



**HAL**  
open science

# Generation and analysis of graphical codes using textured patterns for printed document authentication

Iuliia Tkachenko

► **To cite this version:**

Iuliia Tkachenko. Generation and analysis of graphical codes using textured patterns for printed document authentication. Signal and Image processing. Université de Montpellier, 2015. English. NNT: . tel-01283584v1

**HAL Id: tel-01283584**

**<https://theses.hal.science/tel-01283584v1>**

Submitted on 5 Mar 2016 (v1), last revised 25 Jan 2019 (v2)

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

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



# UNIVERSITÉ DE MONTPELLIER

## Thèse

présentée au **Laboratoire d'Informatique de Robotique et de Microélectronique de Montpellier** pour obtenir le diplôme de doctorat

*Spécialité Doctorale* : **Informatique**  
*École Doctorale* : **Information, Structures, Systèmes**

---

## Generation and analysis of graphical codes using textured patterns for printed document authentication

---

par

**Iuliia TKACHENKO**

Soutenance prévue pour le 14 décembre 2015, devant le jury composé de :

**Co-directeur de thèse :**

M. William PUECH, Professeur des Universités ..... LIRMM, Université de Montpellier

M. Olivier STRAUSS, Maître de Conférences HDR ..... LIRMM, Université de Montpellier

**Rapporteurs :**

M. Patrick BAS, Directeur de Recherche CNRS ..... CRISAL, Lille

M. Jean-Marie MOUREAUX, Professeur des Universités ..... CRAN, Université de Lorraine

**Examineurs :**

M. Atilla BASKURT, Professeur des Universités ..... LIRIS, Lyon

M. François CAYRE, Maître de Conférences ..... GIPSA-Lab, Grenoble

**Invité :**

M. Christophe DESTRUEL, Directeur Scientifique .... Authentication Industries, Montpellier



# Acknowledgements

Completing the PhD and writing this thesis was an amazing journey that would not have been possible without the encouragement and support of my colleagues and my family. This PhD project has opened for me both research and industrial worlds. Working on this research project was a challenging and interesting adventure.

My greatest appreciation and gratitude goes to my co-supervisor, professor William PUECH, head of ICAR (Image & Interaction) project, for his trust and belief in me. I would like to thank him for useful advice and knowledge that he transferred to me during this three year period as well as for deadlines he gave me in order to boost my research results. The environment he created at the research team is probably the best a PhD student could ever hope for.

My gratitude extends to my second co-supervisor, doctor Olivier STRAUSS, for his time and patience during my blocking periods of research. I would like to thank him for his explanations and guideline during my first redaction experiences. Additionally, I would like to thank him for trust in me during my first teaching experience with master students.

I am especially grateful to Christophe DESTRUEL who supports me in the heaviest moments of this important period of my life. I would like to thank you for your encouragements and motivations as well as for your help not only in research but also in workday life.

During my PhD at ICAR team I have had the pleasure of interacting with a number of great colleagues which have become precious friends. Thank you for your friendship and aspiration that offered me the opportunity to fully integrate in our research team and in french society. I had the pleasure to meet, work and socialize with you. A special thanks goes to Vincent ITIER for being such a great office mate. Thank you for correcting thousands of errors in my french e-mails.

I wish to thank the direction of AUTHENTICATION INDUSTRIES (AI), and especially Christian GUICHARD, for their trust and financial support of my PhD research. My work in AI not only allowed me to discover the life of start-up company but also to meet a lot of very skilled professionals. Thank you my dear colleagues for our interesting discussions and active workdays.

I would like to thank again all my colleagues. It was a pleasure and a great honor to work with you!

A special thank goes to my PhD thesis referees, Patrick BAS and Jean-Marie MOUREAUX,



who accepted to read and evaluate my research work. Your remarks and suggestions have helped me to improve my manuscript.

Finally, I would like to express my deepest gratitude to my family and my friends. They all stood by me and shared with me both the great and the hard moments of my life. Thank you for your believing in me!

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	General context . . . . .	1
1.2	Problem description . . . . .	2
1.3	Dissertation structure . . . . .	3
<b>I</b>	<b>Related work</b>	<b>5</b>
<b>2</b>	<b>Printed document protection techniques</b>	<b>7</b>
2.1	Introduction . . . . .	7
2.2	Optical protection . . . . .	8
2.2.1	Technological protection . . . . .	8
2.2.2	Chemical protection . . . . .	9
2.2.3	Security printing protection . . . . .	10
2.3	Printing security patterns for optical protection . . . . .	12
2.3.1	Digital watermark . . . . .	12
2.3.2	Steganography . . . . .	14
2.3.3	Image encryption . . . . .	16
2.3.4	Visual cryptography . . . . .	16
2.3.5	Halftone-dot orientation data embedding . . . . .	18
2.4	Security techniques for document authentication . . . . .	18
2.4.1	Document hash function . . . . .	19
2.4.2	Printing techniques for document authentication . . . . .	20
2.5	Conclusions . . . . .	21
<b>3</b>	<b>P&amp;S modeling and image processing</b>	<b>23</b>
3.1	Introduction . . . . .	23
3.2	Printing process . . . . .	24
3.2.1	Electrophotographic printer architecture . . . . .	24
3.2.2	Inkjet printer architecture . . . . .	25
3.2.3	Printer characteristics . . . . .	26
3.2.4	Paper characteristics . . . . .	29
3.3	Scanning process . . . . .	29
3.3.1	Flatbed scanner architecture . . . . .	30
3.3.2	Scanner characteristics . . . . .	31
3.4	P&S process modeling . . . . .	32
3.5	Conclusions . . . . .	34

<b>4</b>	<b>Rich barcodes</b>	<b>35</b>
4.1	Introduction . . . . .	35
4.2	Standard barcodes . . . . .	35
4.2.1	Barcode evolution . . . . .	36
4.2.2	QR code structure . . . . .	37
4.3	User friendly barcodes . . . . .	39
4.4	High storage capacity barcodes . . . . .	40
4.4.1	Black-and-white barcodes . . . . .	41
4.4.2	Colored barcodes . . . . .	42
4.5	Data hidden barcodes . . . . .	48
4.6	Authentication graphical codes . . . . .	50
4.6.1	Authentication system . . . . .	50
4.6.2	Authentication scenario . . . . .	52
4.7	Conclusions . . . . .	55
<b>II</b>	<b>Contributions</b>	<b>57</b>
<b>5</b>	<b>Textured patterns used for document authentication</b>	<b>59</b>
5.1	Introduction . . . . .	59
5.2	Textured pattern description . . . . .	60
5.2.1	Experimental analysis of characteristics . . . . .	60
5.2.2	Mean opinion score of human perception . . . . .	64
5.2.3	Textured pattern combinations . . . . .	66
5.2.4	Textured pattern characteristics . . . . .	68
5.3	Textured image generation . . . . .	68
5.4	Pattern detection after P&S process . . . . .	70
5.4.1	Representative candidates . . . . .	70
5.4.2	Pattern comparison measures . . . . .	71
5.4.3	Clustering methods . . . . .	72
5.5	Experimental results . . . . .	76
5.6	Conclusions . . . . .	79
<b>6</b>	<b>High density barcodes</b>	<b>81</b>
6.1	Introduction . . . . .	81
6.2	High density QR code . . . . .	82
6.2.1	Reading process . . . . .	82
6.2.2	Binarization methods . . . . .	83
6.3	Proposed centrality bias measure . . . . .	85
6.4	Binarization using WMSE measure . . . . .	88
6.4.1	Classification binarization method . . . . .	88
6.4.2	Global thresholding methods vs. WMSE classification . . . . .	90
6.4.3	Weighted global binarization methods . . . . .	90
6.5	Experimental results . . . . .	92
6.5.1	Database description . . . . .	92
6.5.2	Detection results . . . . .	93
6.5.3	Weight parameter optimization . . . . .	97

---

6.5.4	Simulation of camera capture . . . . .	100
6.6	Conclusions . . . . .	101
<b>7</b>	<b>Textured patterns used for barcodes</b>	<b>103</b>
7.1	Introduction . . . . .	103
7.2	New rich QR with two storage levels . . . . .	104
7.2.1	Generation scheme . . . . .	105
7.2.2	Textured pattern selection . . . . .	107
7.2.3	Storage capacity study . . . . .	108
7.2.4	Reading process . . . . .	109
7.3	Experiments with 2LQR code . . . . .	111
7.3.1	2LQR code generation . . . . .	112
7.3.2	Message extraction . . . . .	113
7.3.3	Storage capacity analysis . . . . .	115
7.3.4	Reading capacity of first and second levels . . . . .	118
7.4	Two level QR code for document authentication . . . . .	120
7.4.1	Printed document authentication system . . . . .	120
7.4.2	Generation of authenticating 2LQR code . . . . .	121
7.4.3	Reading process of authenticating 2LQR code . . . . .	122
7.4.4	Authentication process . . . . .	123
7.5	Experiments with authenticating 2LQR code . . . . .	124
7.5.1	Pattern degradation for authentication . . . . .	125
7.5.2	Message extraction from authenticating 2LQR code . . . . .	126
7.5.3	Authentication test . . . . .	128
7.5.4	Simple attacks . . . . .	129
7.6	Message sharing vs authentication scenario . . . . .	130
7.7	Conclusions . . . . .	131
<b>8</b>	<b>Experimental study of P&amp;S impact</b>	<b>133</b>
8.1	Introduction . . . . .	133
8.2	Statistical definitions and tests . . . . .	134
8.2.1	Random process characteristics . . . . .	134
8.2.2	Some classical statistical distributions . . . . .	136
8.2.3	$\chi^2$ goodness of fit test . . . . .	137
8.2.4	Kolmogorov-Smirnov test . . . . .	138
8.2.5	Mann-Whitney test . . . . .	139
8.3	Proposed methodologies . . . . .	139
8.3.1	Stationarity of first order test . . . . .	139
8.3.2	Ergodicity of first order test . . . . .	140
8.3.3	Stationarity of second order test . . . . .	140
8.4	Experiments: noise after P&S impact . . . . .	141
8.4.1	Study of scanner noise . . . . .	141
8.4.2	Study of P&S noise . . . . .	145
8.5	Experiment of color distributions after P&S process . . . . .	150
8.6	Conclusions . . . . .	152
<b>9</b>	<b>Conclusions and perspectives</b>	<b>153</b>

---

9.1	Conclusions . . . . .	153
9.2	Perspectives . . . . .	154
<b>10</b>	<b>Résumé en Français</b>	<b>157</b>
10.1	Introduction . . . . .	157
10.2	État de l'art . . . . .	159
10.2.1	Outils de protection de documents . . . . .	159
10.2.2	Modélisation du processus d'impression et de numérisation et traitement d'image . . . . .	160
10.2.3	Code-barres enrichis . . . . .	162
10.3	Contributions . . . . .	163
10.3.1	Motifs texturés utilisés pour l'authentification de documents . . . . .	163
10.3.2	Code-barres de haute densité . . . . .	166
10.3.3	Motifs texturés utilisés pour des code-barres . . . . .	169
10.3.4	Étude expérimentale du processus d'impression et de numérisation . . . . .	173
10.4	Conclusion et perspectives . . . . .	176
	<b>List of publications</b>	<b>179</b>
	<b>Bibliography</b>	<b>180</b>

# Notation

$Q$	database dimension
$q$	alphabet dimension
$I$	security tag
$r_a, r_b$	number of horizontally and vertically patterns used in $I$
$I_s$	scanned security tag $I$
$P_l, l = 1, \dots, q$	textured patterns
$r \times r$	size of textured pattern $P_l$
$b$	number of black pixels in $P_l$
$S_l, l = 1, \dots, q$	scanned patterns
$CP_l, l = 1, \dots, q$	representative candidates of $P_l$
$M$	barcode module, i.e. squared image $r \times r$ pixels
$M'$	scanned barcode module
$p(i, j)$	image pixel
$D(p(i, j), p'(i, j))$	distance function
$w(i, j)$	weight function
$W$	weight parameter
$M_{pub}$	public message
$M_{priv}$	private message
$[n, k, d]$	error correction code parameters: $n$ - codeword length, $k$ - message length, $d$ - minimal Hamming distance between the codewords
$C$	cyclic code
$g(x)$	generator polynomial
$m(x)$	message in polynomial representation
$c(x)$	codeword in polynomial representation
$K$	scrambling key
$\epsilon$	distance threshold between textured pattern correlation values
$N \times N$	number of modules in QR code
$N_{code}$	number of codewords
$C_{priv}$	codeword of private message
$X(s)$	random variable
$X(t, s)$	random process
$f()$	function associated to the printing process
$g()$	function associated to the scanning process



# Chapter 1

## Introduction

### 1.1 General context

In today's world securing different data is very important to protect copyright and verify authenticity [28]. The nowadays challenges are the detection of fake invoices, bank checks, diplomas, tax forms and other valuable documents. Due to development and availability of printing and scanning devices, the number of forged/counterfeited valuable documents and product packages is increasing. Therefore, different security elements have been suggested to prevent these illegal actions.

The security elements differ by their production complexity, protection level and verification process. For example, the production process of paper watermarks is difficult and time-consuming, but it can be checked by the naked eyes. In the same time the copy sensitive graphical codes [108], introduced recently, have easy and simple generation process, but the verification can be done only by specialist with high resolution scanning device. Finally, the verification of security paper can be performed only by professionals in laboratories.

The graphical codes sensitive to coping process are very popular in product authentication domain, thanks to its easy creation, cheap production, significantly simple verification and high authentication performance. Therefore, several companies have been created in order to find novel authentication graphical codes and to fill a gap in the market.

Authentication Industries<sup>1</sup>(AI) is a start-up of Montpellier region, that works in document protection and authentication. AI develops solutions against frauds using graphical security elements, that could be used, for example, for protection of product packaging and certification of printed document from a digital support.

In order to boost the research base of the company, AI and the ICAR (Image & Interaction) team of Laboratory of Informatics, Robotics and Microelectronics of Montpellier

---

1. <http://www.authenticationindustries.com/ai2015/?lang=fr>



(LIRMM) have started to collaborate. The contributions of this thesis are the results of this 3 year collaboration period.

The document authentication can be separated in two classes. The first aims to authenticate the document content using forensics methods [45, 81] or perceptual hashes [41]. The authors in [150] suggest to encode the document local hash in barcodes used for document tamper proofing. Nevertheless, the authentication using barcodes can be easily copied if it was printed using ordinary inks [154].

Another approach considers the authentication of the document support and is based on the fact that each time an image is copied, some information is lost [108]. This technique consists of specific graphical codes that could be used to distinguish original from one of its copies. The copy detection patterns [108] are used to fight against counterfeits and for product and document authentication [9, 106].

The both approaches are important for document/product authentication. They are sometimes combined in order to assure the complete document protection against forgeries and counterfeits.

## 1.2 Problem description

In this thesis we present several novel security elements that aim to protect valuable documents and packaging against unauthorized coping process. The additional capacity of these graphical codes is storage of huge amount of secret information.

The overview of considered authentication system using security elements is illustrated in Fig. 1.1. In general, in this system we have two main players. The legitimate source, who encodes the secret message, generates the security graphical code and prints it using authorized printer. The authority center performs the verification process that consists of scanning the printed security graphical code, pre-processing procedures, authentication testing. Then, in the case of authentic graphical code, the message extraction can be provided.

Most of counterfeits are done in the interval between print and scan processes. The third (additional) player is an opponent that tries to counterfeit the document with graphical code. His goal is to create a graphical code that is considered as authentic by the authority center. However, he has access only to printed graphical code. Therefore, he has to scan, reconstruct and re-print the graphical code in order to create a document looking like the original.

