measured between successive frames in the sliding window are all near zero. To circumvent the problem, we propose to combine both approaches, and to compute two thresholds. The first threshold, T_1, is adaptive, while the second, T_2, is "pseudo-global". By "pseudo-global", we mean that the threshold is computed on the fly, on a large sliding window of size $2L_2 + 1 >> 2L_1 + 1$. Both thresholds complement each other. On the one hand, when the distance measured between successive frames is near zero, the pseudo-global threshold avoids detecting non-relevant key-frames. On the other hand, when measured distance indicate large disparities between successive frames, the adaptive threshold prevents the detection of too many key-frames in a short period of time.

Key-frames detection algorithm: a summary

Formally, the key-frame detection algorithm can be described in four steps as follows

- **Measure the distance between successive frames**

$$d_{l_1}(t, t-1) = \sum_{j=1}^{N} |H_t(j) - H_{t-1}(j)|, \tag{3.9}$$

  where $H(t)$ is the 64-bins luminance histogram of frame at position number t.

- **Compute the adaptive threshold.**

$$T_1 = \alpha_1 d_{max2}(t). \tag{3.10}$$

  where $d_{max2}(t)$ is the second maximum value of $d_{l_1}(t+i, t+i-1)$ for $i \in [-L_1, L_1] \backslash \{0\}$.

- **Compute the pseudo-global threshold.**

$$T_2 = \mu(t) + \alpha_2\sigma(t) \tag{3.11}$$

  where $\mu(t)$ and $\sigma(t)$ are the mean and the standard deviation of the discontinuity values, d_{l_1}, which are in the range $[t - L_2, t + L_2]$

- **Key-frames detection.** Frame t is a key-frame if $d_{l_1}(t, t-1) > max(T_1, T_2)$ and if $d_{l_1}(t, t-1)$ is the maximum of the window of length $2L_1 + 1$.

Parameters $L_1, L_2, \alpha_1, \alpha_2$ have been found empirically. Nevertheless, we can help us of these reflections :

- L_1 is the minimum amount of frames that we can have between two shots boundaries. As in a movie we do not have shots shorter than one second, L_1 will be around 10.
- L_2 must be larger than L_1.
- The α_1 value must be taken in the range $2 \leq \alpha_1 \leq 6$ [5].
- The α_2 value must be taken in the range $2.5 \leq \alpha_2 \leq 3.5$, on the basis of the results presented in [4].

For our tests, we have used $L_1 = 10, \alpha_1 = 2, \alpha_2 = 3$, and $L_2 = 20$ or 75.

Representative frames selection

Once the key-frames have been detected, one representative frame is selected between each pair of consecutive key-frames. Specifically, given the indices k_1 and k_2 of two consecutive key-frames, the representative frame is defined as the one that minimizes the disparity measurement with its preceding frame. Formally, the index r of the representative frame is defined by

$$r = \arg \max_{k_1 < k \leq k_2} d_{l_1}(k, k-1) \tag{3.12}$$

In final, the video digest corresponds to the set of image digests computed based on these representative frames, which are thus also named hashed frames in the following.

Computing the video digest based on representative frames rather than on shot boundaries deals with the fact that the key-frames detected in

the original and processed sequences might not be strictly identical. This problem is due for example to the temporal re-sampling performed when shifting from one video format to another (25 fps for PAL, 29,97 fps for NTSC and 24 fps for digital cinema). It is thus preferable to compute the video digest on frames which are in flat regions, where several following frames are quite the same in a visual point of view. Hence, it is better to compute image digests on representative frames, which are surrounded by visually similar frames, rather than on key-frames, which are characterized by large disparities with their neighboring frames.

3.3.3 Video hash experimental validation

This section presents a number of experimental results to validate our approach to video digest. First, we study the robustness of the key-frame detection algorithm, i.e. we analyze whether the same video shots are identified before and after video sequence processing. So doing, we demonstrate that the key-frames detection algorithm based on the combination of an adaptive and a pseudo-global threshold outperforms other approaches. Then, we evaluate the representativeness of the hashed frames, selected between pairs of consecutive key-frames. We show that once again the combination of adaptive and global thresholds results in increased representativeness of the frames that are selected to characterize the video sequence. Finally, we validate the whole system by matching the representative frames that are selected in a processed video sequence, with the representative frames that are selected either from the corresponding original sequence, or from other original video sequences. This computation demonstrates that, even for strong degradation of the video sequence (PSNR lower than 25 dB!), our method remains able to identify the original sequence that corresponds to each processed sequence.

In the experiments presented in this section, we consider three original sequences, each sequence being extracted from a DVD support. The sequences are:

- **Monster**: 576x304 size, 1341 frames selected.
- **Swordfish**: 642x272 size, 1364 frames selected.
- **Star wars Episode I**: 688x320 size, 1092 frames selected.

For each original sequence, a processed video sequence is obtained by capturing the sequence displayed on a screen. The average PSNRs of all pro-

cessed sequences lie between 23 and 25 dB, which corresponds to severe distortions.

For each original and processed video sequence, we detect key-frames, select representative frames, and compute their image digests. These operations are performed in real time by an application developed in VisualStudio on a Pentium III. The numbers of representative frames selected in the original video sequences are respectively 9 for Monster (= 0.7% of the total number of frames), 20 for Swordfish (= 1.5%), and 18 for Star Wars Episode 1 (= 1.6%). Note that Monster is an animation movie, which explains why the number of hard cuts is lower than in other movies.

Key-frames selection robustness

To validate the performance of a key-frame detection algorithm, we compare the set of key-frames detected in a given original sequence with the one detected in a corresponding processed sequence. We introduce the "Similarity" measurement to quantify the equivalence between the sets of original and processed key-frames. The "Similarity" measurement is associated to a given detection method, and to a pair of original and processed video sequences. The "Similarity" measure combines the "recall" and "precision" measures defined in [6], and can be expressed as follows

$$Similarity = \frac{\#Correct}{\#Correct + \#False\,alarm + \#Missed}. \tag{3.13}$$

In equation (3.13), "$\#Correct$" denotes the number of key-frames that are detected in both the original and the processed video sequences. The "Missed" key-frames are the frames that are detected in the original sequence but not in the processed sequence. The "False Alarm" key-frames are the frames that are detected in the processed sequence, but not in the original sequence.

We now exploit the measure of "Similarity" to compare different key-frames detection algorithms. Given a pair of original and processed video sequences, we test four different methods, using either a global, a pseudo-global, an adaptive, or a combination of adaptive and pseudo-global thresholds. In practice, three pairs of original and processed video sequences have been considered. As told above, they correspond to the movies Starwars I (SW1), Monster (Mon), and Swordfish (Swo). In each case, the original sequence has been extracted from the DVD, while the processed sequences have been created by filming the original sequences

projected on a screen. The distortion between the original and the processed sequence is significant, i.e. PSNR $\in$ [23dB-25dB]. Table 3.1 presents the measured "Similarities". From Table 3.1, we conclude that the combination of pseudo-global and adaptive thresholds performs better than the other approaches.

Threshold	SW1	Mon	Swo
Global	$60,7\%$	$66,7\%$	$63,7\%$
Adaptive	$80,9\%$	$64,3\%$	$73,9\%$
Pseudo-Global ($L_1 = 20$)	$53,3\%$	$32,2\%$	$67,8\%$
Pseudo-Global ($L_1 = 75$)	$80,9\%$	$47,4\%$	$75,0\%$
Combined ($L_1 = 20$)	$85,0\%$	$69,2\%$	$77,2\%$
Combined ($L_1 = 75$)	$89,4\%$	$66,7\%$	$75,0\%$

Table 3.1: *Similarity between two key-frames sets*

Video digest representativeness

To compare the *representativeness* of the different key-frames detection algorithm, we consider the percentage of frames in the original sequence that are *similar* to the representative frames selected based on each key-frames detection algorithm. First, we define what we mean by "similar". Then, we present the experimental results.

Given a parameter ϵ, we say that two adjacent frames are ϵ - identical if the distance measured between these two frames is smaller than $(\epsilon/100).d_{max}$, where d_{max} is the maximum distance measured between two consecutive frames on the whole sequence.

Based on this definition, the set of ϵ-similar frames $\mathcal{S}_{r,\epsilon}$ associated to a representative frame r is defined as the largest set of consecutive frames that contains r and such that all pairs of adjacent frames are ϵ-identical.

Figure 3.10 and 3.11 display the percentage of frames that are ϵ-similar to a representative frame as a function of ϵ, for different key-frame detection algorithms. Figure 3.10 aggregates the results for the three original video sequences introduced above, while Figure 3.11 corresponds to the three processed sequences. In these figures, we observe that the curve corresponding to the combined approach lies above all other curves. We conclude that the key-frame detection based on the combination of an adaptive and global threshold better capture the essence of the video sequence.

On these figures we also note that the ϵ values corresponding to a given percentage of ϵ-similar frames are much larger for the processed sequences than for the original ones. This is a consequence of the strong distortion affecting the processed frames.

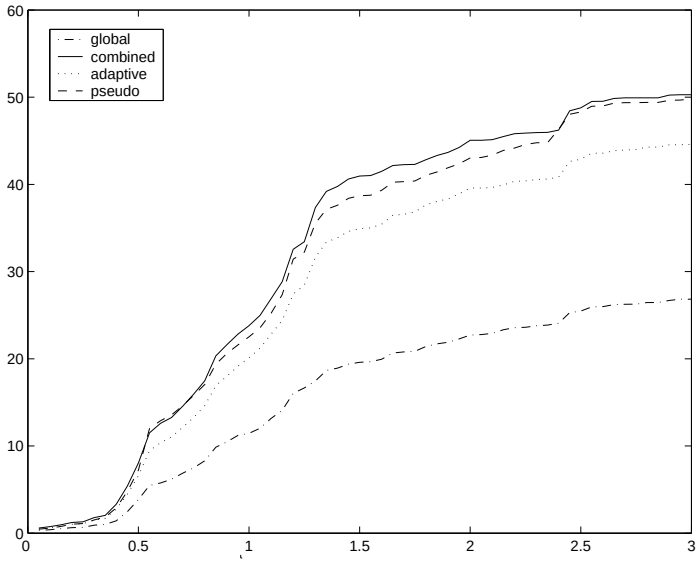

Figure 3.10: *Percentage of video frames that are ϵ-similar to a representative frame, as a function of ϵ. Original video sequences are considered.*

Video digest system validation

In this section, we analyze the matching between the video digests computed on original and processed video sequences. We say that an original sequence *corresponds* to a processed sequence if and only if the processed sequence has been derived from the original sequence.