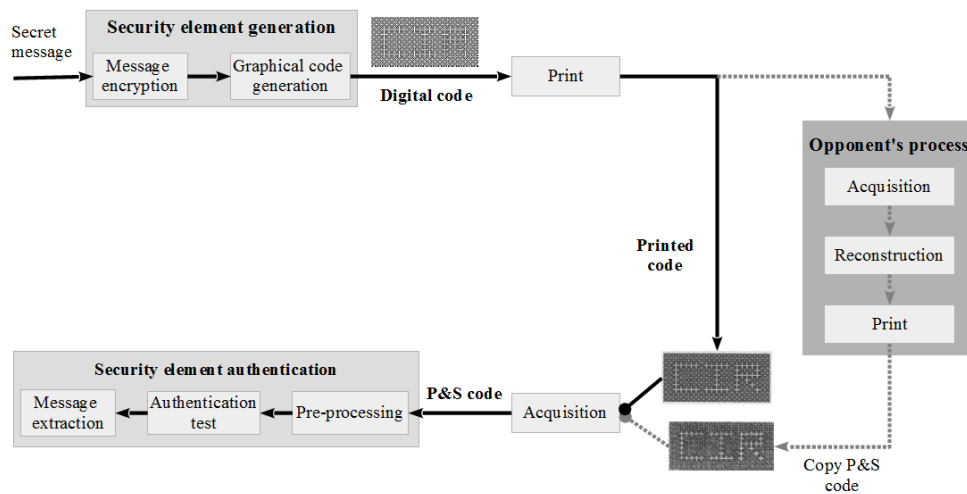


FIGURE 1.1: Considered authentication system using printed security element.

### 1.3 Dissertation structure

This thesis presents several novel graphical codes that are constructed using specific textured patterns. In particular, we suggest to use the proposed graphical code for detection of unauthorized document duplication. Several opponent actions (attacks) have been performed to verify the robustness of proposed graphical codes. This thesis is structured in two main parts.

The first part overviews the related work. The different document protection tools are presented in **Chapter 2**. We start with overview of these tools that comprised technological protections, chemical protections and security printing protections. Then, we focus on one branch of security printing protection: printing security patterns. And we discuss the different techniques for information hidden and document authentication.

In **Chapter 3** we introduce the impact and modeling of print-and-scan process. This process plays a fundamental role in printed document authentication, due to its stochastic nature and defaults of physical process. We discuss the physical process of both printing and scanning as well as their possible impacts to printed documents or images. Further, several print-and-scan models are presented.

**Chapter 4** contains a survey of existing rich graphical codes. We categorize these codes into four big classes depending on their functionality. The first category embraces the user friendly barcodes that aim to increase the aesthetic view of graphical codes. The second category includes the high storage capacity barcodes that allow us to store a huge amount of information using colors and specific structure. The third contains the data hidden barcodes that embed the invisible information using watermark techniques. The last category holds the authentication graphical codes that employ the randomness of print-and-scan process for authentication purposes.

The second part involves the contributions of this thesis. In **Chapter 5** the textured image with visual message is presented. This textured image is generated using specific

textured patterns that change its structure during the printing, scanning and copying processes. We define specific criteria that have to be respected during the textured pattern choice. In order to verify the authenticity of this textured image, we should correctly detect the textured patterns used for visual message construction. For this, we use different correlation based pattern detection methods. During the pattern detection experiments, we note that the correlation measure among original patterns and its print-and-scan degraded versions presents the higher discrimination values.

This last result encourages us to use the comparison methods in order to improve the reading rate of existing high density barcodes. Therefore, in **Chapter 6** we suggest a new measure for module classification that is called weighted mean squared error. This new measure is based on module centrality bias that increases the significance of central pixels and decreases the significance of border pixels of each module. It allows us to improve the binarization results. Moreover, we have shown that it is possible to apply this measure in standard binarization methods.

The results obtained in both previous chapters lead us to contributions of **Chapter 7**. We suggest a two level barcode (using the QR code as a reference barcode). This novel two level barcode stores the information in two levels: the first (public level) is readable by any barcode reader, the second (private level) is constructed by replacement of black barcode modules with specific textured patterns. This two level barcode has three strong points: 1) it stores information into two separated levels; 2) the information stored in the second level is available only to authorized users and is invisible to unauthorized users; 3) finally, this barcode could be used for printed document authentication, thanks to specific characteristics of textured patterns. Numerous experiments have been performed in order to study the storage capacity limits and reading limits depending on pattern size and pattern density. Then, additional experiments have been performed to study the barcode sensitivity to copying process.

The print-and-scan process is generally considered as being a random process that can be modeled by a white additive Gaussian process. The **Chapter 8** aims at experimentally validate or invalidate this hypothesis. Moreover, the experiments we carried on has been conducted in order to separate the printing from scanning impact. It follows that the usual hypothesis cannot be supported by the experiments. The Gaussian modeling is invalidated for the whole print-and-scan process. It has been highlighted also that this process is neither white nor ergodic in the wide sense. The scanner noise seems to be mean ergodic and its contribution to the whole print-and-scan noise is in a minority. Finally, the **Chapter 9** summarizes this thesis and presents several research perspectives and R&D projects.

## Part I

# Related work



## Chapter 2

# Printed document protection techniques

### 2.1 Introduction

In our daily life we often deal with different documents: diplomas, licenses, invoices, tickets, banknotes. And all the time we have to decide whether this document is an authentic or a fake. Each valuable document has a particular set of protection elements. The document protection is a long time explored and well developed field. There exists a lot of different protection elements and techniques that could be classified differently. In this chapter, we chose one way to classify the document protection tools. Of course, due to high links among different protection tools our classification is sometimes permeable. The document protection is a set of document visual elements, special characteristics and techniques, that allows us to identify accurately the authenticity. There are three levels of document protection tools depending on control complexity: 1) for consumer verification; 2) for specialist verification; 3) for laboratory verification. The first level contains the protection elements that could be verified by ordinary persons without any specific device. The examples of these elements are watermarks. The second level contains the protection elements that could be verified by experts and trained staff with simple specific devices as scanners and magnifying glass. The elements which can be verified using automatic tests are also in the second level. Examples of these elements are graphical codes or/and printing techniques. The third level contains the protection elements that could be verified in specific laboratories by experts. The damage of document could be accepted in this level. Thus, this verification test may require high capacity microscopes and chemical expertise. An example is a synthetic DNA that needs specific verification in the laboratory.

The document protection techniques had been created to fight against counterfeits and forgeries. The *counterfeit* is a reproduction of document to earn profit. That means that

the counterfeiter tries to copy all or most of protection symbols without changing the content of the document to produce a fake that cannot be distinguished from the original. The *forgery* is the document falsification. That means that the document content is changed to produce a new document from an unauthorized source with well-formed information.

The main target of this chapter is to present the existing document protection elements and techniques. We start with overview of classical optical protections techniques in Section 2.2. And then, we discuss attentively the information hidden techniques, Section 2.3, and the document authentication techniques, Section 2.4. Finally, we conclude in Section 2.5.

## 2.2 Optical protection

The optical security is a strong robust protection type for valuable documents (diplomas, banknotes, licenses). These types of protection are the most popular and become classics. The strong points of these protections are access (for example, real hologram generation) and implementation difficulties. For example, the see-through devices that are printed onto both sides of the document, have an easy idea, but are very difficult to print. The document protection tools are divided into three main branches:

- Technological protection consists of security elements that are added during paper production or printing technology.
- Chemical protection includes the chemical structure of paper and inks and also physical properties of security elements.
- Security printing protection embraces various security inks, security printing techniques and security printing patterns.

We schematically illustrate all optical protection tools in Fig. 2.1. The combination of these security elements secures the valuable document and offers us the visual or latent proofs of authenticity.

### 2.2.1 Technological protection

Technological protection is a set of features which are detectable during study of materials, papers or document elements. This type of protection consists of watermarks, security fibers, embedded threads, optical and chemical features of paper, optical features of inks and holograms.

Watermark is the most popular technological protection. The watermark is an image created during paper production by local thickness changes of paper fiber mass during the wet phase of paper production [149]. It can be observed against light. The watermarks have high security value due to reproduction difficulties during counterfeiting process.

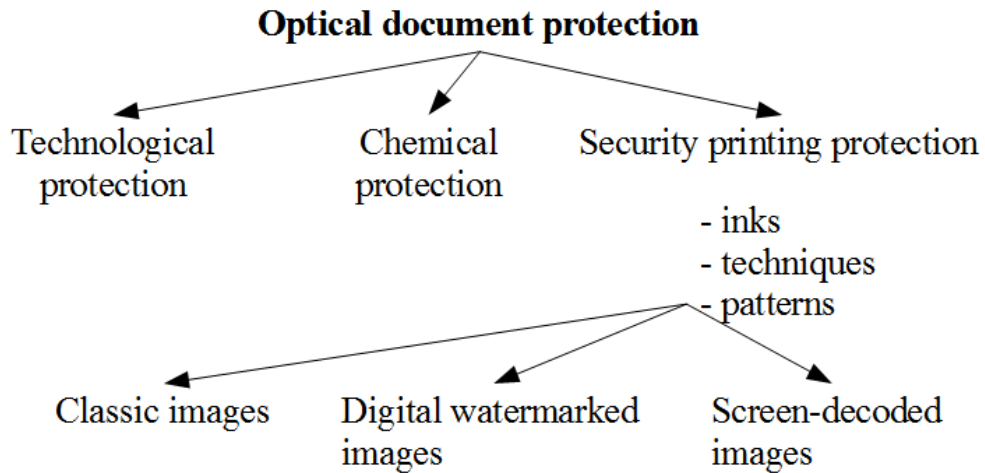


FIGURE 2.1: An overview of optical protection tools.

Security fibers are the particular fibers that are added into the paper during paper production [149]. These fibers have specific properties: color or sensitivity to external irradiation (ultraviolet, infrared, etc.). The security fibers offer low authenticity protection (as it is easy to produce look like structures) but could be used as one tool in a set of protection elements.

Security thread is a thin polyester ribbon [149] which is embedded into the paper during paper production. That can be opaque with a white or metallic coating, transparent with microlettering. The embedded threads are visible in transmitted light and invisible under normal observation in diffuse reflection.

Optically variable inks are widely used in document protection. This kind of protection consists of specific ink use that changes its color depending on light conditions.

Holograms are the most robust, therefore the most used, optical protection tool. The technological process of hologram generation is very complex. Nevertheless, the verification of hologram authenticity is also very complex (as a non-professional does not know what to expect), available only for professional process.

Page perforation is a technique that allows us to write unique variable information to each document substrate [149]. The laser perforation and laser-engraving techniques have more advantages in comparison with watermarks, as it is robust not only against counterfeiting but also against forgery.

The technological tools offer a high protection, nevertheless these elements are controlled with difficulty by non-professionals. The huge gap between production and control sometimes implies the high protection property loss: a customer accepts as authentic any looks like technologically protected product.

### 2.2.2 Chemical protection

Physical and chemical protections include material characteristics that can be discovered in different spectral parts. As an example we can list fluorescence, infra-red protection



and magnetic protection.

Security paper is relatively dark under ultraviolet irradiation, as it does not contain optical brighteners [149]. This type of paper is called ultraviolet dead paper. In addition, the paper, used for valuable documents, could have special tint, gloss or special structure. As the paper used is ultraviolet dead, the specific symbols and text are added using fluorescence inks.

Infrared (IR) protection is based on material properties sensitivity to IR rays. Contrary to devices for fluorescent verification that are available for any user, the devices for verification of IR-protection are only used by specialist. The IR-protection is mostly used in banknotes production.

Magnetic protection means the presence of magnetic characteristics of document materials. This protection belongs to security fibers. The magnetic characteristic does not change the color of fibers, and can be detected only by using specific devices.

### 2.2.3 Security printing protection

Security document printing protection consists of three matters: printing security inks, printing security techniques and printing security patterns.

The numerous variety of *printing inks* allows us to protect the elements of valuable document differently. The reversible photosensitive inks could have photochromic feature (the change of color or optical density), or luminescence feature (the emission of light). The thermochromic inks change the color with temperature. The metamerics inks are visible only in particular light (red filter, near-infrared irradiation). The fugitive inks disappear once bleaches or organic solvents are applied. The iridescent inks display a notable change in color with angle of observation and illumination. The features and examples of all these and other security inks are well introduced in [149]. The security inks have a good protection capacity (as the specific pigments are difficult/impossible to reproduce thanks to secret technology), can efficiently protect the valuable documents, but unfortunately, most of them could be verified only by experts using specific devices.

The *printing security techniques* could be classified into four types by geometrical structure of printing form: the letterpress printing, the offset printing, the intaglio printing and the screen printing. Each of these printing techniques has specific printing process, therefore the same character printed using different techniques will look differently [149]. The character printed using the letterpress technique has thickenings along the edges of the strokes; in addition, a paper deformation occurs due to printing form pressing, that is illustrated in Fig. 2.2.a. The offset printed characters characterized by uniform staining and clear-cut edges without paper deformation, see example in Fig. 2.2.b. During the intaglio printing, the surface of printing element is partitioned into raster elements, that is visible in the edges of printed elements, the example is illustrated in Fig. 2.2.c.

In the characters printed using screen printing, the grid structure is always visible, see Fig. 2.2.d.

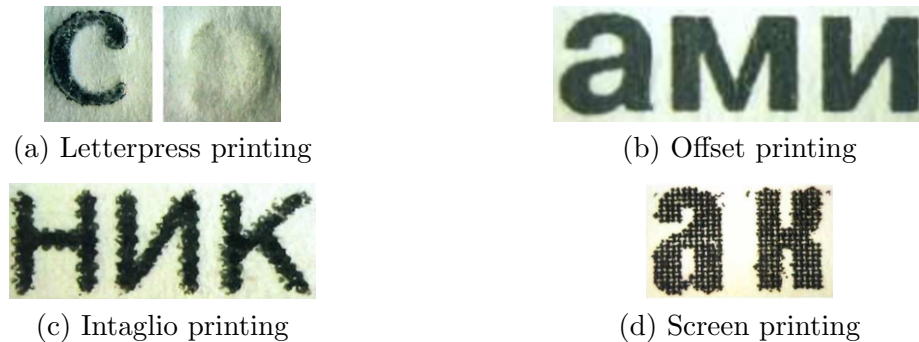


FIGURE 2.2: Examples of characters printed using<sup>1</sup>: a) Letterpress technique, b) Offset technique, c) Intaglio technique and d) Screen printing.

The last matter is *printing security patterns*. The printing security patterns are divided into three main groups: classic images, digital watermarked images and screen-decoded images. The classic images are presented by:

- Guilloches - geometric fine-line patterns that are widely used to denote intricate ornamental borders and emblems consisting of fine curved lines.
- Microprinting - printing of very fine lettering using the resolution that is hardly visible by human eyes,
- See-through register - an image printed in both sides of a document, each side consists of spatially related image elements. The complete image could be reconstructed by viewing against the light.

Sometimes, the valuable documents have a specific background that consists of micro-patterns. These patterns do not disturb the reading process of original document, nevertheless the reading process is impossible in the copy. One of the novel representative of this technique is CopySafe+<sup>TM</sup> [107] technology which is composed on the specific security patterns on a reflective layer composed of aluminum substrate.

We also can use the paper fibers as a security pattern [53]. The fiber structure is detected by using a video microscope, then it is used to produce a unique document identifier which is printed into the document.

The visible copy-detection patterns could also be used to distinguish between the original and duplicates of printed documents with the naked eye. There are two types of anti-copy patterns [69]: delete patterns that are broken or lost during copying or scanning, and survival patterns that are preserved during copying or scanning, but reproduction process is visible. This technique employs the characteristics of digital image input and output devices. Several image patterns disappear or disfigure in the process of coping or scanning due to low pass filtering that blurs the image.

The digital watermark images involve embedding of additional information into digital

1. All examples are taken from the web-page (ru): <http://www.bnti.ru/showart.asp?aid=940&lvl=01.03.05>.

media. This group consists of different multimedia security techniques for information hidden. These techniques were created for digital multimedia but today most of them are extended to hardcopy documents. These hidden techniques are in details discussed in Section 2.3.

The screen-decoded images are invisible or illegible to the human eye but are visualized or decoded by periodic phenomena [149]. There exists a huge amount of different techniques to create the images of this type. The main characteristic of these security patterns is sensitivity to reproduction process that can be appeared by visible moiré patterns, visible encoded messages, image density changes, aliasing effects. In this group there are also security elements that could be verified by decoding screens (for example lenticular screens that render brighter moiré patterns because they do not absorb the light [148]) placed over the printed images. These specific images are created by line angle, dot frequency and line phase modulations.

We separate the techniques from two last groups into two classes depending to their functionalities. The first class consists of techniques used for optical protection, see Section 2.3. The second class, discussed in Section 2.4, embraces the techniques used for document authentication.

## 2.3 Printing security patterns for optical protection

In addition to the optical protection strong points (access and implementation difficulties), a supplementary protection layer is added using encryption or signal processing tools. The techniques that imperceptibly embed security information into a picture of the document become very popular in valuable document protection. They have two advantages: 1) the hidden information could be detected and decoded only by specific device; 2) the embedded data is inseparable from the picture, therefore the forgery become very difficult.

The information hiding techniques can be divided into two groups. First group embraces the techniques that use the multimedia content as support for the embedding process. It involves digital watermarks, steganography, image encryption and visual cryptography. The second one comprises the techniques that use the particular characteristics of the printing process for information hiding. This group covers data embedding using halftoning and dot-orientation. Table 2.1 lists these techniques and their applications.

### 2.3.1 Digital watermark

Digital watermarking (DW) techniques embed the supplementary information into digital media [32]. The DW techniques provide brand protection, intellectual property protection, tracking and tracing of valuable products. Each data hidden scheme is characterized by the following measures:

	<b>Hidden techniques</b>	<b>Application scenario</b>
Multimedia change	digital watermark	brand protection, intellectual property protection
	steganography	secret communication
	image encryption	make unintelligible visual information
	visual cryptography	secret message sharing decoding using human visual system
Printing process	halftoning	automatic document authentication by specific device
	dot-orientation technique	

TABLE 2.1: Different data embedding techniques and its application scenario.

- Storage capacity - message length that can be embedded into the image;
- Imperceptibility - an image with hidden message should have the same perceptual quality and the presence of this message must be imperceptible;
- Robustness - a hidden message should be extracted from a watermarked image after attacks (cropping, scaling, filtering and noise addition);
- Security - it should be difficult to extract the embedding message without secret-key and decoding algorithm;
- Computational complexity - the embedding and extraction processes should not be computationally costed.

Depending on robustness against compression, quantization, filtering and geometrical distortions (cropping, translation, rotation) [13], we distinct robust DW and fragile DW. The robust DW must be insensitive to all kinds of image manipulations, contrary the fragile DW must be destroyed after applying any manipulations.

The other classification of DW is performed by their visibility with the naked eye: overt DW and covert DW. The overt DW clearly identify the ownership and is robust against removal attack. In addition, this DW can be used in hardcopy documents and can perform the document authentication [112] by using position based watermark technique [19]. The covert DW hides information into images and is readable only by dedicated device and software.

The covert DW techniques can embed redundant data or an image. The redundant data embedding implies pseudo-random noise insertion into the image that is invisible but holds in useful information about protected digital media. The reading process is performed by authorized inspectors with dedicated device and allows to differentiate copies from originals. This property is possible thanks to difficult process of correct watermark regeneration and embedding in counterfeit digital media.

The important parameters of redundant data embedding are:

- payload size, i.e. the message length. This message is embedded repeatedly over the image, the shorter message provides the higher robustness.
- modulation strength, which is given by the pixel luminance change. The higher intensity provides the more robust watermark but that is more visible (i.e. the image is less aesthetic).

Only the images with non-uniform density elements can be efficiently used for digital watermark embedding as the inserted pseudo-random noise message would be invisible. We differ two types of watermark methods: spatial and frequency approaches. The spatial approach manipulates with the image itself by changing pixel values of image. One of the more used techniques is the digital watermark scheme that embed information in Least Significant Bits (LSB) that ensures the invisible watermark. Then, for robustness against different attacks, the watermark is embedded in perceptually significant components [31]. The frequency approach consists in manipulation with transform domain: DCT components [114], DWT decomposition [115], quaternion fourier transform [14] (used for color images). All these techniques create robust watermarks against cropping, destroying or filtering attacks.

The exploitation of image content can increase the invisibility and robustness of DW techniques. These techniques based on concept of Human Visual System (HVS) properties, local energy of a narrow band [34] and content similarity [11, 12].

Specific covert watermarking techniques are suggested for hardcopy documents. For authentication of ID photos the watermark based on the polarity of Hadamar transform coefficients are proposed [57]. This technique allows to invisibly insert a 9-character secret message into smooth photographs of size  $180 \times 130$  pixels. The hardcopy document quality suffers from Print-and-Scan (P&S) process distortions. The geometric distortion could be removed from the scanned watermark document in order to improve the embedded watermark quality [62].

The hidden technique based on experimental modeling of the P&S process, called selective embedding in low frequencies, are proposed in [133]. This technique aims to survive cropping by estimating its effect beforehand.

The amplitude modulation halftoning can introduce degradation into a watermarked image. A DFT based watermarking method proposed in [113] depends less on the cluster dot shape than on the halftone frequency. Watermarks that survive the P&S process are still popular research topic.

### 2.3.2 Steganography

Steganography is the art of concealed communication [31]. The steganographic scenario [130] is described in this manner:

*Two prisoners can communicate between them via controlled channel: all their messages are verified by guard. When the guard detects any secret information exchange, he stops communication immediately. That is why the prisoners try to find the imperceptible way for secret message exchange. For realization of this communication, they embed the secret message (image or text) into the multimedia content (image, video, audio or text).*

The multimedia object that is used for information hidden, is named cover-object and the cover-object with embedded secret message is stego-object. Depending on the type of the cover-object, many different steganographic techniques are suggested [42].

In this discussion we focus on image steganography. Generally, pixels intensities are used for information hidden. The parameters used for image steganography are: a cover-image is a multimedia support for hidden information, a message is a secret information hidden into the cover-image, a stego-image is the cover-image after message embedding, a stego-key is a security key used for embedding and extraction of the secret message from stego-image. The steganographic techniques can be divided into three categories: spatial domain methods, frequency domain methods and adaptive methods [24].

The spatial domain methods embed the secret message into the cover-image in the spacial domain which involves encoding at the level of the LSBs [74]. However, these methods produce high visual distortions in the cover-image as the hidden information is seen as "non-natural" [30]. Thus, in spatial domain methods we face with a trade-off between the payload and the cover image distortion. One of the most robust steganographic methods in spatial domain is Highly Undetectable steGOnography (HUGO) scheme [103]. Nevertheless, steganalysis methods have been proposed recently [44, 118]. The frequency domain methods hide information in areas that are less exposed to compression, cropping and image processing. Therefore, these methods have an advantage in comparison with the spatial domain methods. Most of these methods insert information into the DCT coefficients, the DFT coefficients or the DWT coefficients. The methods that embed information into the DCT coefficients often are attacked by statistical approaches. The most known methods are OutGuess [116] and F5 [160]. The DFT based steganographic schemes suffer from round-off errors that render them improper for steganography applications. And the DWT based schemes just start to appear.

The adaptive steganography is a special case of the two previous methods [24]. These methods study the statistical global features of the image before embedding the secret message into LSBs or DCT coefficients. The statistics shows the places where the changes could be done without loss in robustness and perception. The adaptive methods seek images with existing or deliberately added noise and images that demonstrate color complexity [24]. This strategy permits to be robust against cropping, compression and image processing attacks.

As digital watermark, steganography is extended into printed images. The printable steganography is introduced by the Japanese Fujitsu Laboratory <sup>2</sup>. The developed method insert data into a printable picture, so that it is invisible to the HVS, but can be decoded by a mobile phone with a camera. First of all the color cover-image is transformed in its hue, saturation and value components. The secret message is embedded into the Hue domain that is imperceptible by human eyes. In the end, the stego-image is printed. During the reading process the mobile camera retrieves the embedded data.

The security scheme which protects scanned documents from forgery using self-embedding technique is suggested in [23]. This method not only allows to fight against document

---

2. <http://classes.design.ucla.edu/Spring06/159/projects/tina/report.htm>

forgery, but also permits to gain access to the original document despite being manipulated. Thanks to its commercial applications the printable steganography is popular industrial and academic research topic.

### 2.3.3 Image encryption

The image encryption techniques aim to make unintelligible visual information [146]. The encrypted image can be decoded only by authorized users. The image encryption techniques differ from text encryption techniques, as a small distinction between original and decoded images is acceptable. Classic image encryption scheme encrypts image permuting pixel, slice or patch locations using a secret key [25, 46, 101]. There are a lot of other image encryption techniques based on modified block cipher encryption schemes [165, 166], asynchronous stream cipher based on generalized continued fraction [86], selective encryption [117, 121], compression schemes [78], changes of transformation coefficients [73], hash function [124] and digital signature techniques [131]. This field is huge and well developed, we want only to highlight several image encryption techniques that were developed for P&S channel.

The image encryption scheme that survives over P&S channel is introduced in [36, 49]. This method is based on permutation of image pixels according to a secret key  $K$  and a public initialization vector  $IV$ . The initialization vector and shared secret key are used to create a unique pseudo random sequence for each image encryption. Then the original image and the pseudo random pixels are concatenated and scrambled, and the final encrypted image is placed into a grid. The synchronization grid is used for pixel position location after P&S process.

An image encryption method that can be used into the printed documents, and then be decoded using a scanner or a cell phone, is recently introduced [7]. This method applies a special image conversion process before the scrambling process.

### 2.3.4 Visual cryptography

Visual cryptography (VC) is a special case of secret message sharing techniques. The Visual Cryptography is first introduced [93] for realization of the following scenario:

*The four intelligent thieves have deposited their plundering money into the bank. They do not trust each other and they do not want that one of them could withdraw the money and escape. In the same time, the money withdraw is allowed by two of them. Therefore, they split the bank code into 4 parts and encode it into 4 partitions, so that only two or more participants would be able to retrieve the bank code. The additional condition: the code should be decoded visually, without any device use. As a result each thief has a transparency. A single transparency could not be used for the code extraction. Nevertheless, any two transparencies should visualize the secret bank code, if they are staking*



*together and aligned.*

The  $(k,n)$ -VC is a cryptographic scheme which encrypts written material (printed text, pictures) in a perfect secure way and decode directly by HVS [93]. This is the secret sharing scheme where the image is split into  $n$  shares with condition that the combination of  $k$ , ( $k \leq n$ ) shares could decrypt and visualize the secret message while any  $k - 1$  shares could not retrieve it. The VC is a secret sharing scheme extended for images.

A secret image is a black-and-white image. The original secret image is split into  $n$  modified versions (shares) where each pixel subdivides into  $m$  black and white sub-pixels. Therefore, the share structure can be described by an  $n \times m$  boolean matrix  $S = [S_{ij}]$ , where  $S_{ij} = 1$  if the  $j$ th sub-pixel in the  $i$ th share is black. The decoding process is performed by XOR operation of  $s$  transparency shares. If the number of  $s$  shares is appropriate ( $s \geq k$ ) then its stacking allows us to visualize the secret message.

The modified VC scheme is suggested in [63] where authors propose to secure the shares by using the additive homomorphic property of the Paillier algorithm or multiplicative homomorphic property of the RSA algorithm. In their VC scheme the secret image and shares are encrypted using the same encryption algorithm, and the scrambled image is created by addition of secret image and  $n$  shares. During the secret message extraction the  $k$  players need only to subtract their own shares with no specific order from the scrambled image.

The visual authentication and identification protocols are suggested in [92]. The visual cryptography can be used in electronic-balloting system, encryption of financial documents [22]. The schemes where all shares are meaningful images, are named Extended Visual Cryptography (EVC) schemes. The EVC scheme could also be used for ID card authentication [128]. In this scenario the shares are the bar codes that contain the ID photography of ID card owner.

In real life the transparent shares are not so easy to create. The solution is the use of hardcopy shares. In this scenario the shares are printed into classic office paper, then the scanned shares are used for secret image reconstruction. Nevertheless, the classical VC scheme has some drawbacks [163]:

- The restored secret image has lower resolution than the original.
- The scheme is introduced for binary images.
- The superposition of shares is difficult to produce.
- The size is twice bigger.

The grey-level and color VC schemes with hardcopy shares are introduced in [18, 79] and [59, 60, 64], respectively. That is realized by using of additional processing such as halftoning and color-separation.

The share superposition problem can be solved by mark embedding into shares. These marks are inserted in the high frequency coefficients of Walsh transform [163]. That allows us to precise the alignment of shares automatically.



### 2.3.5 Halftone-dot orientation data embedding

In this section we present the data embedding techniques that exploit characteristics of the printing process. The printing and scanning processes add irreversible distortions to image. The detailed overview of these distortions is presented in Chapter 3. The exploitation of these characteristics offers greater potential for data embedding and is well-suitable for hardcopy applications since the embedding occurs just prior to printing [21].

The printed binary patterns can be interpreted as digital data. This technology is called DataGlyphs [55, 56] and uses differently oriented binary codewords for data embedding into real-life images. The detection process starts with estimation of the binary patterns from a scanned image and then they are compared with binary codewords.

The technique that has similarities with previous one uses the halftone-dot orientation modulation [21] for data embedding. The considered data embedding scheme is performed by clustering dot halftones. For a given gray level, the authors generate elliptic halftone dots. The data embedding is performed by control of the ellipse orientation. The data from a scanned image is detected by using statistical criterion that uniquely identifies an ellipse orientation and the probabilistic P&S channel modeling.

The latent images are often employed for valuable document protection. These images are generated by using a continuous-tone cover image and a binary figurative pattern. The artifact-free latent image inside a cover image can be embedded by modified digital halftoning techniques [158]. This latent image can be extracted by applying a frequency domain detection. In addition, during the reproduction using copy machine the latent image becomes visible.

## 2.4 Security techniques for document authentication

Most of the techniques introduced in the previous sections, such as UV inks, holograms, watermarks, can be used for authentication. To increase the security of technical protection (introduced in Section 2.2.1) and to overcome the lack of expertise of common users, numerous automatically authentic security elements are suggested last years. Contrary to embedding techniques, the authentication techniques have to be sensitive to P&S and copying processes. In addition, we should answer to a question: what is the authentication subject? It could be whether the document itself (support) and/or the meaning information (data). In the case of data authentication, we perform the integrity check. Authentication of documents with these specific security elements is impossible by the naked eye. A specific equipment (as a scanner or a microscope) is required to perform the authentication test. But the verification process is automatic, thus it is quite suitable for valuable document (i.e. diploma, check, invoice, passport) authentication.

An automatic authentication techniques can be divided in two groups. The first group

performs the integrity check by using document/image hash techniques. The second group evaluates the document authentication by taking into account the physical characteristics of printing and reproduction processes. In following sections we present the printed document authentication techniques that can be used to prevent the document/product counterfeits.