In Figure 3.12, we consider the "Monster" processed video sequences, and the three original sequences, i.e. "Monster", "Starwars", and "Swordfish". 9 representative frames have been selected in the original sequence, while 12 frames have been selected in the processed sequence. Among these 12 frames, 9 frames have a visually similar frame among the frames selected in the original, while 3 frames do not have a counterpart among the 9 original frames. For each of the 12 processed representative frames, Fig-

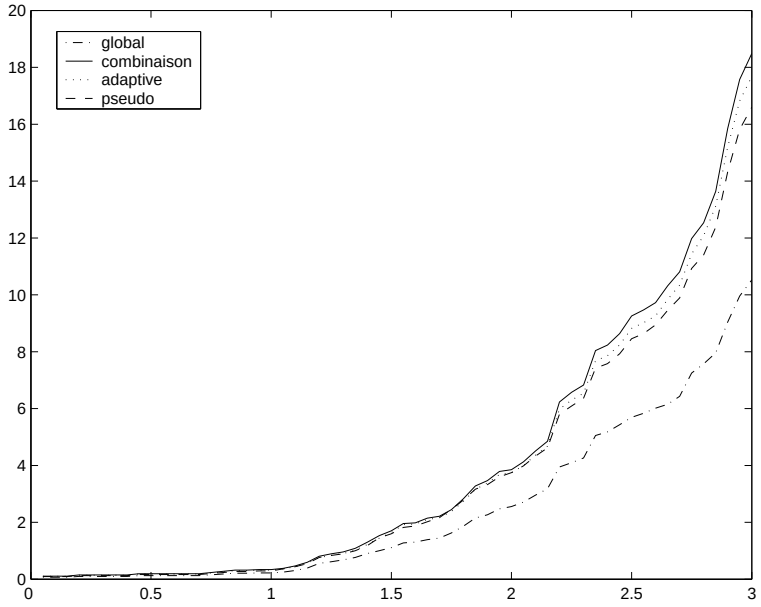

Figure 3.11: *Percentage of video frames that are ϵ-similar to a representative frame, as a function of ϵ. Processed video sequences are considered.*

ure 3.12 compares INTRA and INTER matching. Given a representative frame selected on the processed sequence, INTRA matching is defined as the largest cross-correlation measured with one of the frame digests of the corresponding original sequence, while INTER matching is defined as the largest cross-correlation measured with any of the digests extracted from other original sequences (Swordfish and Starwars) and from the USC-SIPI database. In Figure 3.12, the frames labeled 8, 9, and 11 correspond to representative frames that are selected in the processed sequence, but do not have an equivalent among the representative frames selected in the original sequence. We observe that these frames achieve a poor INTRA matching, which is what we are expecting. We also observe that most other frames provide higher cross-correlation values for INTRA than INTER matching. Moreover, most INTRA matching achieves a cross-correlation higher than 0.85, which is the empirical threshold found in Section 3.2.2. Actually, among the 9 processed frames that have an original counterpart, only the frames labeled 1, 4, and 6 achieve an INTRA cross-correlation lower than 0.85. Among these three frames, only the frame 4 presents a

lower INTRA than INTER cross-correlation. We conclude that the proposed video digest is an efficient tool to estimate the similarity between two video sequences. This conclusion is confirmed in Figure 3.13.

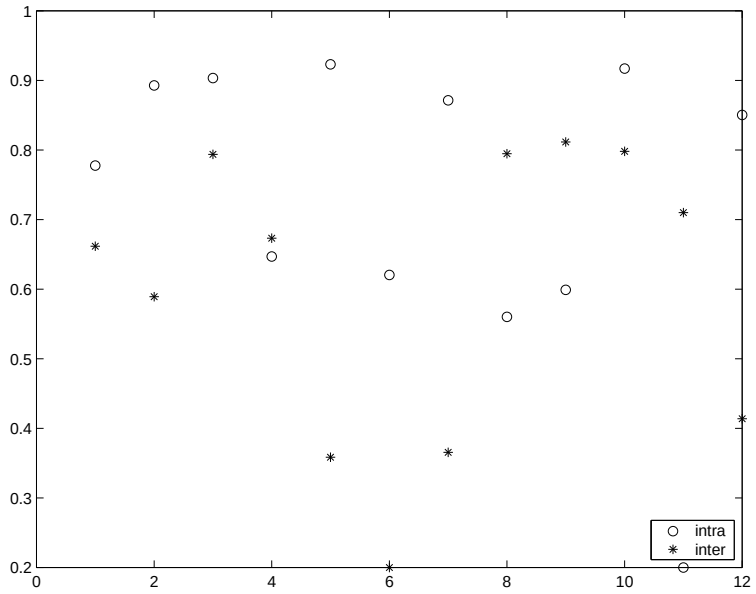

Figure 3.12: *INTRA and INTER peaks of cross-correlation (PCCs) for the Monster video sequence, as a function of the processed representative frame index. Frames labeled 8, 9, and 11 correspond to representative frames that are selected in the processed sequence, but do not have an equivalent among the representative frames selected in the original sequence.*

In Figure 3.13, each line provides the 3 graphs that are computed when matching a candidate processed video sequence with one of the 3 original video sequence. Specifically, given a processed sequence and an original sequence, the graph plots the best match between each processed representative frame digest, and any of the digests associated to the original in the database. To analyse Figure 3.13, it is useful to mention that among the selected representative frames of each processed sequence, some frames do not have an equivalent among the representative frames selected in their corresponding original sequence. These "false alarm" frames are labeled 8, 9, and 11 in "Monster", 9 and 14 in "Swordfish", and 8 and 11 in "Starwars". Not surprisingly, the digests of these "false alarm" frames

present a weak PCC with the representative digests of the corresponding original sequence. Unfortunately, we observe in Figure 3.13 that some processed frames that do have a similar frame among the representative frames of the corresponding original result in poor PCC value. This is due to the importance of distortions between original and processed sequences (PSNRs < 25 dB!). However, and this is the most important, we observe that high cross-correlation values, i.e. PCC values > 0.8, are only achieved when matching a processed sequence with its corresponding original. This criterion can thus be used to associate a candidate sequence to the correct original sequence in the database.

For completeness, note also that more complex and robust decision strategies could be imagined. An example of information that we do not exploit is the temporal ordering of the set of original frames that are matched to the representative frames selected in the processed sequence. If the original corresponds to the processed sequence, the temporal ordering of the processed representative frames should be similar to the one of the matched frame. The design of optimal decision strategies is beyond the scope of this paper, and is left for future research.

3.4 Conclusion

Our proposed image hashing method is based on a set of radial projections of the pixels luminance values. Specifically, each projection computes the variance of the pixels along a line passing through the image center and characterized by its orientation. The set of projections forms a 1-D feature vector. The image digest, also named Radial Hash (RADISH), is then obtained by quantizing the 40 first DCT coefficients of this 1-D feature vector. Experimental results demonstrate that the RADISH is specific to an image in the sense that two digests are significantly different for distinct visual contents, but are similar if they are computed on two images derived from the same original image, e.g. by geometrical transform or low-pass filtering.

Our proposed video digest is defined to be the set of RADISH image digests corresponding to a set of frames that efficiently represent the video content. We have shown that these representative frames can be selected based on conventional key-frames detection algorithms. In final, the proposed video hashing method has the advantage to be fast, and to resist to temporal subsampling, slight geometrical deformation, and compression-related distortions.

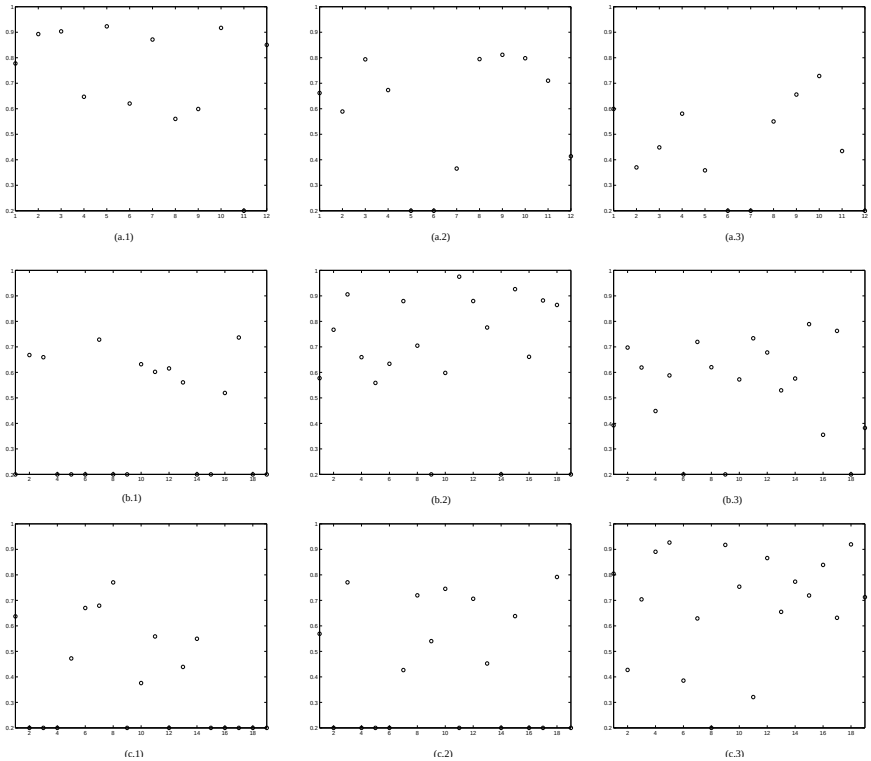

Figure 3.13: *Peak of cross-correlation (PCC) between the digests extracted from the representative frames of a processed sequence, and the digests corresponding to an original sequence. A graph corresponds to a specific processed sequence, and to a specific original sequence. For each representative frame of the processed sequence, the graph plots the best PCC obtained with one of the representative frames of the original sequence. Graphs are labeled with one letter and one figure. The letter refers to the processed video sequence, i.e. (a) for Monster, (b) for Swordfish, and (c) for Starwars. The figure refers to the original video sequence, i.e. (1) for Monster, (2) for Swordfish, and (3) for Starwars.*

Bibliography

[1] S.-C. S. Cheung, and A. Zakhor, "Efficient video similarity measurement with video signature", *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, No. 1, pp. 59-74, January 2003.

[2] D. Gatica-Perez, A. Loui, and M.-T. Sun, "Finding structure in home videos by probabilistic hierarchical clustering", *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, No. 6, pp. 539-548, June 2003.

[3] H. Zhang, Y. Gong, C.Y. Low, and S.W. Smoliar, "Image retrieval based on color features : an evaluation study," *Proceedings of SPIE*, Vol. 2606, pp.212-220, 1995.

[4] H. Zhang, A. Kankanhalli, and S.W. Smoliar,"Automatic partitioning of full-motion video,"*Multimedia Systems*, vol. 1, pp.10-28, 1993.