### 2.4.1 Document hash function

To ensure image integrity we can use image hashing technique. Image hashing is a branch of perceptual hashing. In comparison with cryptographic hashing algorithms, perceptual hash should be designed to withstand various intentional/unintentional legitimate modifications (that could occur during the document life cycle), but it should be sensitive to various intentional malicious modifications [150]. A perceptual image hashing system generally consists of four pipeline stages: the transformation stage, the feature extraction stage, the quantization stage and the compression and encryption stage [51]. The robust visual hashing [41] is an active upcoming research field. It has to fulfill three constraints: robustness to distortions, security and universality [72]. Image hashing is focusing on perceptual features which are invariant under perceptually insignificant distortions [88]. Therefore, the bit changes do not change the perceptual content, and the hash value will not be affected [162].

The image hashing techniques can be classified into the following categories [76]:

- Statistic-based schemes extract the hash by employing the image statistics [67].
- Relation-based schemes are based on some invariant relationships between two coefficients of DCT or DWT [85].
- Coarse representation-based schemes extract the hash by making use of coarse information of the whole image [43].
- Matrix-based schemes construct the hash by using the matrix factorization [89].
- Low-level feature-based schemes extract the hash from the image edge or salient feature points [17].

Several perception hashes are robust to Gaussian noise, JPEG compression and low-pass filtering [52]. Nevertheless, there are less image hashing schemes robust to P&S process. A P&S resistant image hashing algorithm based on the Radon and wavelet transform are suggested in [162]. The Radon transform is applied to the image luminance distribution, then the wavelet extracts the relationship of the different areas from this distribution. This construction allows the hash to be sensitive to perceptual changes and robust to attacks.

The calculated document hash can be stored in the database, in a barcode, that is embedded into the document or in the document itself [154]. Each of these protocols has its advantages and disadvantages. The hash storage into the database implies the online verification and database infrastructure. The use of barcode is robust to copying,

but cannot assure the uniqueness of the document. The storage in the document itself cannot be separated from the document that is not the case in the barcode protocol.

### 2.4.2 Printing techniques for document authentication

The printing process and paper offer advantages to protect the document against copying. However, the standard document background and image components do not use these benefits. The techniques presented in this section are based on two statements:

- Every time an image is printed or scanned, some information is lost about the original digital image, that is called "information loss principle" [108]. Printing and scanning processes are affected by noise, blur and other changes discussed in Chapter 3. The communication channels where transmitted information is suffer from noise and changes, are always characterized by loss of information. The loss could be minimal and imperceptible by the naked eye but it could be significant for authentication test.
- Each printing and copying device has its own signature. The use of this signature and specific modifications added by device are employed for authentication test. Specific image analysis systems can detect alterations made to laser printed documents even if the alteration is invisible [140].

These two statements include each other as the lost of information is particularly done due to specific signature of printing, scanning and copying devices.

The security element that was constructed with respect to "information loss principle" is Copy Detection Pattern (CDP) [108]. A CDP is a noisy, maximum entropy image, generated with secret key. A CDP is designed to be maximally sensitive to the coping process. It has large variation in high frequencies that are the most difficult to capture by the scanning device. In addition, it has a non-predictable content that secures against reproduction attack. The CDP is impossible to verify by the naked eye. In order to determine whether the document is original or a copy we need a scan of a printed graphic. Then, a specific software makes a comparison of the pixel values in the digital and in the scanned CDP. The comparison can be made by a correlation, a distance or a combined score of different features [108] or by using the copy detection metrics sensitive to P&S process [37]. This CDP can be one of the security elements for identity documents [110]. Due to CDP sensitivity to reproduction, the print quality needs to be reasonably good and the printer, the media and the scanner must be of known types [37].

The innovative primitives to derive secrets from complex physical characteristics of integrated circuits are named Physical Unclonable Functions (PUFs) [136]. The paper characteristics as well as production printing technology can be considered as PUFs. The printer identification [94] and printer technology authentication [95] can be performed by microscopic analysis of paper print. A microscopic analysis is based on the phenomena that each dot in printed document has its own shape. Therefore, the dot can be considered as printer signature at a microscopic scale.

Physical object authentication can also be done by using PUFs. For these objects a

PUFs patterns can be presented by surface microstructure images. The impact of camera noise added to microstructure images can be used for digital content fingerprints computation [15]. The other type of PUFs used for physical objects authentication is Laser-written PUF (LPUF), that put the laser mark into a packaging [127]. More details about P&S process as a PUF can be found in Chapter 3.

The camera identification could be done by the Sensor Pattern Noise (SNP), that is unique for each model and each device of the same model. This SNP is used for digital image forensics, as it is considered as a sensor fingerprint of a camera [58]. Analogically to camera identification, there are some developments in printer identification [87]. That can be done by using a printer intrinsic signature which differs for every printer model and manufacturer's products. This strategy is passive and requires an understanding and modeling of the printer process. Another strategy is based on extrinsic signature embedding. This signature contains an encoded identifying information and is embedded into the document during printing process. The different physical and technical processes produce different signatures during printing using laser printer, inkjet printer and electrostatic copiers [126]. These characteristics allow us to compute the document features based on noise energy, contour roughness and average gradient, and then identify the printer device.

## 2.5 Conclusions

The valuable document protection is not new and is still a very important and upcoming field of multimedia security. The optical document protection is based on three big elements: technological protection, physical and chemical protection and security printing protection. The tools of security printing protection are security inks (that have specific optical, physical and chemical characteristics), printing techniques (that could significantly influence the document quality) and security patterns (that are sensitive to copying and reproduction process).

Most of the security elements such as special papers, special inks or holograms are very expensive and often require special equipment. Less expensive solutions for document protection are security pattern techniques. The security patterns are created in respect to "information loss principle" and unpredictable printing and scanning processes. In our research we focus on security patterns that are sensitive to copying process. They are printed using classical black inks and laser printers.



## Chapter 3

# P&S modeling and image processing

### 3.1 Introduction

Today the printer and scanner devices are accessible for everyone that induces the increasing of counterfeited or falsified valuable or administrative documents. However, the hardcopy documents suffer from printing distortions during production process. In addition during automatic authentication and reading it suffers from scanning impact. The printing phase could be split into before printing phase and while printing phase [134]. The most substantial processes of printing and scanning processes are illustrated in Fig. 3.1.

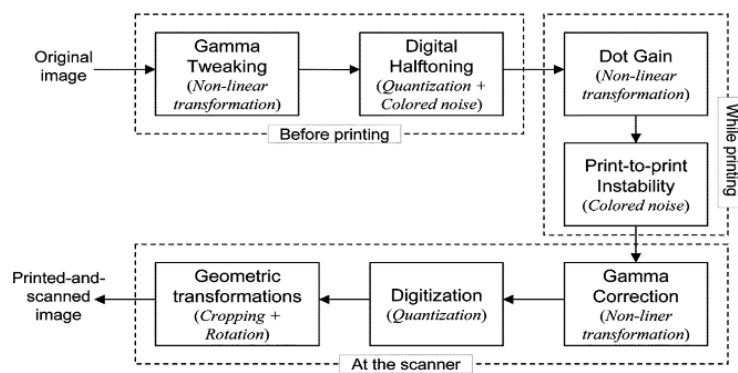


FIGURE 3.1: Processes that add distortions into the image during printing and scanning [134].

Two hardcopy exemplars of the same item (image or document) differ from each other in digital sense, nevertheless they are the same by naked eyes. In order to perform the comparison of printed item and original digital image the printed image is digitized using scanner. That is why the printing and scanning processes are not separable from

each other and the distortions always belong to both of them [164]. This section aims to introduce the characteristic of printing and scanning processes as well as to highlight the P&S modeling techniques.

The changes made by P&S process are often considered as a Physical Unclonable Function (PUF) [106], due to the physical changes and the stochastic nature of the changes. In this section we try to resume the changes added by printing and scanning processes. Printers differ by printing mechanism and marking technology that are used to pass from numerical document to hardcopy document. Scanners differ by the type of sensor and mechanism that generates the numerical document from hardcopy. In this work we focus only on office printer and scanner devices. However, all presented approaches and results could be extended and applied to industrial printers that have higher resolution and better printing quality.

The structure of this chapter is as follows. We discuss the printing and scanning process characteristics in Section 3.2 and Section 3.3, respectively. Then we continue with P&S process modeling in Section 3.4. Finally, we conclude in Section 3.5.

## 3.2 Printing process

The printing process can be produced using various technologies but the most available for common usage are inkjet printer, laser printer and electrostatic copier. The laser printers and electrostatic copiers have almost the same structure, the inkjet printer structure differs a lot from laser printers.

Before being printed the original image is transmitted into control system. Raster image processor in control system converts it into halftone image and correspondent electronic pulse level [164]. After this process the printing technologies of laser and inkjet printers are different. Let observe each of them in details.

### 3.2.1 Electrophotographic printer architecture

The underlying marking technology for laser printers and office copiers is called *electrophotography (EP)*. The EP process consists of six steps: charging, exposure, developing, transferring, fusing and cleaning. The Fig. 3.2 illustrates the EP process.

The Optical Photo Conductor (OPC) drum is uniformly charged through a charger roller during the first step. Then the drum is lighted by a laser and considering the image to be printed, the specific locations are discharged on the drum. These discharged locations are used to develop a toner image by attracting the toner particles. The developed image is next transferred electrostatically onto the paper. Further the toner image passes through a fuser and pressure roller which permanently affixes the toner particles to the paper. The final step is to clean the OPC drum surface from any excess toner and charges.

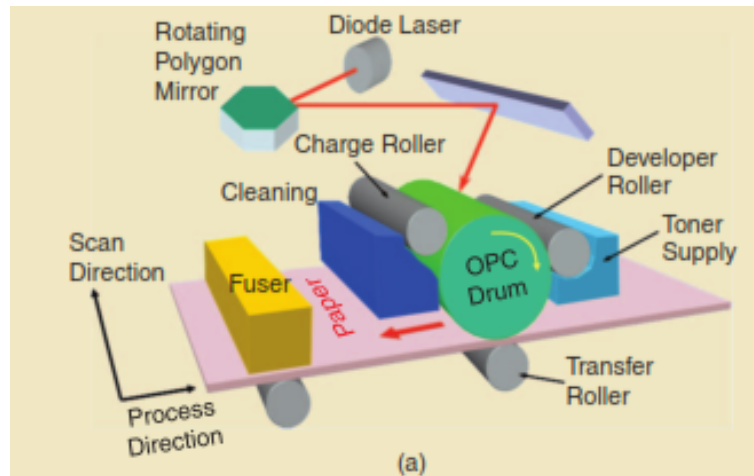


FIGURE 3.2: The electrophotography technology for laser printers and office copiers [28].

The printed output from any printer has defects caused by electromechanical fluctuations or imperfection in the printing mechanism [87]. Let analyze the physical processes that can effect the printed output:

- Laser light, right localization, size and focal can produce imperfections and also can be changed during printer exploitation life.
- Mechanical and optical element motion can introduce imperfections.
- Lens in optical system can particularly introduce non-linearity [164].
- Fluctuations in the angular velocity of the OPC drum can cause fluctuations in developed toner on the printed page.
- Ink transfer from OPC drum onto the paper can produce random errors.
- Fuser can imply thermic defects.
- Manufacture defects lead to nonuniform distribution of toner.
- Cleaning bland errors and deterioration in time cause the printing of additional artifacts on the printed page.

Different printers have different sets of banding frequencies, depending on brand and model [28]. The EP printers can also be characterized by measures of image sharpness, toner fusing characteristics, dot gain and asymetry of tonner distribution.

Most of listed characteristics and effects have a random nature that is statistically hard predictable. All these distortions change an output image frequencies and pixel values, however these changes are not visible by naked eyes.

### 3.2.2 Inkjet printer architecture

The *inkjet mechanism* consists of three components: print-head, carriage and paper-advance mechanism. The paper-advance mechanism passes the paper in the printer process under the carriage. The carriage moves the print-head across the paper in the carriage motion direction and then back at the beginning of the paper. This print-head



consists of a nozzle plate that has several columns of nozzle openings and reservoirs for each ink type. It fires the drops of inks onto the paper. The printer process of inkjet printer is illustrated in Fig. 3.3.a.

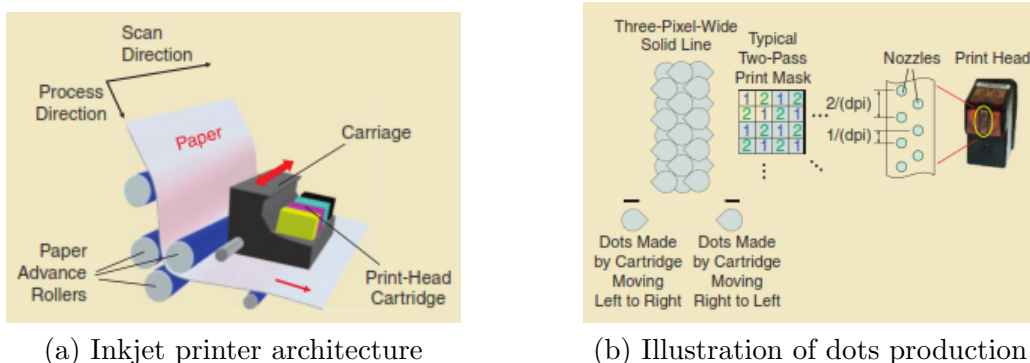


FIGURE 3.3: Illustrations of a) Inkjet printer architecture [28], b) Inkjet printer print-head and dots produced by it [28].

The nozzles in the print-head are arranged in columns (two columns in an example in Fig. 3.3.b). By appropriately timing the firing of the nozzles in each column it is possible to achieve vertical resolution that is equal to the vertical offset between adjacent columns [28]. Each column of nozzles allows the printing of several rows of pixels during a single print-head pass. Depending on printing options, the number of printing passes over each point on the paper differs. In addition, the vertical and horizontal printing resolutions could be different due to instability of carriage motion.

The options of inkjet printer can produce a very complex intrinsic signature [28], as the inkjet printed output can be affected by:

- Inconstant speed and fluctuation of carriage that could introduce printing errors;
- Obstructed nozzles that could impact to printed image;
- Physical process of fired drops that affects to printed image;
- Dot shape that differs depending on carriage motion and the number of passes over each paper point.

Analogically to electrophotography printing, inkjet printing impact is mostly random due to physical and mechanical imperfections.

### 3.2.3 Printer characteristics

Additionally, any printer process has several significant characteristics that could impact the printed document quality. For example, many printer manufactures change the transfer characteristics of the printer to make sure the printed image appears the same as on a monitor. This nonlinear adjustment is called gamma tweaking [134]. The most significant elements of printing process are digital halftoning, printer resolution, ink distribution and paper quality. These elements also distort the image while printing.

### Digital halftoning

The numerical document ought to be formatted into the printer-ready halftone page image [28] in order to be printed. The printer driver renders the objects into the bit-mapped form and generate the continuous-tone document. Then this document is modified by color space conversion, gamut mapping and halftoning.

The numerical black-and-white image is represented by a matrix of pixels  $p(i, j)$ , where each pixel  $p(i, j)$  has its own gray level,  $p(i, j) \in [0, 255]$ . During the printing the halftone process reduces visual reproductions to an image that is printed with only one color of ink. The color changes are simulated by dots of varying size (amplitude modulation) or spacing (frequency modulation). The difference between these two halftoning setups is illustrated in Fig. 3.4. Thanks to the tiny size of halftone dots, the human eyes blend its into smooth tones.

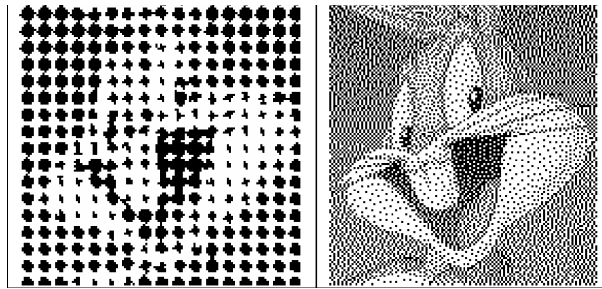


FIGURE 3.4: Examples of amplitude modulation and frequency modulation halftoning techniques [158].

The digital halftoning is a bit-depth reduction step where the original continuous tone image is reduced into a binary image. Halftoning aims to produce an illusion of continuous tone by cleverly trading off amplitude resolution for spatial resolution [21].

The halftoning technology with several modifications is used for printing of color RGB<sup>1</sup> images. The RGB image is represented by three matrices of pixels:  $p_R(i, j)$ ,  $p_G(i, j)$ ,  $p_B(i, j) \in [0, 255]$ . First of all, the RGB image is converted into the CMYK<sup>2</sup> color space. This conversion to secondary printing colors is required as the printing inks are always represented by the secondary colors. The colored images in digital form are always displayed in RGB color space (additive color synthesis), contrary, the printed colored images are always in CMYK color space (subtractive color synthesis). After this image conversion it could be easily printed by varying the density of the four color inks of the printer. An example in Fig. 3.5 shows the use of all four color inks to print red, blue and green colors.

The inverse halftone could be done by human eyes and by scanner. The Human Visual System (HVS) interprets a halftone image as a continuous tone image through blending of tiny halftone dots. The scanner transforms a halftone image into a gray level image that introduces several errors in color values.

The digital halftoning algorithm [147] consists of:

1. Red, Green and Blue color space
2. Cyan, Magenta, Yellow and Black color space

- Point process that separates the image into the dots;
- Neighbor process that gets the average intensity of the pixels in each block;
- Search based method that draws the dots such that the surface area of the dot is equal to the percent of average intensity.

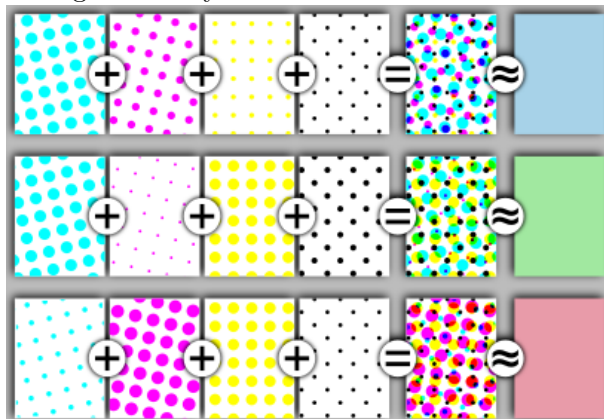


FIGURE 3.5: Examples of color halftone using CMYK color space <sup>3</sup>.

During the digital halftoning quantization noise is added to the image and affects the high frequencies of the image [133]. As the HVS is not very sensitive to high-frequency noise, these modifications are not visible by naked eyes. That is why, due to its sensitivity, the high frequencies suffers a lot from digital halftoning and are not robust to survive the P&S process.

Image blurring is considered by many authors [80, 153] to be a P&S impact due to low pass filtering during digital halftoning and inverse halftoning.

The authors report [80] that the inkjet printers at lower resolution more often blur the images. The authors in [164] suggest that halftone introduces a spatial-variant lowpass linear blur that could be partially eliminated by sharpen or unsharpen filters. The halftone features are also used to printer identification in digital forensics [68].

### Printer resolution

The printer resolution is specified in dots per inch (dpi) and it ought to be equal to the number of pixels per inch to be printed for an image. A number of pixels per inch in an image is specified by a user. The printer quality of tiny details depends on high printer resolution as well as on the type and age of printer and toner status.

The printer resolution mentioned by manufacture can vary sometimes as it is difficult to qualify. For example, the print-head geometry and carriage motion of inkjet printer change the printer resolution over the one printed page.

### Toner distribution

During the dot-by-dot printing process of halftone image the image tends to appear darker than expected, this phenomenon is called dot gain [134]. During the printing the glyphs appears darker and larger than intended and its shape is distorted [167]. Dot

3. <https://en.wikipedia.org/wiki/Halftone>

gain is caused by spreading of the colorant on the medium or optical and electrostatic causes. In laser printers the deposition of toner within the area is strongly influenced by both the image value at the pixel and the image values of the neighboring pixels [167]. Several strategies are introduced for dealing with dot gain based on prediction of printed absorbance value for halftone image and for inverse halftoning process. Dot-gain is a nonlinear transformation that could be approximated by a piecewise-linear curve [134]. The toner distribution and dot shape vary from one printer device to another, therefore these characteristics are used for printer identification and authentication [95, 94]. The authors highlight that each dot is a random pattern whose shape depends on the technology, the setting of the printer, the ink quality and the paper properties [94].

### 3.2.4 Paper characteristics

The type of paper used also causes variations of the resulting printed image. Regarding ink absorption, the two paper types could be determined: coated paper and uncoated paper.

The most commonly used paper is uncoated paper. This paper type does not have a coating that is why it is not so smooth and tends to be more porous. With this paper, the ink diffuses into the fibers and causes a loss of resolution of the printed image texture. Coated paper is coated by a compound or polymer to impart certain qualities to the paper, including weight, surface gloss, smoothness or reduced ink absorbency. Coated paper is generally very smooth and can be either very shiny (high gloss) or have a subtle shine (matte)<sup>4</sup>. Coated paper is more resistant to dirt, moisture and wear. It also makes the printed material more shiny. Coating restricts the amount of ink that is absorbed by the paper and how the ink bleeds into the paper. To obtain a more accurate printing we use the coated paper since it has an additional layer on which the ink is fixed both by absorption and oxidation.

## 3.3 Scanning process

The scanner device optically scans images, printed documents, handwriting documents, or objects, and converts it to a digital image. Two mainly used scan technologies can be appoint: 1) a scanner that pulls a flat sheet over the scanning element between rotating rollers and 2) a scanner that illuminates the glass plane where an item is placed, and moves the scanner-head to read the item. The second type of scanners is called flatbed scanner. In this work we focus on this scanner and its characteristics.

---

4. <http://maconprinting.com/coated-and-uncoated-paper>

### 3.3.1 Flatbed scanner architecture

The scanning process using flatbed scanner is started by placement of hardcopy document on a glass plane (window). Then the page is illuminated by a bright light and the printed information (image or text) are reflected into a photosensitive element. A typical flatbed scanner consists of a motor, a scan-head, a lamp and a scanner bed, see Fig. 3.6.

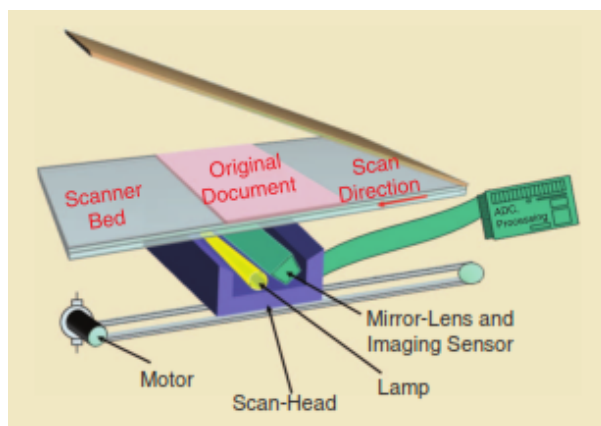


FIGURE 3.6: The flatbed scanner architecture [28].

A cold cathode fluorescent lamp, a xenon lamp or a lamp using LEDs is used to illuminate the document. The scan-head slowly moves, using a motor, to capture the image. It consists of a set of mirrors, lenses, filters and the imaging sensor. Most current scanners use Charge-Coupled Device (CCD) imaging sensors.

Analogically to the printing process, the physical characteristics of scanner process could effect the output digitized image:

- Velocity fluctuations of the motor's motion may lead to color registration variations in the scanned document [28].
- Manufacturing process of imaging sensors could also introduce various distortions that causes the noise impact into the output.
- Electronic photo capture may introduce variations into the digitized image pixels.

These physical characteristics add noise in the pixel values. This sensor noise could be of three types [28]. The noise of first type is caused by array defects (point defects, dead pixels, column defects, etc). That leads to large deviation in pixels values and now is easily corrected in devices. The second noise type is pattern noise that is caused by dark current and PhotoResponse NonUniformity (PRNU). Dark currents are stray currents from the sensor substrate into the individual pixels that vary from pixel to pixel. This variation is known as fixed pattern noise that is caused by differences in detector size, doping density and other matters trapped during fabrication. The PRNU is the variation in pixel responsivity that is presented in illuminated device. The PRNU noise is caused by variations among pixels, spectral response and thickness of coating. That does not lead to large pixel variations, that is why the correction is not applied.

The random noise components are the matter of the third noise type. This noise varies from frame to frame and cannot be removed by calibration. The third type noise is used in scanner forensics due to its statistical characteristics [28].

### 3.3.2 Scanner characteristics

The scanned images can undergo the rotation on a few degrees. In addition to a physical manufacturing imperfection, gamma correction and scanner resolution are the significant characteristics to digitized output image quality.

#### Scanner resolution

The scanner resolution determines the number of pixels scanned per inch of the document. This resolution also can be chosen by a user.

The resolution of the scanner is determined by the horizontal and vertical resolution. The horizontal resolution is determined by the number of elements in the linear CCD sensor. The vertical resolution is determined by the step size of the motor controlling the scan-head and the sensor data retrieval time [28].

The optical Modulation Transfer Function (MTF) plays an important role in scanner resolution [164]. The MTF has a characteristic form approximated by:

$$MTF_{scan}(f) = \left| \text{sinc} \left( \frac{f}{r_s} \right) \right|^n, \quad (3.1)$$

where  $r_s$  is the scanner resolution in pixels per millimeter,  $MTF_{scan}(f)$  is the magnitude of the modulation transfer function at spatial frequency  $f$ ,  $n$  is number of data points. For large  $n$ , both  $\text{sinc}(\cdot)^n$  and its corresponding spatial sensitivity function approach a Gaussian curve. Therefore, the authors model the optical scanner MFT as Gaussian blur [164]. In addition, the CCD sensors introduce the thermal noise and the dark current noise [164].

#### Gamma correction

Scanned images ought to be correctly displayed in a monitor, that is why a gamma correction procedure is applied during the scanning process. Each computer monitor has an intensity to voltage response curve which is a power function with parameter  $\gamma$  [133]. If a pixel should have intensity equal to  $x \in (0, 1)$  then the displayed pixel will have intensity equal to  $x^\gamma$ . The default value in most monitors is equal to  $\gamma = 2.2$ . In addition to printing process, the gamma correction introduces non-linearity too [133].

The scanned image must be digitized before storing and exploited on the computer. The digitization process invariably leads to quantization errors, that may be amplified due to nonlinear adjustment of gamma correction [134]. The digitization process does not affect a lot the visual quality of scanned image as human perception is not sensitive to this impact.

### 3.4 P&S process modeling

The P&S process modeling is now quickly developing field. Due to difficulties and randomness of this process it has not been yet suggested a perfect mathematical model of this process. However, multiple applications need to have a general P&S process model. In this section we discuss several proposed P&S models based on signal processing techniques and channel communication approach.

The purpose of a printer model is to accurately predict the gray level of a binary image produced by a printer. The authors in [155] suggest such a printer model that can be used by halftoning algorithms. They use a previously proposed physical model to train the adaptive signal processing model offline. Then this model is used to calculate the average exposure of each subpixel for any input pattern in real time.

In [141], a print-quality perception model has been proposed. This model uses an image analysis system and a neural network trained to be able to differentiate different print qualities.

The authors in [80] model the P&S process by considering the pixel value and the geometric distortions separately. According to this model, the distortion of pixel values is caused by the luminance, contrast, gamma correction, chrominance variations and blurring of adjacent pixels. This distortion introduces a visual quality change into the scanned image. The distortion of the geometric boundary is caused by rotation, scaling and cropping. That may introduce considerable changes at the signal level, especially on the DFT coefficients.

The pixel value distortion model for inkjet printers and flatbed scanners, proposed in [80], consists of a high-pass filter like a point spread function, a white normal random noise, a thermal noise and a dark current noise. The size of P&S image is usually different from the original, even if the printer and scanner resolutions are the same [80]. The authors consider the image geometric distortions to be an important factor and view it as an additional source of noise.

In [133], the authors view the image cropping as a cause of blurring in frequency domain. They model the P&S process on three main components: mild cropping, correlated high-frequency noise and nonlinear effects. The conclusions of the authors, based on observing the DFT coefficient magnitudes during their experiments for laser printers, are as follows:

- The low and mid frequency coefficients are less sensitive to P&S process than the high frequencies,
- The coefficients of the low and mid frequency bands with low magnitudes suffer from a much higher noise than their neighbors with high magnitudes,
- Coefficients with higher magnitudes have a gain of roughly unity,
- Slight modifications of the high magnitude low frequency coefficients do not add significant distortion to the image.



These observations suggest that the printing operation does not cause blurring, since several dots are used to print each pixel of a digital image [133].

As the halftone introduces the distortions to P&S image too, the authors in [70] proposed an accurate linear model for error diffusion halftoning. This model predicts the high-frequency noise and edge sharpening effects introduced by halftone. The halftone image  $s$  is obtained from a gray scale image  $x$  as:

$$s = Hx + Qw, \quad (3.2)$$

where  $H$  is a linear shift-invariant halftone (LTI) filter,  $Q$  is a LTI filter corresponding to the error diffusion coloring,  $w$  is white noise and  $Qw$  represents colored noise. The authors in [153] use this halftone model to represent halftone/inverse halftone printing channel by an equivalent channel shown in Fig. 3.7. In this channel communication approach, the  $X \sim \mathcal{N}(0, C_X)$  and  $Z' \sim \mathcal{N}(0, C_{Z'})$  are vector Gaussian signals and  $H_W$  represents a Wiener "restoration/equalization" filter.

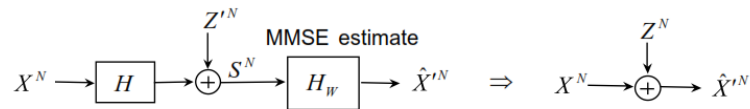


FIGURE 3.7: Equivalent halftone/inverse halftone channel from [153].

This model was constructed for inkjet printer with supposition that scanning process does not introduce any distortions.

The researchers also model the P&S channel as authentication channel [105]. The printing process at a very high resolution can be seen as a stochastic process due to the nature of printer characteristics [106] mentioned in Section 3.2.3. The authors simulate the printing process as a Generalized Gaussian distribution and log-normal distribution (that was proposed by [10]). For scanning process model the mentioned distributions are quantized and truncated.

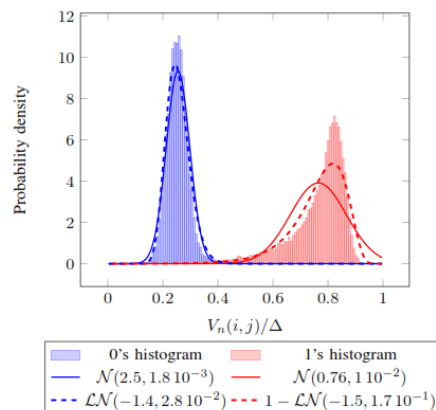


FIGURE 3.8: Normalized histogram of binarized images when the printed element is black (0) or white (1), compared with normal and log-normal distributions from [10].



Assuming the statistical properties of P&S channel, the authors in [10] experimentally show that the metric values of black/white block of pixels are related to random variables following asymmetric log-normal distributions, see Fig. 3.8.

### 3.5 Conclusions

In this chapter we highlight the characteristics of printer and scanner devices. Most of them impact to the image perception and quality of printed and/or scanned image.