[5] B.-L. Yeo, and B. Liu, "Rapid scene analysis on compressed video,"*IEEE Transactions on Circuits and Systems for Video Technology*, vol. 5, pp. 533-544, December 1995.

[6] 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*, vol. 22, No. 12, pp. 1349-1380, December 2000.

[7] S.V. Porter, M. Mirmehdi, and B.T. Thomas, "Video cut detection using frequency domain correlation," *Proceedings of the 15th International Conference on Pattern Recognition*, *IEEE Computer Society*, pp. 413–416, September 2000.

[8] S.-B. Jun, K. Yoon, and H.-Y. Lee, "Dissolve transition detection algorithm using spatio-temporal distribution of MPEG macro-block types," *ACM Multimedia 2000*, pp.391-394, 2000.

[9] R. Zabih, J. Miller, and K. Mai, "A feature-based algorithm for detecting and classifying scene breaks,"*Proceedings ACM Multimedia '95*, San Fransisco, CA, pp. 189-200, November 1995.

[10] A. Hanjalic,"Shot-boundary detection: unraveled and resolved ?," in *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 12, No. 2, February 2002.

[11] A. Aner, and J. R. Kender,"A unified memory-based approach to cut, dissolve, key frame, and scene analysis," *ICIP '01*.

[12] A. Hanjalic, and H. Zhang, "Optimal shot boundary detection based on robust statistical models," *IEEE Int. Conf. on Multimedia Computing and system (ICMCS'99)*, Florence, 1999.

[13] E. Ardizzone, G.A.M. Gioiello, M. La Cascia, and D. Molinelli, "A real-time neural approach to scene cut detection," *IS&T/SPIE - Storage & Retrieval for Image and Video Databases IV*, San Jose, January 1996.

[14] N. Vasconcelos, and A. Lippman, "Bayesian video shot segmentation", *NIPS*, pp. 1009-1015, 2000.

[15] B.T. Truong, C. Dorai, and S. Venkatesh,"New enhancements to cut, fade and dissolve detection processes in video segmentation," *ACM Multimedia 2000*, pp. 219-227, November 2000.

[16] R. Lienhart,"Reliable dissolve detection," in *Storage and Retrieval for Media Databases 2001*, *Proc. SPIE 4315*, pp.219-230, January 2001.

[17] N. Chong-Wah, P. Ting-Chuen, and R. T. Chin,"Video partitioning by temporal slice coherency," *IEEE Transactions on Circuits and Systems for Video Technology*, Vol.11, Issue 8, pp.941-953, August 2001.

[18] C. Qin-Sheng, M. Defrise, and F. Deconinck "Symmetric phase-only matched filtering of Fourier-Mellin transforms for image registration and recognition,"*IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, No. 12 , pp. 1156-1168, December 1994.

[19] A. Vailaya, M. Figueiredo, A. Jain, and Hong-Jiang Zhang, "Image classification for content-based indexing", *IEEE Transactions on Image Processing*, vol. 10, No. 1, pp. 117-129, January 2001.

[20] Y. Lu, H.-J. Zhang, L. Wenyin, and C. Hu, "Joint semantics and feature based image retrieval using relevance feedback", *IEEE Transactions on Multimedia*, vol. 5, No. 3, pp. 339-346, September 2003.

[21] M. Schneider, and S.-F. Chang, "A robust content-based digital signature for image authentication", in *Proceedings of the International Conference on Image Processing*, vol. 3, pp. 227-230, September 1996.

[22] M. Johnson, and K. Ramchandran, "Dither-based secure image hashing using distributed coding", *International Conference on Image Processing*, Barcelona, September 2003.

[23] K. Otsuji, Y. Tonomura, and Y. Ohba, "Video browsing using brightness data", in *Proc. SPIE/IS&T VCIP*, vol. 1606, pp. 980-989, 1991.

[24] R. Lienhart, "Comparison of automatic shot boundary detection algorithms", in *Proceedings of SPIE on Storage and Retrieval for Still Image and Video Databases*, Vol. 3656, pp. 290-301, January 1999.

[25] A. Nagasaka and Y. Tanaka, "Automatic video indexing and full-video search for object appearances", in *Proceedings IFIP 2nd Working Conference on Visual Database Systems*, pp. 113-127, 1992.

[26] J. Smith, "Integrated spatial and feature image systems: retrieval, analysis, and compression", *PhD dissertation*, Columbia University, New York, 1997.

[27] M. Swain, and D. Ballard, "Color indexing", *International Journal on Computer Vision*, vol. 7, pp. 11-32, November 1991.

[28] U. Gargi, R. Kasturi, and S. H. Strayer, "Performance characterization of video-shot-change detection methods", *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 10, No. 1, February 2000.

[29] W. Zhao, J. Wang, D. Bhat, K. Sakiewicz, N. Nandhakumar, and W. Chang, "Improving color based video shot detection," *IEEE International Conference on Multimedia Computing and Systems*, Italy, June 1999.

[30] C. O. Toole, A. Smeaton, N. Murphy, and S. Marlow, "Evaluation of automatic shot boundary detection on a large video test suite," *The Challenge of Image Retrieval (CIR 99) - 2nd UK Conference on Image Retrieval*, Newcastle, 25-26 February 1999.

4

An advanced architecture for movie Digital Right Management

4.1 Introduction

A part of this chapter has be written in collaboration of Damien Delannay.

Digital cinema is the on-line distribution of digital movies from content providers to movie theaters servers, via satellite, optic fibers or other high speed communication lines. The relationship between distributor and users are defined by usage rules. The movie theaters (or users) receive content (movies) from distributors. They store them and project the movie in one or more theaters under some contract conditions. Piracy happens at two levels:

- The first one is obvious and consists in direct bit to bit copies done in the storage device. The pirated tapes are then sold on the black market. The corrupted copy are also distributed throught peer to peer network. This kind of piracy can be solved by proper uses of conditional access systems.
- The second one is also the responsibility of the movie theater owners. It consists in letting a spectator film the projected movie with a handy cam at the back of the theater. The distortions applied to the image constitute a real problem for watermark extraction.

The recent progresses in image processing such as image compression and progresses in signal processing such telecommunication and network facilitate copy and distribution of corrupted digital movie. The Digital Right Management issue constitues a bottleneck for a large use of digial contents and specially for digital movies. Some works have been done in this domain. A Persistent Access Control developped by Schneck present a secure and efficient system and lead to a platform combining access control based on Public Key Infrastructure and watermarking fig (4.1).
This scheme is adapted for any kind of analog waveform data. This type of global process can only ensure a tracking of the waveform or ensure a copyright protection, but not both of them. As many other architectures, the watermark is embedded at the source 4.1. This type of architecture ensure a copyright management but not a tracking management after distribution and projection in a movie theater.

4.1.1 Acces Control context

A conditional access system is much more than movie encryption or decryption. Digital Right Management is a conditional acces combined with

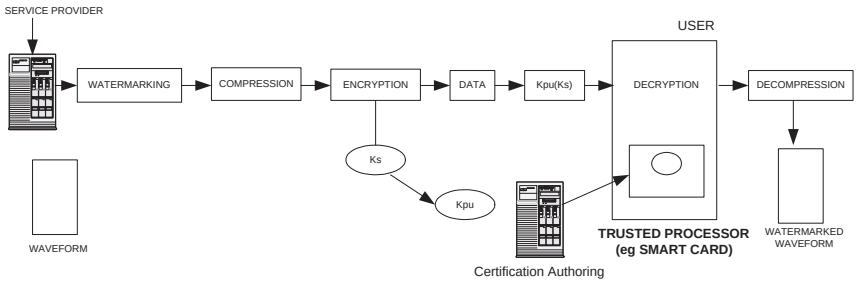

Figure 4.1: *Schneck data delivery proposal.*

watermarking. Watermarking ensures a right usage tracking. A conditional access has to manage all the projection rights and entitlements exchanged between distributors and exhibitors, it garanted the business to business system. The goal of an efficient conditional access system is to implement the usual conditions of today's practice Film Rental Agreements. Exhibitors and distributors are negotiating the projection rights together. Once the agreement is established, the system will ensure the respect of this agreement while preserving all the exhibitor possibilities to react to unplanned events. Figure 4.2 gives an overall view of an advanced conditional access system.
In this proposal, the system is working with modules located on three different places, one on the distributor side, the two others on the theater side.

The transmission of the movie and the projection rights management are handled independently. The distributor can at any time encrypt and package the film and send it to exhibitors. The encrypted film is stored on the theater central server. At the same time, distributors and exhibitors can negotiate the Film Rental Agreement. When the negotiation is concluded, the distributor encodes the projection rights for a given period through user-friendly interfaces. The system creates the entitlements, protects them and sends them to the exhibitor. The exhibitor then plans the projections for the given period. The system checks if the planning is coherent with the available entitlements and stores it in a database. Some minutes before the planned projection, the system checks if the projection is compatible with the available entitlements and with the projection history. If all the conditions are respected, the entitlements are processed to produce a new entitlement specific to the projector.

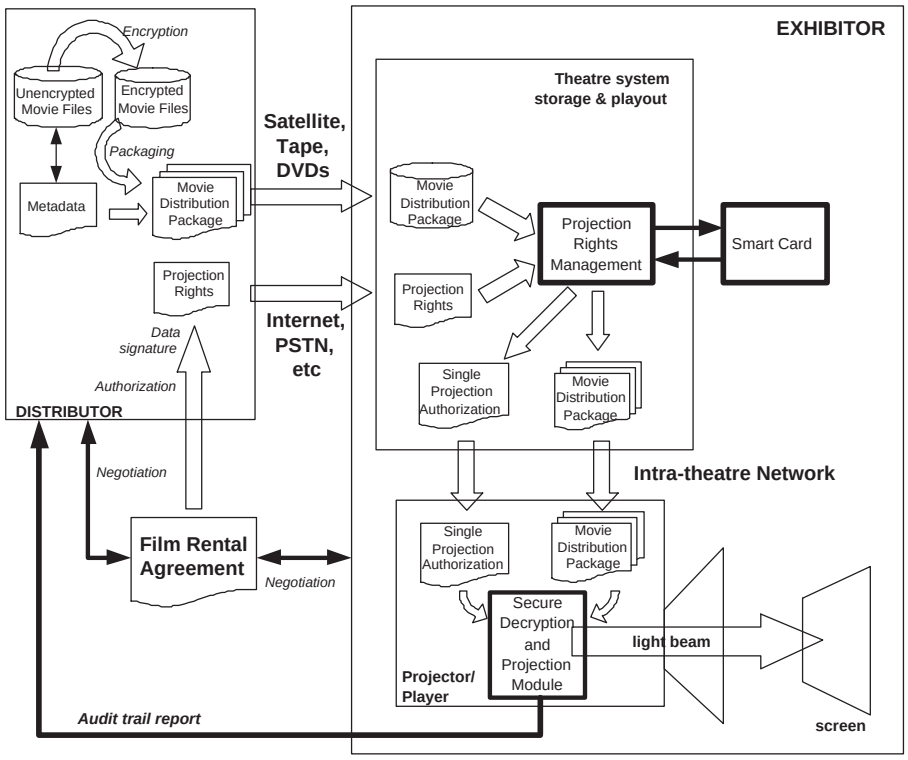

Figure 4.2: *Digital cinema conditionnal access system proposed in [1].*

Depending on the smart card memory, several distributors can use the same smart card, sharply reducing the number of smart card switches while keeping a maximum security. At the time of the projection, the new entitlements are sent with the film to the different players. Inside the player, the key is decrypted in a secure module and used for the film decryption and playing. In case of an exceptional projection or for a test projection, the projection parameters are memorized and an audit trail is securely reported later to the distributor. In this architecture, the bit to bit copy piracy is fixed but not the handy-cam.