Due to numeric-analog-numeric channel the printing and scanning processes cannot be separated, that is why in our contribution presented in the second part of this manuscript we talk about P&S distortions. These distortions are mainly caused to a random noise related to the quantification and thermal noises. The P&S process is often modeled by an additive or a multiplicative Gaussian process [164].

# Chapter 4

## Rich barcodes

### 4.1 Introduction

The barcodes are the most popular carriers that are able to survive the P&S process. The barcode is an optical machine-readable representation of data that is used to create the relation among data and object which carries it. The requirements that are set to barcode designers are a high density of encoded information, a reliability of reading process, a minimal cost of printing production and a minimal cost of reading equipment [102].

A lot of different types of barcodes are existing. They are separated into different classes depending on information storage dimension: we can find one dimensional (1D), two dimensional (2D) barcodes, as well as four dimensional (4D) barcodes.

In this chapter we focus on 2D barcodes with extended features which are called *rich barcodes*. The rich bar code is a barcode that has specific features (for example, data storage and/or identification), in addition to standard functionality. More precisely, we introduce the rich barcodes with visual significant property, with increased storage capacity, with hidden information embedded into the barcode and with automatic authentication property.

A short overview of standard barcodes is presented in Section 4.2. The following sections introduce the rich barcodes: user-friendly barcodes in Section 4.3, high density barcodes in Section 4.4 and barcodes with hidden data in Section 4.5. Afterwards, the graphical codes used for document authentication are discussed in Section 4.6. And we conclude this chapter in Section 4.7.

### 4.2 Standard barcodes

Today the barcodes could be found in products, plane and train tickets (for product and person identification), in museums, books and advertising billboards (for additional

information or web-page address). The most successful and well-commercialized barcodes were certificated by International Organization of Standardization (ISO). These barcodes can be easily read by a specialized devices or smartphone applications.

### 4.2.1 Barcode evolution

The one-dimensional (1D) barcode represents data by varying the widths and spacings of parallel lines. The 1D barcode is specified as a Universal Product Code (UPC) [1] that is used for tracking and tracing of trade items in stores. It is also used in tractability of baggage at the airports (Fig. 4.1) and logistic systems. However, the 1D barcodes have huge limitations of storage capacity. That could only be increased by increasing the number of barcode digits or by laying out multiple barcodes. But it leads to multiple scans in order to get all information contained in the barcodes that is not practical.



FIGURE 4.1: Example of 1D barcode into baggage ticketing.

In order to solve this limitation problem the barcodes that use geometric patterns (represented by any shape as squares, dots, triangles) in two-dimension (2D) have been suggested. There are a huge variability of 2D barcodes that differ by storage capacity and by principles of information encoding. The most well-known barcodes are Quick Response (QR) barcodes [4], Data Matrix barcodes [3], Portable Data File (PDF417) barcodes (that are not 2D barcodes, but stacked linear code with high storage capacity) [2] and Aztec barcodes [5]. The referenced 2D barcodes are illustrated in Fig. 4.2. The important features of mentioned barcodes are presented in Table 4.1.

Most of 2D barcodes have a small printing area, a high storage capacity, a quick reading process and an error correction capability, that is why the barcodes are successfully used in ticketing, labeling and administrative structures.

The generation process and code structure vary a lot from one barcode type to another. We chose the Quick Response (QR) code as a reference. In following section we discuss the QR code features, specificities of its structure and encoding/decoding process.



(a) QR code example



(b) Data Matrix code example



(c) PDF417 code example



(d) Aztec code example

FIGURE 4.2: Examples of high storage capacity barcodes a) QR code, b) Data Matrix code, c) PDF417 code, d) Aztec code.

Barcode characteristics		QR code	PDF 417	Data Matrix	Aztec code
Code type		2D barcode	Stacked barcode	2D barcode	2D barcode
Maximal	numeric data	7,089	2,710	3,116	3,832
	alphanumeric data	4,296	1,850	2,355	3,067
	binary data	2,953	1,018	1,556	1,914
	kanji data	1,817	554	778	-
Large capacity		yes	yes	yes	yes
Small printout size		yes	no	yes	yes
High speed scan		yes	yes	yes	yes
Error correction		yes	yes	yes	yes
Weak points		Reading of higher versions	Scanning device must be carefully aligned	Localization difficulties	Difficulties in generation for a layman user
Applications		Ticketing Advertising	Airline ticketing Postage	Product labeling	Transport Governmental

TABLE 4.1: Characteristics of several well-known barcodes: QR code, PDF 417 barcode, Data Matrix barcode and Aztec code.

### 4.2.2 QR code structure

The QR code was developed by DENSO WAVE corporation<sup>1</sup>. The target application for QR codes was the automotive industry in Japan. Therefore, this code was created to respect two features: 1) large storage capacity in a small area, and 2) high speed scanning and reading process. In addition, the QR code has four standard encoding models: numeric, alphanumeric, byte/binary and kanji. Thanks to these features the QR code has become very popular in many other application domains.

By default the QR code is a black-and-white image (or an image with two contrasting colors), where each information bit is associated to a module (black or white). The QR code consists of some particular patterns: position patterns, alignment patterns,

1. <http://www.qrcode.com/en/index.html>

timing patterns, format information patterns and version patterns, that offer high speed scanning process. The QR code structure is presented in Fig. 4.3. There are three position patterns in each QR code which have a specific form with the 1:1:3:1:1 dark-light ratio. These patterns are used for QR code localization. The alignment patterns are presented in any QR code from version 2. These patterns also have specific forms with the 1:1:1:1:1 dark-light ratio and are used for code deformation adjustment. The timing patterns aim to set the module coordinates of QR code. The format information pattern contains information about error correction level and the mask pattern used for symbol creation. The format information pattern is encoded with a BCH code [132] and is duplicated in the QR code. In the end, the QR code version and the error correction bits are stored in the version pattern. The version pattern is also duplicated in the QR code.

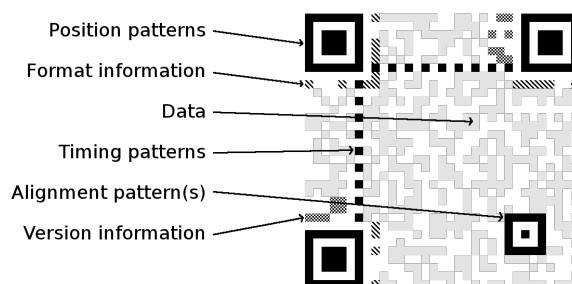


FIGURE 4.3: QR code structure with specific patterns.

There are 40 versions of QR code depending on storage capacity. The smallest QR code version is version 1, this has a  $21 \times 21$  module size, and a maximum admissible number of 152 data bits for the lowest correction level. The highest QR code version is version 40 which has a  $177 \times 177$  module size. The lowest correction level for version 40 allows to store a maximum of 7,089 data bits.

All stored information is encoded with the Reed-Solomon error correction code [132]. Each QR code version can store information using four error correction levels: 1) Low, which restores 7% of codewords; 2) Medium, which restores 15% of codewords; 3) Quartile, which restores 25% of codewords; and 4) High, which restores 30% of codewords. Logically the highest storage capacity corresponds to lowest error correction level.

**QR code generation algorithm** includes the following steps. Firstly, the input data is encoded with the Reed-Solomon code with error setting correction level. The bit streams are formed and divided into codewords which have an 8 bit length. The codewords form the blocks in which the error correction codewords are added. Then, the mask pattern is used for codeword masking. The target of this process is to balance the number of '0' and '1' occurrences in the QR code. The codewords are placed from the bottom-right corner to the top-left corner in a zigzag pattern. A more complex codeword placement is used for the highest QR code versions due to the alignment pattern presence and interleaved error-correction blocks. In the last step the function patterns (position, alignment, timing, format and version patterns) are placed into the QR code.

Several applications for QR code generation can be found easily on the internet.

**QR code recognition algorithm** consists of three steps: 1) binarization; 2) pre-processing; and 3) decoding algorithm. A grayscale image with rotated QR code is an input. First, the image binarization is performed to locate the black and white modules. Then, the position pattern localization is applied and we re-sample the QR code in order to put it in correct orientation. In the end the standard decoding algorithm is executed. The output is the extracted message.

### 4.3 User friendly barcodes

The barcodes are accused to be non-user friendly and have a non-aesthetic view, due to barcode construction using only two contract colors (usually black and white) and strict squared form. People can not determine the vendor, brand or purpose of the code just by looking at it. That attracts designer and research interests to create user friendly bar codes.

The designers have suggested "artistic QR codes" by replacing the square modules by rounds, triangles, dots (see Fig. 4.4.a-b), by using more colors and textures, by replacing modules with a logo (see Fig. 4.4.b). All these designer techniques exploit the QR code error correction capabilities. It weakens the correction capabilities and tends to non-optimal encoding rate [38].



FIGURE 4.4: Examples of design QR codes: a) Replacement of modules by logo<sup>2</sup>, b) Change of module shape, colors and logo insertion<sup>3</sup>.

In the same time, the image and signal processing researchers start to suggest the user friendly barcodes that does not use the error correction capacities for visual significance and aesthetics.

The visual significant QR code can be obtained by image blending of embedded informative image [8]. The method allows the insertion either brand logo or even a family photography in full color, Fig 4.5.a. This blending method is based on the fact that the QR code reading process starts by binarization step: the pixels with values  $p(i, j) \in [\lambda, 1]$  are considered as white pixels, and  $p(i, j) \in [0, \lambda[$  are considered as black pixels. Therefore, the authors modify the QR code pixels so that white module pixels are transformed from white to any RGB value whose luminance value exceeds  $\lambda$ , and similarly a black

2. [https://en.wikipedia.org/wiki/QR\\_code](https://en.wikipedia.org/wiki/QR_code)

3. <https://www.unitag.io/fr/qrcode>

module pixels are transformed to any RGB value whose luminance falls below  $\lambda$ . This image blending does not create decoding errors and does not use the error correction bits for image embedding.



FIGURE 4.5: Examples of user friendly barcodes using a) image blending technique [8], b) optimization blending technique [40], c) novel image-like barcode [38].

The HVS is more sensitive to edges but the barcode readers can manage with smaller size modules [40] to decode it. These properties are used to formulate an optimization problem that gives an efficient two-phase algorithm to generate aesthetic 2D barcodes. The first phase determines the bits that can be changed considering a minimal visual distance and the targeted sampled value for these bits. The second phase finds the sub-image minimizing visual distance for each module (the sampled value is the same as the value determined in the first phase). This approach also makes use of image blending technique but it incurs the additional bit-errors in order to improve the barcode aesthetics (see Fig. 4.5.b).

New image-like 2D barcodes that looks like recognizable image, Fig. 4.5.c, or logos but carry a reasonable payload, are suggested in [38]. This image-like 2D barcode is generated using the information theory perspective. The image-like 2D barcode is constructed using random dithering in which the pixel gray level is considered as the probability of assigning "1" in the code. The maximal size of hidden message is equal to  $proposed\ coding\ rate \times image\ size$  (in pixels). The maximal size of hidden message in the example Fig. 4.5.c is equal to 256 bytes. Nevertheless, this code does not have an error correction capacity, therefore it could not correctly survive the P&S process.

## 4.4 High storage capacity barcodes

The popularity and successful use of barcodes attract people to use it in new application scenario. Now we meet a huge need of barcodes with high storage capacity for personal identification and document authentication. The storage capacity may be increased by use of data compression techniques, bi-directional encoding techniques, colors and time dimension. Nevertheless, the increased density could cause reading problems, due to P&S impact mentioned in Chapter 3 and/or optic and system limits of capture device. In this section we introduce some high capacity barcodes that survive the P&S distortions.

#### 4.4.1 Black-and-white barcodes

In this section we focus on specific barcode constructions and present several black-and-white barcodes with high data density.

##### High capacity 2D barcode

The High Capacity 2D Barcode (HC2D) is a novel barcode that could store up to 24400 bits of compressed data [135]. This barcode can store both text and binary data. It is designed to reduce the visual impact to a printed document by removing classic size constraints (for example, the QR codes and the Data Matrix need a square area to be inserted into a document). An example of such HC2DB is illustrated in Fig. 4.6.

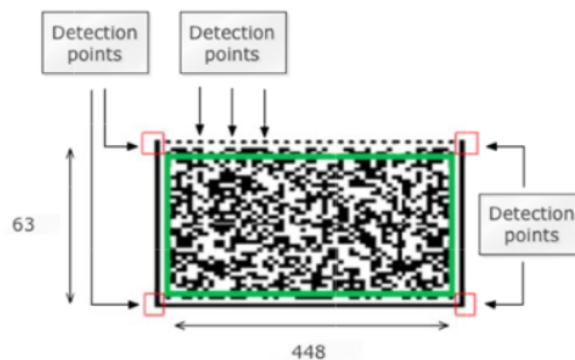


FIGURE 4.6: Example of HC2DB barcode [135].

The HC2D barcode consists of two vertical lines on the left and right of the barcode, a horizontal line on the bottom and a dash line on the top of the barcode. The top dash line is used for sampling column widths. And all the lines together allow to obtain four detection points in the corners of barcode, see Fig.4.6. The data area is a  $63 \times 448$  matrix where each module consists of one bit of data.

The header part of HC2D consists of format information: version, data type (numeric, ASCII or binary), compression option, checksum and data length. The rest of the HC2D contains the encoded data. The data is encoded using Reed-Solomon Error Correction Code (ECC) with two error correction levels: low - that restores approximately 7% of codewords, and high - that restores approximately 32% of codewords. The data encoded in the HC2D can be compressed that allows to increase the storage capacity.

##### HD barcode

A bi-directional two dimensional matrix barcode, called HD barcode, is suggested in [91]. This barcode encodes the data using the lines. A representative example of HD barcode in its smallest configuration is illustrated in Fig. 4.7.

The HD barcode in Fig. 4.7 consists of basic data blocks arranged in a rectangular or a square form. The number of data blocks in columns and rows must both be even. The smallest version of HD barcode consists of a  $2 \times 2$  superblock (formed from four basic



data blocks). The largest possible version contains a matrix of  $254 \times 254$  blocks. This largest version could carry 709,676 bytes of data when using 48 byte data groups and 4 bytes of error check bytes per data group [91].

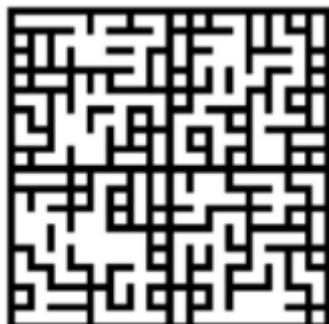


FIGURE 4.7: Example of HD barcode consists of four basic data blocks [91].

As it was mentioned, the data is encoded using lines with uniform thickness in both horizontal and vertical directions, thus it creates a bi-directional code. Each line represents a bit: if a line is presented, it represents a '1' in encoded sequence, otherwise it represents '0'. Each track consists of 8 pieces that encode a single byte. Each data block encodes seven bytes horizontally and seven bytes vertically for a total of 14 bytes per data block.

Among 14 bytes, twelve are used for data storage, the rest two are used for block error correction. The Reed-Solomon ECC with sixteen correction levels is put on. In addition to block error correction, the HD barcode employs group error correction.

The HD barcode presents high storage capacity and a strong error correction capacity due to its construction. However, this barcode is commercialized, thus can not be used for free.

#### 4.4.2 Colored barcodes

In addition, to previously presented high capacity black-and-white barcodes, the storage capacity could also be increased by using colors. However, the color barcodes have several problems due to:

- Color printing technology (see Chapter 3, Section 3.2.3);
- Difference of color balance in different code readers;
- Capture of barcode by inexperienced user that causes unconstrained barcode location, orientation and slope;
- Geometry distortions during the capture process;
- Light conditions that can vary a lot.

In this section we present the color barcodes that resist quite good the mentioned reading problems.