4.1.2 Fingerprinting context

This previous system based on Digital Right Management can protect digital right against the bit to bit copy piracy but not against projected digital

movie copy. A solution is to embed an information in the media that allows content owners to track the corrupted copy and identify the eventual projection room corrupted. To achieve this goal, a watermark has to be embedded in the movie theater, we call it a fingerprint in opposit to a watermark that identifies content owners. The watermark process take place in the offline computation.

Fingerprints are applied during each projection. These fingerprints do not exist in the content distribution. It would indeed be quite difficult to manage the distribution of different specimen of the content to each movie theater. The fingerprint should include identification of the theater as well as the exhibition context. According to the Identification Multimedia License Plate (IMLP) for still images and the International Standard Audiovisual Number (ISAN) for videos, these metadata should hold in 64 bits. For IMLP, 64 bits are composed by 16 bits to describe the country, 16 bits to define the trusted person, 32 bits to give an IDenfication number.

This kind of watermarking, called fingerprinting requires real-time embedding schemes. This constitutes a severe constraint given the data rate. Moreover, very low distortion on the media is tolerated as perceptual fidelity is primordial. The visual quality of pictures projected does not be affected by the deterioration brought by the watermark pattern addition. The most critical operation is the perceptual masking as it often requires complex content analysis. One could imagine preprocessing the movie before exhibition, but such approach would necessitate complementary storage and would probably lower the system flexibility.

A fingerprinting/watermarking platform is essentially based on a quickly method of watermarking embedding algorithm and, a robust and invisible spread signal, called watermark, added to the original multimedia stream. In digital cinema, quickness is the priority. Due to image size (over than 2000*1000 pixels to display) and frame rate, the algorithm have to be able to process a high flow of information.

Due to technical specifications, the watermarking algorithm has to achieve a tradeoff with quickness, visual quality and robustness against different kind of attacks.

A real time process is expected in the projection room to uncompress and personalize each projection using fingerprinting technics. This method is used to identify and eventually purchase the corrupted projection room. An offline process is expected in the provider room to guaranty authentication and digital right management. The offline protection process is

realized only one time for each media.

According to previous consideration, an evolution towards to a global scheme, including offline and real time process, with tracking (fingerprinting) and copyright protection (watermarking) processes, is presented fig(4.3):

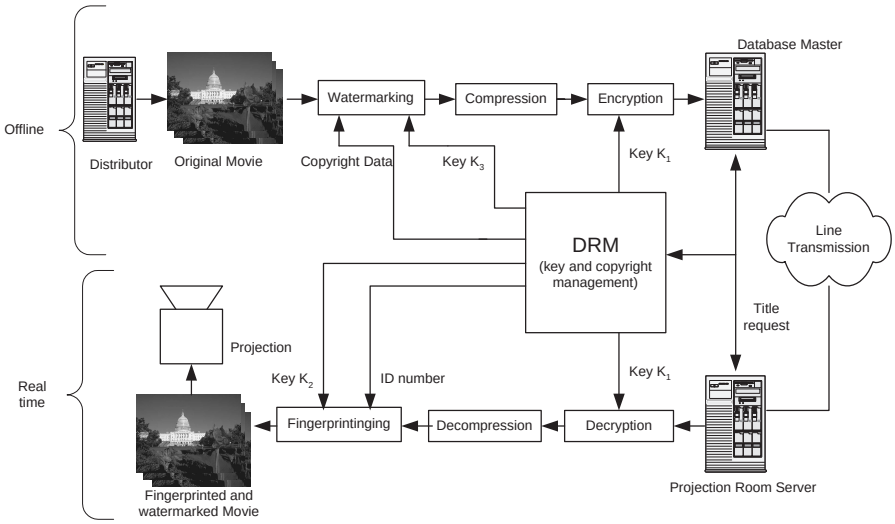

Figure 4.3: *Online and offline architecture*

4.1.3 Screen distortion and temporal distortion context

A still image watermarking algorithm presented and described in MMSP [6] highlights some interesting resistance properties against geometrical deformation and image processing attacks such as print and scan computation. Using this watermarking algorithm, some tests applied in projection room bring to the fore some deficiencies. The watermark embedded in the projected picture is not detected. Some factors can contribute to amplify this gap of no-detection:

- Geometrical deformation of the screen in the projection room.
- Difference of pixel precision between the pixels projected by the lens of the projector and pixel displayed in the screen.

In the first case, the geometrical deformation generates a lost of synchronization between the original sequence and the candidate sequence.

In the second case, digital-analog and analog-digital conversion modify the pixel intensities and colorimetry measurement. In the next experiment, we put in prominent position that our algorithm [6] is full resistant against conversion attacks but only partially against geometrical deformation. Now, we project a whole of watermarked pictures without synchronization block, with the intention of testing the geometrical deformation effects brought by the screen and pixel conversion. The pictures are projected in:

- A normal screen in a movie theater with natural geometrical deformation.
- A flat screen with no deformation.

After some screen shots applied in the two screens, the captured pictures are resized and cropped manually according to original size of the projected picture, using image-processing software. From this experiment, we extract correctly the watermark. In opposite to the synchronization block, the watermarking block insertion is efficient for cinema application.

During projection, synchronization modification in still images brings too many distortions for the watermark extraction. Temporal attacks including frame rate modification, scene removal and temporal cropping bring also too many distortion in original movie.

Watermarking as copyright tool is not necessary. Each movie is attributed to only one owner. An authority concerned can easly distinguish a movie and authenticate its owner. Watermarking is a well-known tool to authenticate a movie but it is not the only one. Digital signature is largely used in cryptography to guaranty a document authentication. Voyatziz and Pitas [4] talked about Digital signature as an efficient way to authenticate document. But they explained that this solution is not adopted in multimedia application due to the length of the output hash function bit stream. A digital signature evolution applied in images can be an alternative and a efficient way to recognize and describe a movie.

A big deficiency in the global scheme presented previously is the addition of the fingerprint pattern to the watermarked picture. Fingerprinting can be considered as a strong attack against watermarking. In the follow new global scheme, the digital signature for images does not modify the image;

it extracts only some characteristics from the picture. There is no effect for the fingerprinting and fingerprinting has no effect for the visual hash. The advanced architecture combining digital signature, DRM processes and fingerprinting technics are presented in the figure (4.4).

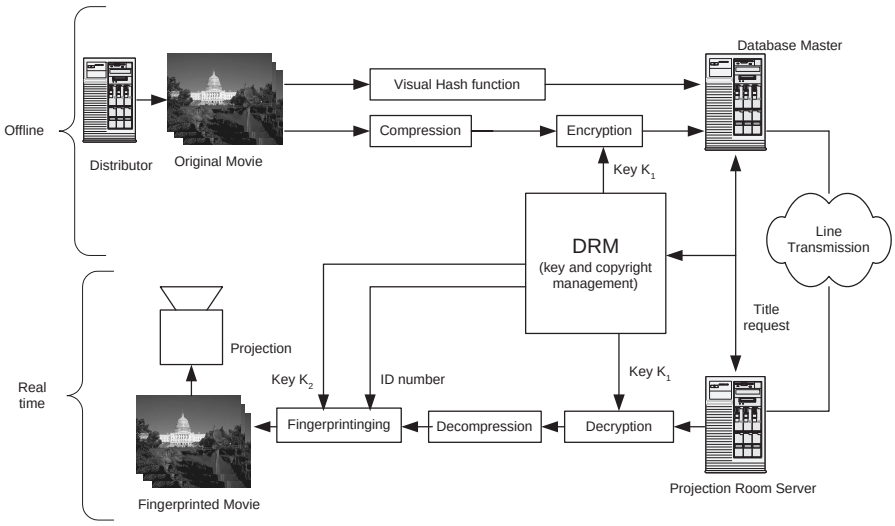

Figure 4.4: *Secure global scheme for movie*

To be effective, the fingerprint should be resistant against attempts to disable it. This requires placing the implementation within a secure device. Moreover, the efficiency of fingerprints may require to integrate embedding modules in a global protection system, with cryptography, key management and conditional access.

The objectives of this chapter are two-fold:

- To explain a watermarking algorithm and the evolution to a light algorithm according to technical specification and quickness priority.
- To present a visual hash which combined with fingerprinting, is describing as an alternative to watermarking/fingerprinting.

4.2 Watermarking

The private watermarking presented in MMSP [6] ensure a strong resistant against some attacks such as print and scan. Tracking and copyright

protection is a practical application for this watermarking algorithm.

This process is based on private keys and it is a blind architecture, it means that we do not need original image to extract watermark. The algorithm is divided in three parts: pattern generation which is the pseudo mark embedded, the psychovisual mask which is the mark weighting for the invisibility, and the synchronized block which is a template added to image source to detect geometrical deformation according to image source fig.4.5. The security of the algorithm is based on the private secret key and not on the algorithm secret (totally or partially). It is why we can say that we respect KerKhoffs laws, widely used in crypto-system.

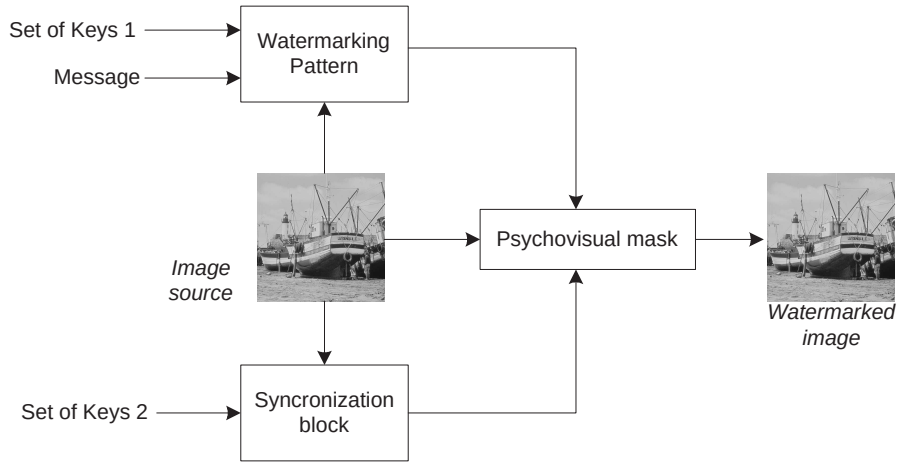

Figure 4.5: *Global watermarking scheme.*

Even if the algorithm is very efficient for still images, video tests applied in projection room show some deficiencies. The watermark embedded in the projected pictures is not always detected. The problem is due to the inefficiency of the synchronization block. In addition, this block is too slow and too complex for real-time digital cinema applications. Our algorithm is described in this section. The force and weakness are detailed.

4.2.1 Description of the light algorithm

To avoid these previously detailed problems concerning digital cinema, we propose an evolution to a lighter algorithm as detailed in Figure

4.6. We replace the synchronization block by an off-line process called RADISH [1]. This off-line process is robust hash for images and described in the previous section. RADISH property allows an efficient recovering of geometrical manipulations. An harware implementation is proposed in the next subsection. Therefore, we get a fast and robust fingerprinting scheme, but the scheme is not blind now.

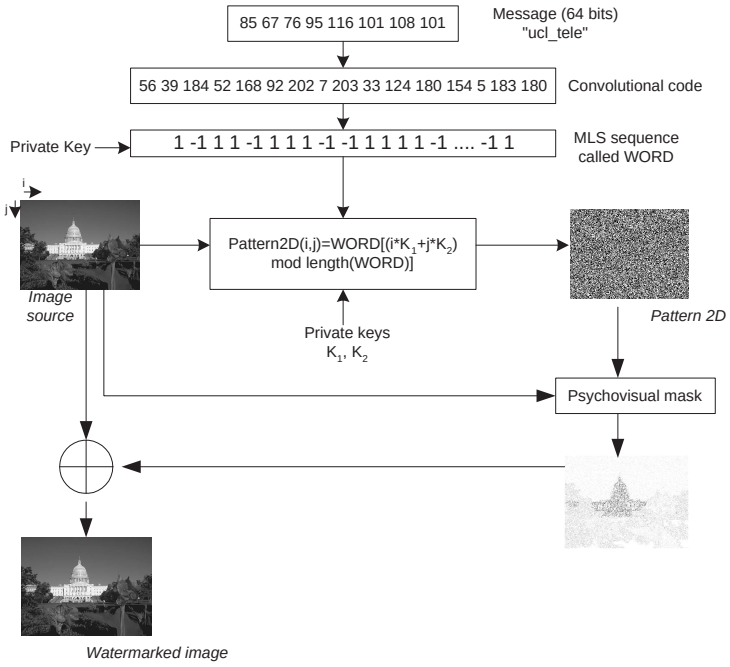

Figure 4.6: *Watermarking block scheme.*

A watermark is a sequence of real value or binary value. It depends on the signal. It is a spread signal added to original signal. In [2], an embedding process is defined as:

$$v_i' = v_i + \alpha b_i p_i$$

Where v_i' is the fingerprinted signal, v_i is the original signal, b_i the embedded message, p_i pseudo noise sequence, α is the force of the mark. In our case $\alpha.b_i.p_i$ are weighted by the psychovisual mask.

Convolutional code

An error correcting code is added on the 1-D information. The message to encode is very short (64 bits) and the possible corrupted bits are randomly spread on the payload. For these reasons, we have chosen convolutional code to encode our information. The message to modulate is two times the original message.We chose a convolutional code to encode the original message. Therefore, we extend the 64-bit original message to a 128-bit code. Soft Viterbi is used to recover the original message. Convolutional code offers real improvement for the watermarking extraction due to the redundancy of the 128-bit code. Indeed, the original message could be correctly revealed even if some errors appear in the extracted 128-bit code.

Pseudo-noise sequence

An efficient watermark is a robust mark based on redundancy, an accurate recovery method and an undetectable mark for a user without right. A MLS[1] pseudo-random sequence provides most of the previous requirements. The maximum Length shift register is a class of cyclic codes [8].

As defined in [8], an $(n, k) = (2^m - 1, m)$ linear code C is called a cyclic code if every cyclic code shift of a code vector in C is also a code vector in C. So any attacks repr'esented by a shifting in the MLS code can easly be deteced by cross-correlation with the original sequence. The generator polynomial for encoding a (n,k) cyclic code is given by:

$$g(X) = 1 + g_1 X + g_2(X) + ... + g_{n-k-1} + X_{n-k} \tag{4.1}$$

And the encoding operation can be represented as fellows fig. 4.7:
Coefficients of stages connected are well known and the implementation, using shift register is low cost. The length of this cyclic sequence is $n = 2^m - 1$, where m[2] is the number of stages. This code generates a Gaussian noise appearance and provides interesting detection properties. For secure extraction, we define a 40-bit key Key_0. This key is used as the secret seed for the generation of our MLS code. Then, we extract the first 7 bits of the MLS sequence. This value corresponds to the index of the previous convolutional code. We extract the bit corresponding to the index of the convolutional code. This bit is the first bit of the new sequence called $WORD$. And we continue until that all bits of the convolutional code are

[1] Maximum Length Shift register or m-sequence.
[2] 40 in our case.

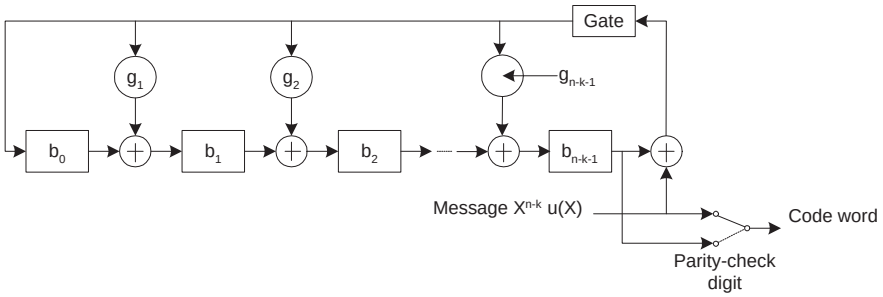

Figure 4.7: *Encoding circuit for an* (n, k) *cyclic code.*

represented 7 times or more in $WORD$. Typically, the length of $WORD$ is 2^{15}.

2-D Pattern

The goal of a watermarking 2D algorithm developed in spatial domain is to create a 2D matrix based on the message (to embed) randomly spread in the 2D matrix. The first work on this domain is developed by Tirkel [7]. Delannay proposed an evolution to this method based on MLS sequence. He creates a 2-D cyclic pattern [5], expanding our $WORD$ into a matrix. The proposed processed method is a linear computation between the image point coordinates and two 8-bit secret keys Key_1 and Key_2:

$$Pattern(i, j) = WORD[(i * Key_1 + j * Key_2) \ \ length(WORD)] \quad (4.2)$$

where (i, j) represents the image pixel coordinates. $Pattern(i, j)$ represents the way we are going to modify the (i, j) pixel intensity: if it is equal to 1 (0), the pixel intensity will be increased (decreased).
A translation or cropping operation on the captured image is equivalent to the same transformation on the 2-D pattern. It simply corresponds to a cyclic permutation of the $WORD$, that fully permits a robust extraction of the original mark:

$$Pattern(i + i_0, j + j_0) = WORD[((i + i_0) * Key_1 + (j + j_0) * Key_2) \ \ mod \ \ length(WORD)]$$

$$Pattern(i + i_0, j + j_0) = WORD[((i * Key_1 + j * Key_2) + (i_0 * Key_1 + j_0 * Key_2)) \ \ mod \ \ length(WORD)]$$

Psychovisual Mask

An efficient watermarking algorithm has to combine invisibility and robustness. Robustness is guaranteed by the redundancy of the insertion scheme and invisibility by a psycho visual mask. The purpose of this mask is to modify the watermark according to the image energy to make it invisible. Our psycho visual mask is based on two principles:

- **Image activity**: or local mean that compares medium pixel intensity inside its environment (its neighbors). In fact, if we increase or decrease some pixel intensity in high contrast region, we cannot detect a difference between two pixels.
- **Importance of pixel intensity**: A pixel modification is more visible in black intensities than in white intensities. This property is well defined and explained by the Weber-Fechner law.

The visual increment threshold is defined as the amount of light ΔB_T necessary to add to a visual field of intensity B to become visible. This threshold can be approximated by picewise functions and the minimum amount can be computed as

low intensities region:
$\Delta B_T = \sqrt{x_1 x_2} * \beta * \sqrt{B} * \left(\frac{\Delta B}{B}\right)_{max}$ for $B \leq x_1$
De Vries-Rose region:
$\Delta B_T = K_2 * \sqrt{B}$ for $x_1 \leq B \leq x_2$
Weber region:
$\Delta B_T = K_1 * B$ for $x_2 \leq B \leq x_3$
Saturation region:
$\Delta B_T = K_3 * B^2$ for $B \geq x_3$

x_1, x_2, x_3 determine the boundaries of the different regions and K_1, K_2, K_3 are constants of proportionality.
According to the previous equations, the Weber Fechner figure is 4.8:
These two principles of perceptual model compute a threshold. Over this threshold the watermark is visible, and under this threshold the watermark is supposed to be unvisible.

This watermarking algorithm is almost fully image processing resistant. Due to cyclic properties of the pattern, the fingerprint algorithm does not need to another synchronization process to be resistant against cropping/translation.

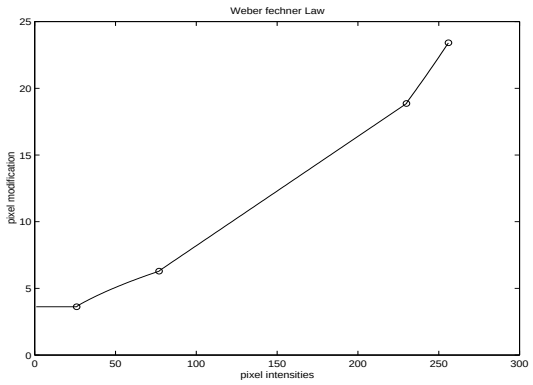

Figure 4.8: *Weber Fechner law with a saturation of 1*

4.2.2 Watermarking detection performance

To evaluate the robustness and performance of our watermarking method, we experiment on 40 real-world images taken from the USC-SIPI database [13].