### Multilevel 2D barcode

The high-rate 2D barcode for P&S channel is introduced in [151]. An example of this barcode is illustrated in Fig. 4.8. The multilevel 2D barcode encodes data using multiple gray levels. Earlier, the four gray-level 2D barcode was suggested in [33].

The halftone cells are used as modules. Therefore, the 2D barcode is considered as a signal modeled by amplitude modulation. The storage rate of this multilevel barcode is equal to  $U = (r_p/a)^2 \cdot \log_2(a^2 + 1)$ , where  $r_p$  is printer resolution in dpi, and  $a$  is the length in dots of the side of a square halftone cell [151]. The authors mentioned that the storage rate of its scheme is 1403 *bytes/in*<sup>2</sup> at a bit error rate of  $2 \times 10^{-4}$ .

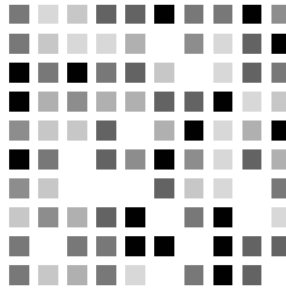


FIGURE 4.8: Multilevel 2D barcode [151].

This code is an alternative to black-and-white high capacity barcodes. One of applications for multilevel barcodes is the used for storage of document hash function [151].

### HCCB barcode

The High Capacity Color Barcode (HCCB) is introduced by Microsoft<sup>4</sup>. This barcode consists of triangles placed in rows separated by a white line. The triangles are of different colors that could up to eight, see Fig. 4.9. The number of rows in HCCB can vary but the number of modules in each row is always a multiple of the number of rows. Each module is a basic entity for storing the information.

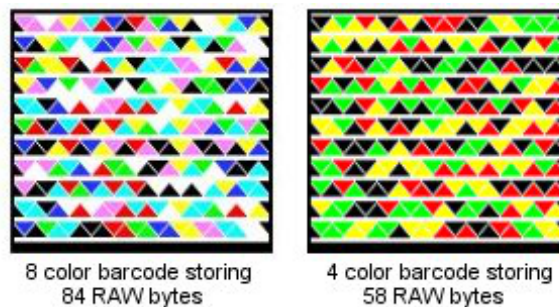


FIGURE 4.9: HCCB barcode examples with four-colored and eight-colored barcodes <sup>4</sup>.

As all barcodes, the HCCB has a specific structure, Fig. 4.10. To locate correctly barcode in an image: a black boundary around the HCCB, that is further surrounded by a thick white band, is added. The bottom black boundary is thicker in order to determine the

4. <http://research.microsoft.com/en-us/projects/hccb/>

barcode orientation. The last triangles in the last row have always the same fixed order (two triangles per color), that are used as a palette during reading process. The reading process starts by search of white border around the code, and continues by performing the alignment process by looking for the thick bottom boundary. Consecutive rows are separated by white lines.

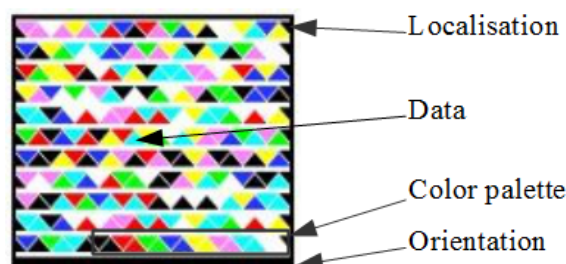


FIGURE 4.10: HCCB barcode example with eight-colored barcode <sup>4</sup>.

The Microsoft Laboratory publish the tests that have yielded using eight colors. The maximal mentioned storage capacity <sup>4</sup> is 2,000 binary bytes or 3,500 alphabetical characters per square inch using a 600 dpi business card scanner. The barcode can be printed using a regular inkjet or laser jet printer.

The elliptic curve cryptography and public key infrastructure techniques are integrated in the HCCB for digital signature and document authentication. Nevertheless, these techniques increase the size of the HCCB.

The weak points of HCCB code do it less robust than other 2D barcodes in the scanning process [119]:

- The automatic recognition is fragile, since the position detection could start from any image contained a white border inside (which is not the HCCB).
- The only one sample of palette makes it fragile to dirt, distortion or damages in the palette area.
- The row slop could be changed by P&S distortions that might result in failure recognition in case of too big impact.

Several of these localization and segmentation problems have been already solved in [100].

### HCC2D barcode

The color QR code, named High Capacity Colored 2-Dimensional (HCC2D) code [47], is designed to preserve the robustness and the error correction capacity of standard QR code. The HCC2D code increases the storage capacity by increasing the number of module colors: it could use up to 16 colors. The storage capacity is increased, in the same time the error correction capacity is kept the same as in standard QR code. Examples of HCC2D code with four and sixteen colors are illustrated in Fig. 4.11.

In order to keep the same strong points as standard QR code, the position patterns, the alignment patterns, the timing patterns, the version and format information are

preserved in the HCC2D code. In addition, the space required to these specific patterns is small. The modifications of these space may led to failures in the recognition process.



FIGURE 4.11: HCC2D barcode examples with four-colored and sixteen-colored barcodes [47].

The important changes are gathered in the data and error correction codewords areas. The modules of the HCC2D code may be of different colors with a palette composed at least 4 colors. That allows to store more than one bit in each module. The Bits per Module (BpM) can be defined as the number of bits that a single module is able to store:  $BpM = \log_2(\text{number of colors})$  [119]. That is why the more colors are used, the more data can be stored into the HCC2D code.

Due to the distortions added by P&S process, the reading process of colored barcode needs an additional information about color used. Therefore, a new specific pattern is introduced in the HCC2D code: the color palette pattern. This pattern is repeated in four different places (two times in the bottom-right corner and two times in the top-left corner of the HCC2D code), and it takes only two rows and two columns in the HCC2D code. An example of such color palette patterns is illustrated in Fig. 4.12. The color palette pattern ensures that the reading application is able to know how many and which colors are used in the HCC2D code generation [119].

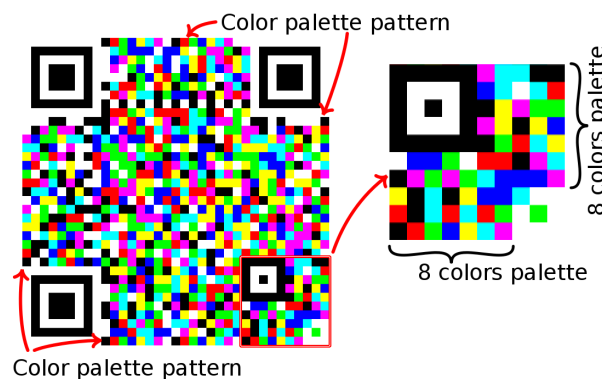


FIGURE 4.12: HCC2D color palette example with eight-colored barcode [119].

The recognition process of such a barcode starts by recognition of color palette. If the colors in the palette are not recognized as expected, the process terminates with a failure, otherwise the reading process continues. Since the colors are represented as vectors in multidimensional space, the Euclidian distance is used to solve the identification problem: the recognized color is given by the color in the palette, which minimizes the vector

distance [119].

The HCC2D barcode can be used to store facial biometric features [120], due to its high storage capacity. For example, the 4-ary HCC2D code with  $147 \times 147$  module size can store 24,656 bits, in comparison, the standard QR code V33 in corrected level M with  $149 \times 149$  module size can store 13,048 bits.

### COBRA barcode

The barcodes are attractive candidates for near field communication in many scenarios like contactless mobile advertisements, data exchange in real stores or museums [54]. This scenario means to display the barcodes in LCD displays and to transmit more information directly to user smartphone.

A novel visible light communication system, called COlor Barcode stReaming for smArt-phones (COBRA), is suggested [54] to encode information into specially designed 2D color barcodes. This 2D color barcode is designed to be optimized for real-time streaming between small-size screens and low-speed smartphone cameras.

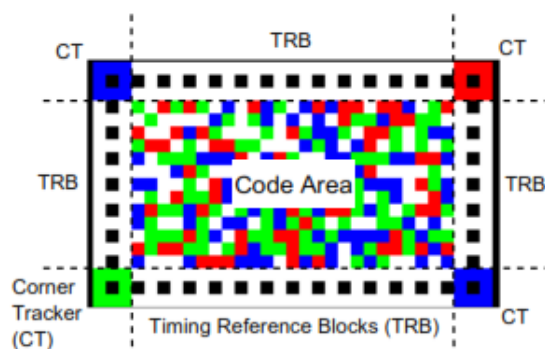


FIGURE 4.13: Example of COBRA barcode [54].

This barcode is formed by square color blocks of the same size as shown in Fig. 4.13. It consists of three area types: corner trackers (CT), timing reference blocks (TRB) and code area. The CT patterns are used to quickly locate the COBRA barcode corners. The CT patterns have a specific structure: a black block is surrounded by 8 blocks of the same color (red, green or blue). The color of CT pattern helps to determine the barcode orientation in the image: the green CT is placed at the bottom-left corner and the red CT is at the top-right corner. The other two corners have blue CT patterns. The TRB patterns are placed on the border of barcode to locate all color blocks in code area. The TRB pattern is a black block surrounded by 8 white color blocks that significantly accelerates the decoding process. The code area stores the encoded data.

Data (header, payload and CRC checksum) is encoded using a sequence of color blocks in code area. The COBRA barcode uses four colors (white, red, green blue) for data encoding. In general, the number of colors can be increased up to eight colors by adding complementary colors (magenta, cyan and yellow). More colors can not be used due to image blurring problems during reading process.

The four-colored barcode permits to encode 2 bits of data in each color block. And on a 4-inch phone screen with  $800 \times 480$  resolution, a single color barcode with 6-pixel block size contains 18.8K bits.

#### 4D barcode

The 4D barcode encodes data in four dimension: width, height, color and time [75]. This barcode can not be printed on paper but it is displayed on mobile or office screen. This 4D barcode can transmit much more information robustly to off-the-shelf mobile phones without requiring an explicit synchronization. An example of such a barcode is illustrated in Fig. 4.14.

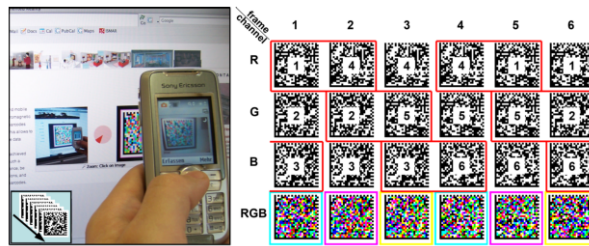


FIGURE 4.14: Example of 4D barcode [75].

To generate the 4D barcode the data is split into smaller parts that are encoded in a series of colored 2D barcodes. Then, the animated colored 2D barcodes are displayed in an endless-loop on screen, and are recorded by mobile camera phone. After that, the individual barcodes are extracted, assembled and decoded to reconstruct the embedded data [75].

The generator encodes three different 2D barcodes simultaneously into each frame of the displayed sequence. Each of them is encoded into the red, green and blue color channels - that will create a colored barcode. Each 2D barcode of the original sequence is embedded three times into the same color channel of three subsequent colored barcodes. To facilitate the detection process each colored barcode is surrounded by a colored border. The border color allows to detect which barcodes are encoded and which will be replaced in the next frame. An example of frame composition is illustrated in Fig. 4.14. The authors use the complementary border colors for indicating an upcoming barcode transition within a particular color channel [75].

The decoding process consists of preprocessing steps (frame capture and corner detection, image rectifying, contrast and brightness image adjustment, 2D barcode extraction in gray scale) and decoding step. Currently, the 4D barcode can transmit 1,400 characters per minute (23 characters per second). Experienced users are able to read this barcode with a success rate of 95%.

## 4.5 Data hidden barcodes

Thanks to popularity of barcodes two different approaches of data hidden techniques using barcodes were suggested last years. The first approach means of barcode use in nested image steganography [26]. As steganography is mainly suitable to protect nontext data, the authors suggest to embed the QR code containing text data and the face image into a cover image. The QR code with text data is the lossless secret data as the extracted data and original data do not have any distortions among them, thanks to QR code error correction capacity. And the face image is the lossy secret data that could suffer from distortions after embedding.

The second approach embraces the methods that conceal the secret data into the cover QR code without distorting the readability of QR content [83, 20]. These methods exploit the error correction capacity to conceal the secret into a cover QR code. The secret message is embedded so that the standard applications could read the QR code content but the secret message is decoded only by the validated receivers.

In [83], a secret message is encoded using key and then is randomly embedded into the QR code by insertion of additional errors, see Fig. 4.15.a. The secret message of length from 24 to 9,720 bits (depending on QR code version and error correction level) could be hidden using this method.



FIGURE 4.15: Examples of QR codes with hidden information (in both images red modules correspond to changes added to QR code): a) QR code V1, low error correction level, before and after message hidden by method [83], b) QR code V5, high error correction level, before and after message hidden by method [20].

Nevertheless, this hidden method has two drawbacks [20]:

1. If any bit of encoded message is damaged it will be impossible to retrieve the secret message.
2. If a transmission process or an attacker adds some extra bits into the QR code the secret message will not be recovered from the QR code.

In order to solve these drawbacks the robust message hiding method for QR code is suggested in [20]. The secret message is encoded using Reed-Solomon ECC in order to correct the erasure or additional errors. In addition, the list decoding [39] is used to improve the error correction capacity of Reed-Solomon code and to retrieve the secret message. An example of such QR code is illustrated in Fig. 4.15.b. This method is more robust against modifications and damage attacks, but it could embed in maximum 1,215 bits in the QR code V40 in high error correction level.



The main disadvantage of both methods is the use of QR code error correction capacities, as it make the standard QR code less robust against dirt and damage attacks. In addition, the experimental results have been performed only for numerical QR code without considering the P&S impact. Nevertheless, the highest QR code versions could suffer a lot from P&S distortions, we introduce these problems in Chapter 6.

There are also some approaches that embed an invisible watermark into the QR code image: the authors use the discrete cosine transform in [156] and the discrete wavelet transform in [138].

### Barcodes with secondary information

An approach that suggests the barcodes with supplementary information is also considered in some literature as data hidden barcode technique. These barcodes insert the supplementary information by changing the black barcode modules.

First reference technique consists on adding the supplementary reading direction to 1D barcodes [77]. These 1D barcodes have the primary information encoded in horizontally direction and the secondary information encoded in vertically direction. The secondary information can be encoded in one or multiple tracks. An example of such 1D barcodes is illustrated in Fig. 4.16.a. Thus, the supplementary information is also encoded by binary digits.

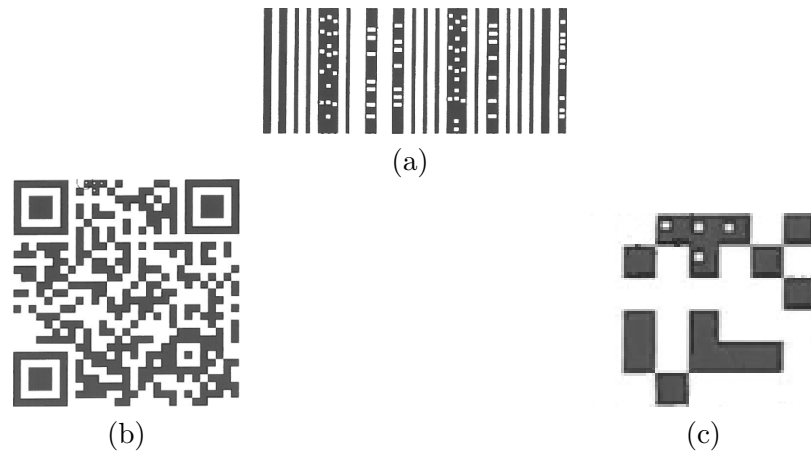


FIGURE 4.16: Examples of barcodes with secondary information: a) 1D barcode with two reading direction (horizontally and vertically)[77], b) QR code with hidden message using two modules with different chrominance [27] and c) zoom of QR code (b).