For each of the 40 images of the data base, we embed a message with a range of six α (0.02,0.04,0.08,0.1,0.15,0.2). To evaluate the image processing degradation due to fingerprinting insertion, we calculate the PSNR mean for each modified images according to the force α. Figure 4.9 shows the resulting PSNRs. An empirical value of $36db$ is a good PSNR threshold to achieve a not too visible added template.

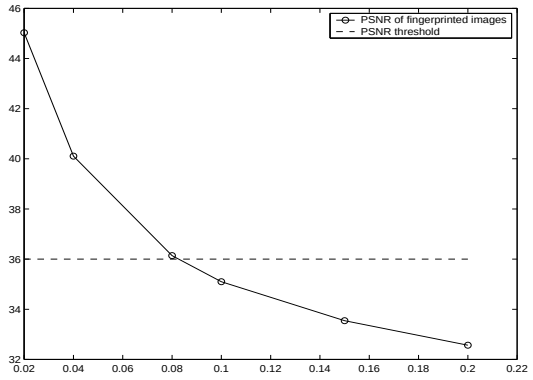

Figure 4.9: *PSNR mean of 40 fingerprinted images regarding the force of the fingerprint*

For each fingerprinted image, we consider 4 image processing attacks, generating $40 * 6 * 4 = 960$ images, named processed images. The attacks are filtering (3x3 Gaussian filtering with standard deviation of 0.5), compression (JPEG compression with 80% and 60% quality factor), and noise (salt and pepper).

The robustness results are given by Tables 4.1, 4.2, 4.3, 4.4. The term *Extracted* represents the number of processed images where the mark is correctly detected and extracted, *Only detected* represents the number of processed images where the the mark is correctly detected but too many bits are lost to compute a correct extraction process and *No detected* represents the number of processed images where the mark is not detected and no extracted.

Force	0.02	0.04	0.08	0.1	0.15	0.2
Extracted	29	39	40	40	40	40
Only detected	4	1	0	0	0	0
No detected	7	0	0	0	0	0

Table 4.1: *Gaussian attack.*

Force	0.02	0.04	0.08	0.1	0.15	0.2
Extracted	27	40	40	40	40	40
Only detected	4	0	0	0	0	0
No detected	9	0	0	0	0	0

Table 4.2: *Noise attack.*

Force	0.02	0.04	0.08	0.1	0.15	0.2
Extracted	26	37	40	40	40	40
Only detected	3	3	0	0	0	0
No detected	11	0	0	0	0	0

Table 4.3: *Jpeg attack, quality=60.*

Attacks and PSNR figures provide a good empirical value of the force, closed to 0.06, to obtain a good tradeoff robustness/visibility of the fingerprint. Using this empirical value, we obtain excellent results (closed to previous one) for extraction of the mark in a projection room.

Force	0.02	0.04	0.08	0.1	0.15	0.2
Extracted	30	39	40	40	40	40
Only detected	3	1	0	0	0	0
No detected	7	0	0	0	0	0

Table 4.4: *Jpeg attack, quality=80.*

4.2.3 Hardware implementation

This part was developed in collaboration with Gael Rouvroy for TACTILS project.

Detailed block implementation

As previously mentioned, the fingerprinting insertion process needs to be hardware implemented to deal with the high bit rate of digital cinema. Figure 4.10 illustrates the global FPGA architecture of our fingerprinting scheme. We propose a complete unrolled and pipelined design to ensure the data processing throughput of digital cinema. We adapt the design to support 2048×1024 frames with a dataflow of 24 images per second.

The first watermarking step is to compute the convolutional code from the 64-bit original mark and the MLS sequence until the $WORD$ sequence is completely generated. $WORD(n)$ means the n^{th} bit of the sequence. The proposed design allows us to change Key_0 and the mark to embed for each new frame. About 100,000 clock cycles ($\simeq 1$ ms) are required to generate a new $WORD$ from a new key or a new original message. Therefore, it is not a judicious choice to change these inputs for every new frame.
Once the WORD is generated, we start to compute the 2-D pattern assuming that the image pixels are received in a one by one serial way (cycle by cycle) in the YUV domain, line by line. We first receive $(0, 0)$, then $(1, 0)$, $(2, 0)$, ... , $(0, 1)$, $(1, 1)$ and so on. Every cycle a new pixel (i, j) is processed and a new $Pattern(i, j)$ is computed. Secret keys Key_1 and Key_2 can be modified for every new frame without any dead cycles, which it is not the case for Key_0. Nevertheless, changing only Key_1 and Key_2 regularly is not secured enough. It is better to change sometimes all the secret keys between two frames. Figure 4.11 shows the architecture concerning the calculation of Equation 4.2. The 32.768 bits of the $WORD$ sequence are stored inside two separate RAM blocks[3].

[3]Virtex-II FPGAs have only internal 18-Kbit RAM blocks.

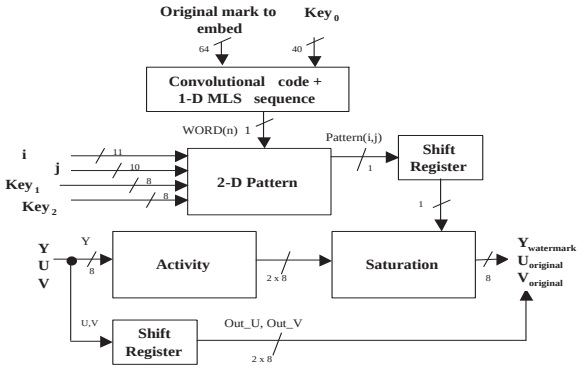

Figure 4.10: *The global fingerprinting architecture.*

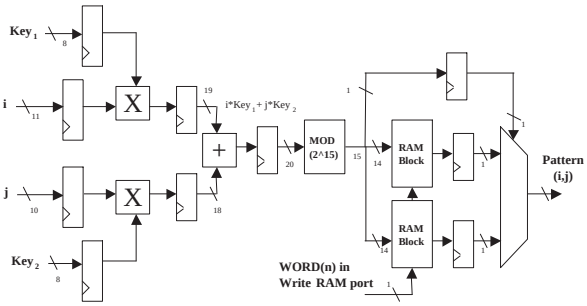

Figure 4.11: *The 2-D pattern block.*

In parallel with the 2-D pattern calculation for the (i, j) pixel, we compute the first part of the psychovisual mask based on the local activity of the Y component. The activity of (i, j) pixel is the difference between the intensity of the (i, j) pixel and the mean intensity of the close pixels[4].

Figure 4.12 illustrates the activity calculation of one pixel. In addition, the block computes the absolute value of the activity value and returns Out_Y.

Figure 4.13 completes the psychovisual mask applying the Weber-Fechner law (stored in a ROM). In addition, the saturation block inserts the mark thanks to the $Pattern(i, j)$ bit and ensures that the modified intensity is between 0 and 255.

[4]In total, there are 8 pixels involved for the mean calculation, the direct 8 neighbors of the (i, j) pixel.

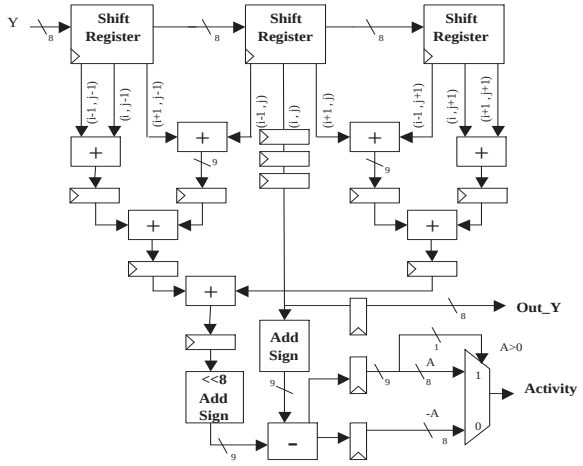

Figure 4.12: *The Activity block.*

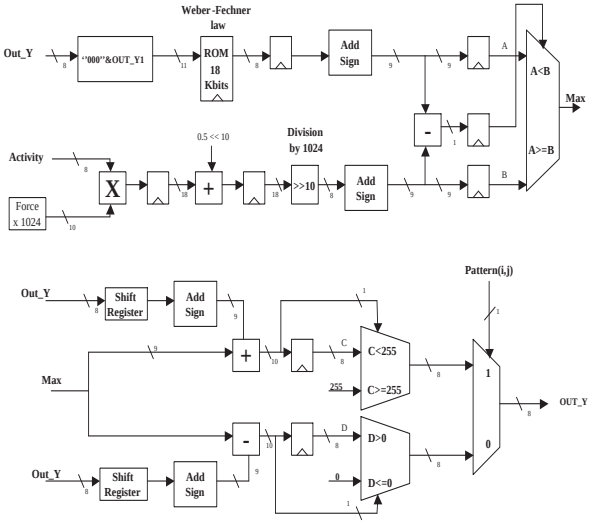

Figure 4.13: *The Saturation block.*

Implementation results

The synthesis of our complete fingerprinting design was done using Synplify Pro 7.2 from Synplicity. The placing and routing were done using Xilinx ISE 6.1.i. The final results are given in Table 4.5 for a Xilinx Virtex-II

FPGA (XC2V500-6). We detail the resources used according to two frame sizes (2048×1024 and 1024×768).

Frame size	1024×768	2048×1024
LUTs used	1670	2727
Registers used	759	761
Slices used	**1065**	**1617**
RAM blocks used	4	4
Multipliers used	4	4
Max. Output every (cycles)	1	1
Frequency (MHz)	143.9	143.9
Max. Throughput (Mbps)	**3454**	**3454**
Nbr Images/seconde	**182.98**	**68.62**

Table 4.5: *Final results of our complete fingerprinting scheme.*

Our design is able to fingerprint all 2048×1024 video frames even if we need to project at a dataflow of 48 images per seconde. We fully meet the digital cinema requirements.
This light fingerprinting algorithm is perfectly adapted to digital cinema and tracking of media after its projection/diffusion. We evaluate the performance of the complete watermarking scheme and we tune the algorithm parameters to reach a good tradeoff robustness/visibility. To avoid the slowness of the software implementations, we propose a complete FPGA implementation of the fingerprinting insertion. The resulting design is able to deal with 2048×1024 video frames at a throughput of about 68 images/sec. This solution completely meet the digital cinema requirements for a very reasonable hardware cost.

4.3 Digital signature real time process

As detailed in previous section, we define the RADIal Hashing (RADISH) feature vector as follows. Let $\Gamma(\phi)$ denote the set of pixels (x, y) on the projection line corresponding to a given angle ϕ. Let (x', y') denote the coordinates of the central pixel. According to figure 2.5, $(x, y) \in \Gamma(\phi)$ if and only if:

$$-\frac{1}{2} \leq (x - x').cos\phi + (y - y').sin\phi \leq \frac{1}{2} \tag{4.3}$$

Let $I(x, y)$ denote the luminance value of the pixel (x, y), the RADISH

feature vector $R[\phi]$, $0 \leq \phi < 180$, is then defined by:

$$R[\phi] = \frac{\sum_{(x,y)\in\Gamma(\phi)} I^2(x,y)}{\#\Gamma(\phi)} - \Big(\frac{\sum_{(x,y)\in\Gamma(\phi)} I(x,y)}{\#\Gamma(\phi)}\Big)^2 \tag{4.4}$$

A software implementation of the RADISH transform allowed us to evaluate its time-performances for different image sizes, from 512 $\times$ 512 to 4000 $\times$ 2000 pixels. We observed that the memory requirements of the algorithm does not significantly increase with the image size whereas its computational requirements linearly depends on it. Therefore, the throughput of the RADISH transform is independent of the image size. On a AMD Athlon, XP 1800 with 512 MBRAM, we found 2^{18} pixels/sec. In practice, the RADISH of a 512 $\times$ 512 image is computed in 1 second whereas a 4000 $\times$ 2000 image will need about 30 seconds. In the next section, we investigate the relevance of an hardware coprocessor for computing the RADISH transform.

4.3.1 Hardware implementations

This part was developed in collaboration with Francois Xavier Standaert for TACTILS project.
For hardware implementations, we assume that the pixels are received in a one by one serial way. That is, we first receive $(0,0)$, then $(1,0)$, $(2,0)$, ... , and finally (X,Y). Let the different resources needed to implement the RADISH be divided as follows:

R1. Resources for computing condition (4.3).

R2. Squaring multipliers for the left term of equation (4.4).

R3. Averaging adders for the sums of equation (4.4) (i.e. $\sum I$, $\sum I^2$, $\#\Gamma(\phi)$).

R4. Registers or memory to store the averages of equation (4.4).

From these resources, we may build different implementation scenarios:

Serial-serial

This scenario refers to the situation where we compute condition (4.3) for one pixel (x,y) and one angle ϕ in one clock cycle. The resulting design has the lowest possible area requirements and a low throughput. As only one pixel is managed by clock cycle, only one multiplier is necessary for

the squaring operation of equation (4.4). For the averaging, we need a memory with 180*3 addresses and three adders[5] to compute the different sums of equation (4.4). In terms of throughput, if the work frequency is f, we expect to have a throughput of about $\frac{f}{180}$ pixels/sec. It should not significantly improve software performances.

Serial-parallel

This scenario refers to the situation where we compute condition (4.3) for one pixel and all the 180 angles ϕ in one clock cycle. The resulting design needs to implement condition (4.3) 180 times in parallel. However, as only one pixel is managed by clock cycle, we still only need one multiplier. For the averaging, we need the same number of memory addresses, but they have to be accessed in parallel so that we need 180*3 registers. 180 times more adders[6] are also necessary. As a consequence, the expected throughput becomes f pixels/sec.

Parallel-parallel

This scenario refers to the situation where we compute condition (4.3) for several (n) pixels and all the 180 angles ϕ in one clock cycle. Compared with the previous scenario, it means that we have to multiply by n the number of times we compute equation (4.3) and the number of multipliers for the squaring operations of equation (4.4). As it is then possible that several pixels influences the same angle in one clock cycle, we need either multi-operand adders or additional FIFO memories for the averaging of equation (4.4). The resulting design has an increased complexity and an expected throughput of $f.n$ pixels/sec.

Comparisons

The estimations for the different implementation scenarios are summarized in Table 4.6, where the symbol * indicates that additional resources are needed to deal with multiple pixels. (M) means that the storage is implemented in a single access memory and (R) means that the storage uses registers.

[5]The factor three is due to the three sums that we have to compute, that is $\sum I^2$, $\sum I$ and $\#\Gamma$.

[6]This is due to the fact that some pixels are in included in several angle lines, e.g. (x', y') is included in all of them (although in practice most pixels influence only one angle line).

	S-S	S-P	P-P
#R1	1	180	180.n
#R2	1	1	n
#R3	3	180.3	180.3*
#R4	180.3 (M)	180.3 (R)	180.3*(R)
Throughput	$\frac{f}{180}$	f	$n.f$

Table 4.6: *Area and throughput (pixels/sec) estimations for different implementation scenari.*

	Nbr LUTs	Nbr Flip flops	Nbr slices
Cond.(4.3)	84	19	47

Table 4.7: *Implementation results for condition (4.3).*

Based on these estimations, the serial-parallel scenario appears to be an interesting combination of circuit size, throughput and simplicity. In the next section, we investigate its efficient implementation for a 512 x 512 pixels image with 8-bit luminance.

4.3.2 Efficient implementation of a serial-parallel architecture

Computation of condition (4.3)

Let X (resp. Y) be the number of pixels by line (resp. column) as suggested in Figure 2.5. For every $(x, y), \phi$, we want to compute the following condition:

$$-\frac{1}{2} \leq (x - x').cos\phi + (y - y').sin\phi \leq \frac{1}{2} \tag{4.5}$$

As the pixels are provided in a one by one serial way, we observe that $(x - x').cos\phi + (y - y').sin\phi$ may be modified in only three different manners:

1. Initially, it is set to: $init_\phi = -x'.cos\phi - y'.sin\phi$.
2. For a new pixel, we add a constant value $a_\phi = cos\phi$.
3. For a new line, we add a constant value $b_\phi = -(X - 1).cos\phi + sin\phi$.

If we store the values for a_ϕ, b_ϕ and $init_\phi$ in a memory, it is therefore possible to compute condition (4.3) with only one adder, one register and two comparisons (These comparisons are actually implemented as one adder and one substractor), as it is shown in Figure 4.14. The implementation results are in Table 4.7.

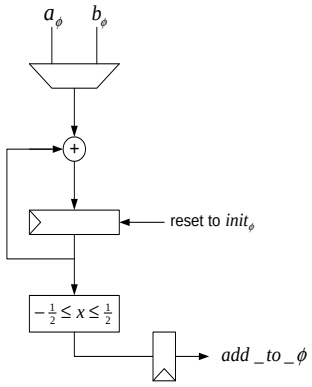

Figure 4.14: *Computation of condition (4.3).*

	#LUTs	#Flip flops	Nbr slices
Av. circuit (4.3)	94	54	54

Table 4.8: *Implementation results for the Av. of (4.4).*

Computation of the RADISH (4.4)

Thanks to the add_to_ϕ signals, we can compute the different sums of equation (4.4) for a fixed ϕ with three adders, three registers and three multiplexors. We decided to implement only $\sum_{(x,y)\in\Gamma(\phi)} I^2(x,y)$, $\sum_{(x,y)\in\Gamma(\phi)} I(x,y)$ and $\#\Gamma(\phi)$ in hardware. The final squaring, division and substraction can more efficiently be implemented in software. An efficient solution to do it would be to use the embedded processors available inside some recent FPGAs. Anyway, these operations are not critical in software and do not involves a need for hardware implementation. The design is represented in Figure 4.15 and its implementation results are in Table 4.8. Remark the use of output multiplexors that allow to chain the different averaging registers in order to have a serial output of the 180.3 coefficients os the RADISH.

Complete implementation

The complete implementation uses the previous cells 180 times in parallel. An additional multiplier is used to compute the current I^2. Some additional logic is required for the control signals. After implementation within a XILINX Virtex2-6000, we obtain the results of Table 4.9 for a 512

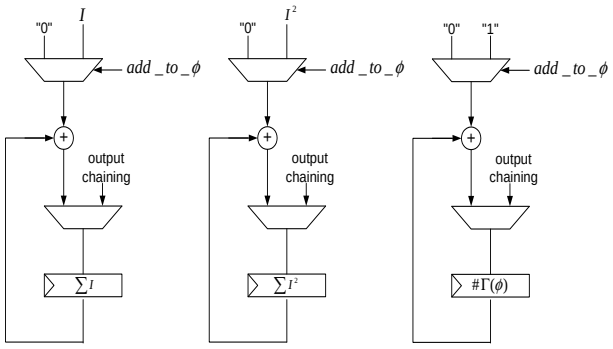

Figure 4.15: *Averaging circuit (4.4).*

	Nbr LUTs	Nbr Flip flops	Nbr slices
RADISH	28979	13174	16846

	Frequency	Throughput
RADISH	75 Mhz	75.10^6 pixels/sec

Table 4.9: *Complete implementation results.*

$\times$ 512 image.

For larger image sizes, the work frequency is only very slightly reduced (e.g. 74 Mhz for 4000 $\times$ 2000 images). It is due to the slight modification in the averaging adders size. As a consequence, we may assume that, as in the case of SW implementations, the throughput is independent of the image size.

4.4 Fingerprinting/video digest for movie authentication and tracking

The video hash digest is robust and invariant against geometrical deformation, video compression such as Divx, xvid,... and temporal distortion. Due to mathematical properties detailed in previous chapter, a comparison between original video digest and an eventual corrupted video digest is possible. An architecture is proposed in the figure 4.16.

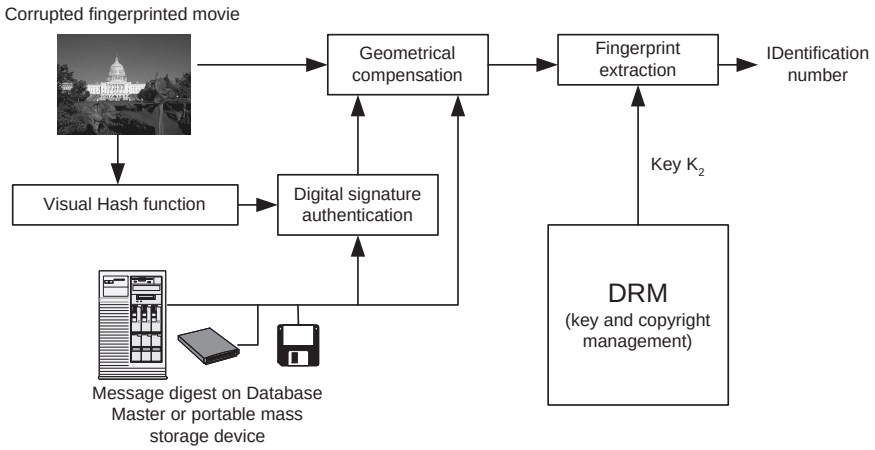

Figure 4.16: *architecture for a movie authentication and tracking process*

All message digests from all digital movies are stored in a storage portable device such as CD Rom or DVD Rom, or in a secure database separated to the movies database. For authentication, each message digest from corrupted copy are computed, using a robust soft algorithm for video, and are compared to message digest localized in the device storage. The movie is recognized and the preferences associated to the movie are automaticly selected to extract the copyright and the ID number, using fingerprint extraction process.

Bibliography

[1] J.-F. Delaigle, D. Delannay, B. Macq, J.M. Mas Ribs, J. Nivart, "Integrated fingerprinting in secure digital cinema projection" , SPIE 47th Annual Meeting - Application of Digital Image Processing XXIV, San Diego, USA, July 29 - August 3, 2001, Proc. Vol. 4472, pp. 167-174

[2] S.P. Mohanty, "Digital watermarking: a tutorial review", 1999, http://www.csee.usf.edu/ smohanty/research/Reports/WMSurvey1999Mohanty.pdf.

[3] F. Hartung and B. Girod, "Copyright Protection in Video Delivery Networks by Watermarking of precompressed Video", *ECMAST*, 1997, p423-436, Tokyo, Japan.

[4] G. Voyatziz and I. Pitas, "The Use of Watermarks in the Production of Digital Multimedia Products", Proceedings of IEEE Special Issue Identification and Protection of Multimedia, july 1999, pp.1197-1207.

[5] D. DELANNAY and B. MACQ, "Generalized 2-D cyclic patterns for secret watermark generation", ICIP '00.

[6] F. Lefebvre, D. Gueluy, D. Delannay and B. Macq,"A print and scan optimized watermarking scheme", *MMSP* 2001, p511-516, Cannes, France.

[7] A.Z. Tirkel, R.G. van Schyndel, and C.F.Osborne, "Digital watermark," in International Conference on Image Processing (ICIP), 1994, vol. 2, pp. 8690.

[8] S. Lin, D. J. Costello Jr., "Error Control Coding: Fundamentals and Applications", pp. 85, Prentice-Hall.

[9] Xilinx: "Virtex2 Field Programmable Gate Arrays Data Sheet", http://www.xilinx.com.

5

Conclusions and perspectives

Due to geometrical deformations and voluntary attacks aimed to delete the watermark, watermarking algorithms suffer from some troubles. The fingerprinting algorithm described in the last chapter is efficient for print and scan applications but deficient for digital movie. The projection room test shows that each watermarking scheme is designed for a specific scenario and a watermarking scheme is not suited to every application.

Usually, to be quick compression resistant, watermark block scheme is designed in the same watermark domain insertion as compression domain. The algorithm benefits from domain compression properties. The compression operation is integrated into watermark block scheme and is not considered as an attack. For this reason, most of watermarking algorithms design their insertion model in Discrete Cosinus Transform to be fully resistant against JPEG (image compression), MPEG (video compression) and Divx (video compression). The psychovisual mask valid in space domain [4] is transposed in transform domain using the different transform domain properties. Zao [3] modifies medium frequencies in the 8x8 DCT block, Barni [2] weights wavelet coefficients with Lewis-Knowles psychovisual mask model.

JPEG2000 based on wavelet domain transform, is the new image compression standard and JPEG 2000 Secured (JPSEC) is standardizing security tools in order to guarantee secure transmission, protection of contents (IPR), and protection of technologies (IP). Watermarking is an essential part of JPSEC and has a growing influence in intellectual properties protection. DRM methods such as encryption are included in JPSEC. An algo-

ritm developed in order to be optimized in a domain transform is image file format dependent and fragile for some kinds of attacks. Its resistance is only based on domain transform resistance against compression attacks. But in term of security, embedding an information directly in the same domain transform as the image compression domain is better than spatial domain insertion. The image information is never in clear access. The algorithm developed in ASPIS and its hardware design highlights interesting tradeoffs robustness/quickness/invisibility. Currently we are working in wavelet domain adaptation of this spatial domain algorithm. The first results are conclusive. There was a number of proposals to hide data information in JPEG2000 [6, 7]. A video adapation for MJPEG (Motion JPEG) is studied using representative frame selection (last chapter) for information insertion in video bit stream.

In TACTILS, devoted to the security aspects of digital cinema, Francois Xavier Standaert, Gael Rouvroy and Frederic Lefebvre are designing an hardware advanced architecture for Digital Right Management. Usually each block is considered independently: compression , encryption and watermarking. In this project, all blocks are grouped into one FPGA. Hardware design allows high data flow applications and a high level of security. A digital cinema application is expected as soon as possible.

The new message digest defined in this thesis identifies the image content. Thanks to this property, robust hashing can be introduced in pattern recognition and provide a solution for content authentication and indexing. DSA and ECDSA are largely used in crypto-systems, their designs are based on hash functions to obtain a summary plain text called message digest. The image message digest computed by robust image hashing and combined with a cypher algorithm can also be integrated in digital signature methods, and used in integrity and authentication purposes.

An interesting watermarking and image hashing perspective should be the creation of media content dependent payloads. Copy attack [5] is a classical attack to copy digital right of a media. The watermark is extracted, copied and embedded in a new media. The new created media is now protected with the same digital right as the other one. A watermark with a content dependent payload will be resistant against the copy attack. An image authentication architecture, combining watermarking and visual hash is given in the fig.5.1.
This proposal fig.5.1 is also resistant against intentional attacks such as collusion. With this method, each message embedded is content and keys

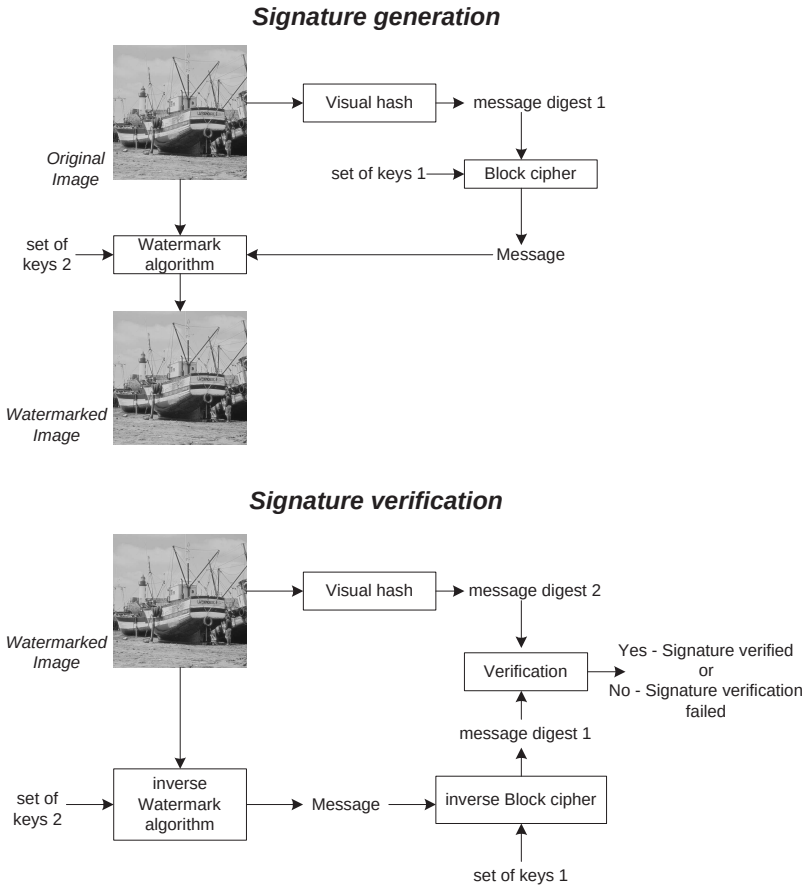

Figure 5.1: *Image signature generation and image signature verification*

(user) dependent. The watermark is different for each image and for each user. It is impossible to generate an un-watermarked copy of the image by averaging all the authorized recipients of the image. This architecture is collusion resistant.

Evenif this visual hash provides interesting properties and robustness against many attacks, it suffers from defficiencies against a large cropping and small pixel modification in the image. Some pixels can be modified in the image without changing message digest.

Bibliography

[1] F. LEFEBVRE, D. GUELUY, D. DELANNAY and B. MACQ,"A print and scan optimized watermarking scheme", *MMSP* 2001, p511-516, Cannes, France.

[2] M. BARNI, F. BARTOLINI, V. CAPELLINI, A. LIPPI and A. PIVA, "A DWT-based technique for spatio-frequency masking of digital signatures", Proceeding of SPIE vol. 3657, Electonic Imaging '99, San Jose, CA, January 1999.

[3] J. ZHAO, and E. KOCH, "Embedding robust labels into images for copyright protection",In: Proc. of the Int. Congress on Intellectual Property Rights for Specialized Information, Knowledge and New Technologies, Vienna, August 1995.

[4] A.S. Lewis and G. Knowles, "Image Compression Using the 2-D Wavelet Transform", IEEE Transactions on Image Processing, Vol. 1, No. 2,pp. 244-250, April 1992

[5] J. Fridrich, and M. Goljan, "Robust hash functions for digital watermarking," *ITTC 2000*, Las Vegas, USA, 2000.

[6] N.Thomos, N.V.Boulgouris, E.Kokkinou and M.G.Strintzis, "Efficient Data Hiding in JPEG2000 Images Using Sequential Decoding of Convolution", Proceedings of the International Conference on Digital Signal Processing, DSP 02, Santorini, Greece, july 2002

[7] P. Meerwald, "Quantization Watermarking in the JPEG 2000 Coding Pipeline", International Federation for Information Processing, Conference on Communications and Multimedia Security, Darmstadt, Germany, May 2001

A

Publications

1. "A print and scan optimized watermarking scheme", F. Lefebvre, D. Gueluy, D. Delannay and B. Macq, IEEE MultiMedia Signal Processing 2001, p511-516, Cannes, France

2. " Hardware Implementation of a Fingerprinting Algorithm Suited for Digital Cinema", G. Rouvroy, F. Lefebvre, F.-X. Standaert, B. Macq, J.-J. Quisquater, J.-D. Legat, EUropean Signal Processing COnference 2004

3. "RASH:RAdon Soft hash algorithm", F. Lefebvre, B. Macq, EUropean Signal Processing COnference 2002, Toulouse, France

4. "AGADDIS: Authentication and Geometrical Attacks Detection for Digital Image Signature", F. Lefebvre, B. Macq, Information Theory 2002 Benelux,p171-178, Louvain La Neuve, Belgium

5. "A Robust soft hash algorithm for digital image signature", F. Lefebvre, J. Czyz, B. Macq, IEEE International Conference on Image processing 2003, Barcelona, Spain

6. "A video digest based on the robust hashing of representative frames", F. Lefebvre, C. DeRoover, C. De Vleeschouwer and B. Macq, Manuscript submitted on IEEE Special Issue on Multimedia Security

7. "An invariant soft video digest", F. Lefebvre, C. DeRoover, C. De Vleeschouwer and B. Macq, Manuscript submitted on IEEE International Conference on Image processing 2004

8. ”Une fonction de hachage robuste et invariante pour image”,
F. Lefebvre, B. Macq, Medianet 20002, edition Hermes, p221-230, Sousse, Tunisie