Second reference technique inserts the hidden information into black QR code modules [27]. The insertion is done by replacement of center part of black modules with square hidden codes, that creates the modules with different chrominance. The square hidden codes are located at the center part of the black modules of the QR code. The area of the hidden code is one-ninth of the area of black QR code modules. Therefore, these supplementary information is also encoded by binary digits. An example of this rich QR code is illustrated in Fig. 4.16.b.



## 4.6 Authentication graphical codes

One of the strong points of barcodes is robustness to P&S distortions. That feature makes them unusable in authentication applications, as printed document or product authentication. Nevertheless, the barcode advantages: the high storage capacity and cheap production, attract the researchers to look for rich barcodes sensitive to P&S process. In this section we aim to present graphical codes that are sensitive to copying process and could perform protection against forgeries and counterfeits. Here, the graphical code is the security pattern such as a barcode, a specific pattern or a device signature that could be used for document or product authentication.

### 4.6.1 Authentication system

The goals of secrecy and message integrity are different: encryption does not (in general) provide any integrity, and encryption should never be used with the intent of achieving message authentication unless it is specifically designed for this purpose [66]. In modern cryptography several high-level approaches exist for authentication transmission [16].

The first approach is called an **authenticated-encryption scheme** [16]. It uses the secret message  $K$  to authenticate a message  $M$ . Schematically, this approach is illustrated in Fig. 4.17. The sender applies an encryption algorithm  $\mathcal{E}$  to a message  $M$  and a key  $K$  to generate a ciphertext  $C$ :

$$C = \mathcal{E}(M, K). \quad (4.1)$$

The sender will transmit  $C$  to the receiver. The receiver, on receipt of  $C'$ , will apply some decryption algorithm  $\mathcal{D}$  to  $C'$  and  $K$ . The decryption output is either a message  $M'$  that is the original message  $M$ , or an indication  $\perp$  that  $C'$  can not be regarded as authentic:

$$\mathcal{D}(C', K) = \begin{cases} M', & \text{if } M' \text{ is authentic} \\ \perp, & \text{otherwise} \end{cases}, \quad (4.2)$$

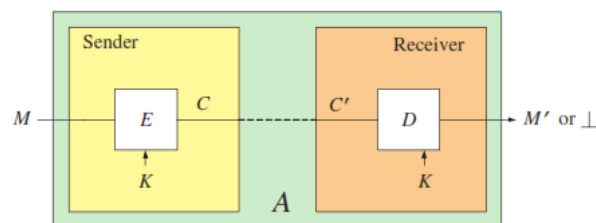


FIGURE 4.17: An authenticated-encryption scheme [16].

Since authenticity goal is not privacy, most often the transmitted ciphertext  $C$  simply consists of the original message  $M$  together with a tag  $T$ :  $C = \langle M, T \rangle$ . The second approach called a **message-authentication scheme** sends the ciphertext of this form

[16]. This authentication scheme is illustrated in Fig. 4.18 where  $TG$  is a tag-generation algorithm and  $VF$  is a tag-verification algorithm.

The tag-generation algorithm  $TG$  produces a tag  $T \leftarrow TG_K(M)$  from a key  $K$  and the message  $M$ . The tag-verification algorithm  $VF \leftarrow VF_K(M', T')$  produces a bit from a key  $K$ , a message  $M'$ , and a tag  $T'$ . If the output is the bit '1', then the receiver accepts message  $M'$  as authentic, otherwise the receiver rejects  $M'$ .

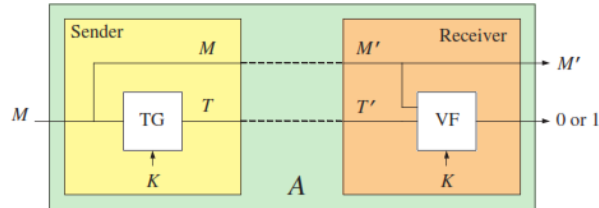


FIGURE 4.18: A message authentication scheme [16].

The third authentication scheme consists of the tag-generation algorithm  $TG$ , that is deterministic and stateless [16]. This scheme is called a **message authentication code** (MAC) and that is illustrated in Fig. 4.19.

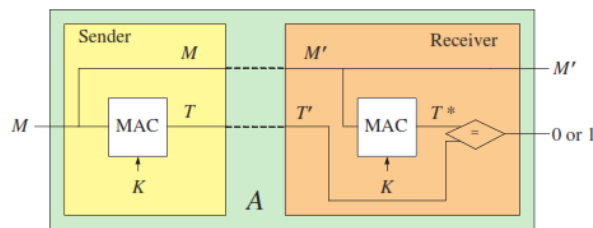


FIGURE 4.19: A message authentication code scheme [16].

The receiver, having received  $\langle M', T' \rangle$ , computes  $T^* = MAC_K(M')$ . If this computed-tag  $T^*$  is identical to the received tag  $T'$  then the receiver regards the message  $M'$  as authentic; otherwise, the receiver regards  $M'$  as inauthentic.

Document authentication scenario can adopt each of presented authentication schemes. An overview of general authentication system for valuable documents is illustrated in Fig. 4.20. The legitimate source generates a valuable document and a security pattern, and inserts this pattern into the numerical valuable document to ensure its genuineness. Then, the document is printed with high resolution by the legitimate source. During the authentication process, the receiver scans the hardcopy document and applies a verification test to the security pattern.

Most of counterfeits are produced in the interval between print process and scan process, since the opponent has only access to the printed protected document. As the P&S process can be considered as a PUF, see Chapter 3, the security pattern is hard to reproduce by the opponent. In addition the opponent has to scan, reconstruct, and reprint this valuable document, so the forged security pattern will differ from the legitimate security pattern.

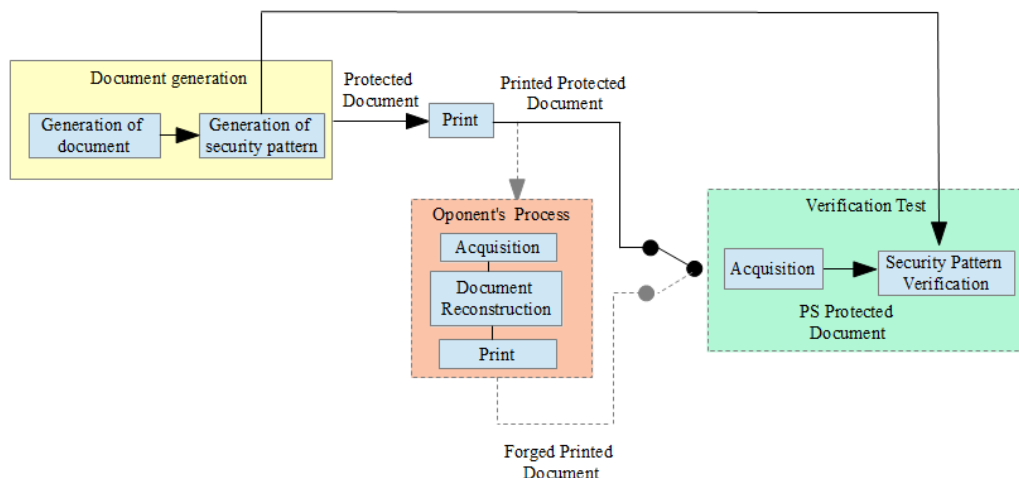


FIGURE 4.20: An overview of general authentication system using graphical codes.

## 4.6.2 Authentication scenario

While talking about document authentication, we should differ two types: document content authentication (integrity check) and document support authentication. The integrity check ensures that the data content has not been modified (or not). The document support authentication ensures that the document is not copied (distinguish original from copy). The brief descriptions of both document authentication approaches are presented below.

### Document integrity check

The document integrity check is based on the printed document hash value. This approach [154] consists in the generation of the document's hash based on the knowledge of a secret key  $K_H$ . This hash value is securely stored somewhere. During the integrity test the hash value is computed again from the document under investigation and compared with the one that was stored. The identical hash values declare the document integrity. While the document integrity needs to calculate the hash from the entire document, the document tamper-proofing is based on the concept of local hashing [150]. In tamper-proofing, a hash is calculated from each local part and could be used for identification of local malicious modifications.

Mainly three document content authentication scenario (for entire document integrity check and for document tamper-proofing) depending on where the hash is stored can be described [150]:

- Hash storage in an electronic database. The drawback of this solution is the necessity to have a direct access to the hash database [154].
- Hash storage onto the document itself using special element such as 2D barcodes, special inks, memory chips. The authors in [159] encode the compressed digital document signature into the barcode. The security of such approach is based on one-way and

collision resistant properties of hash function, and public-private key cryptography.

The drawback of this solution is related to specific aesthetics, storage limits and security issues [154]. The barcode can be easily copied if it was printed using ordinary inks. In addition, the use of barcode could not provide copy evidence verification.

- Hash storage onto the document’s content itself using data-hidden technique. This approach can be easily integrated into any text or image, it does not need access to database, it cannot be easily separated from the document and it provides a unique copy evidence [154]. The self-authentication of document by considering the combination of robust text hashing and text data-hiding technologies, is introduced in [150]. The drawback of this approach is the limited data storage capacity due to the document visible degradation and the physical factors of P&S process.

### Document support authentication

The document support authentication is based on the degradation of security patterns by the physical P&S process. Due to the physical defaults and the stochastic nature of the P&S process this interaction can be considered as a PUF [35].

Generally, PUFs are innovative circuit primitives that extract secrets from physical characteristics of integrated circuits (ICs) [136]. The two main properties of PUFs are:

- It is almost impossible to create a physical copy of PUF.
- It is impossible to create the exact mathematical model of PUF.

The PUFs can enable low-cost authentication of individual ICs and generate volatile secret keys for cryptographic operations [136]. The cheap enrollment, the non-invasive character of the protection, the easy and fast verification by non-experts make this protection scheme highly competitive and attractive for large-scale mass market applications [152].

Based on supposition that P&S is a PUF, the Copy Detectable Pattern (CDP) is introduced [108, 111]. A CDP is a maximum entropy image, generated using a secret key or password, that takes full advantage of information loss principle during P&S process (see details in Chapter 2, Section 2.4.2). An example of original numeric CDP and its degradation by P&S version are illustrated in Fig. 4.21.

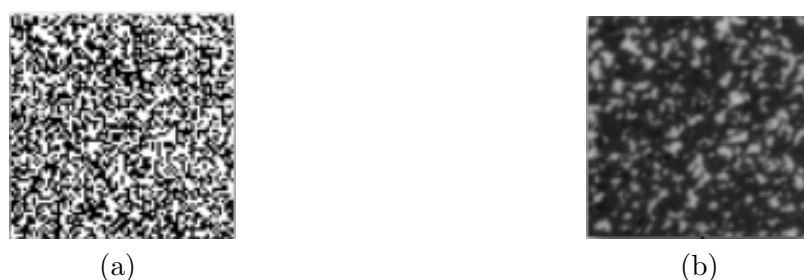


FIGURE 4.21: Example of a) Original numeric CDP and b) its degraded by P&S version [104].

The authentication test performs the comparison of the pixel values in the digital and in the scanned CDPs [108]. This comparison takes place in the spatial or frequency

domain and is based on a correlation, distance or a combined score of different features (for example: the authentication system performs a first correlation to determine if the captured CDP indicates the correct document identification and then if necessary, a second correlation to determine authentication by performing a 1 pixel shift multiple autocorrelation comparison of the two CDPs [50]). In addition to these copy detection metrics, the entropy metric, Fourier domain sharpness metric, Wavelet domain sharpness metric and prediction error metric have been suggested in [37]. The authors also highlight two attack scenario:

- Simple attack: an attacker can scan a CDP with a high resolution scanner and reprint it (possibly after image processing enhancements) with a high quality printer to create an illegitimate duplicate.
- Smart attack: an attacker may try to estimate the original CDP pixels utilizing inverse print-and-scan model. Then, he can generate genuine-like CDPs and copy CDP protected documents safely.

In most experiments the CDP samples resist to both attacks. However, it was shown that an attacker can produce a fake that successfully fools the detector with reasonable number of genuine goods [9].

This authentication problem can be presented as an optimization game between the legitimate source and an attacker where each player tries to select the best P&S channel to minimize/maximize his authentication performance [106]. For P&S process simulation the lognormal and general Gaussian additive processes [10] are used. The conclusions after studying this minimax game [104]: the opponent optimal parameters are close to the legitimate source parameters for both distribution families and the legitimate source can find a configuration which maximizes its authentication performance.



FIGURE 4.22: Examples of barcodes with authentication property [109]: a) DataMatrix with replaced black modules by CDPs, b) Zoom of DataMatrix (a).

The same type of images can be used to add the authentication property to standard barcodes [109]. The authors suggest to replace each black module of DataMatrix by the CDP image. This technique permits to read the standard barcode and to assure its authenticity. An example of such rich barcode is illustrated in Fig. 4.22. Additionally, authors mention, that we can encode the supplementary information in these CDPs by binary digits (represented by one pixel in CDP).

Another document support authentication approach is based on printer signature. The shape profiles define unique features of the non-repeatable print content and can be used

as a feature to represent the uniqueness of particular printed document [170]. These features are considered as printer signature and are used for document authentication. The security pattern illustrated in Fig. 4.23.a is used for document authentication.

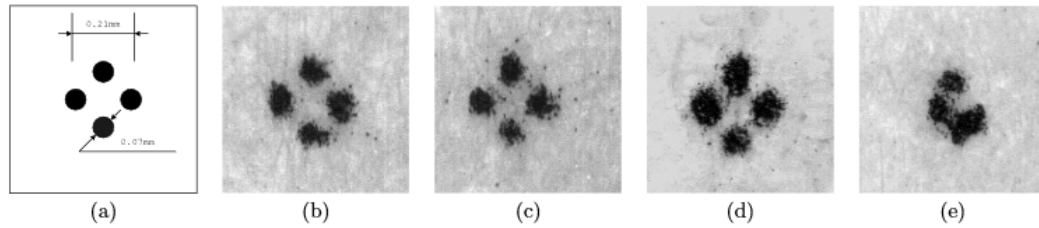


FIGURE 4.23: Proposed printer signature pattern [170]: a) original numeric security pattern, b-c) samples of security pattern printed using HP LaserJet 8100 at 600 dpi, d) sample of security pattern printed using HP LaserJet 4050 at 1200 dpi, e) photocopy of image (b) using a  $600 \times 600$  dpi digital photocopier Minolta Di152.

As it can be noticed from Fig. 4.23, even the security pattern printed twice using the same printer has different shape and distortions. The authentication process is based on the print signature feature extraction. For this, the security pattern is binarized and segmented, then the security pattern features based on dot shapes are extracted for shape matching test evaluation [170]. The experiment evaluations have been done for laser printers, but this authentication method can be extended to other printer types (offset or inkjet printer).

Recently, the document support authentication system based on a micrometric scan of a single printed dot is suggested [95]. The contribution of such authentication system is that a dot at the microscopic scale can be considered as an intrinsic signature of printer technology. The micrometric scan model of the printed document is based on the power exponential distribution and on an associated unsupervised identification algorithm. In this work the dot shape is also considered as an important parameter of printing technology. The experimental results show that such authentication system can pertinently be used for identification of original printing process [95].

The mentioned rich barcodes and graphical codes can be fused in order to ensure high document protection. For example, the authors in [110] suggest to use digital watermark, 2D barcode, personal biometrics and copy-detection pattern to protect the ID document against forgeries (see Fig. 4.24.a). An example of a protected e-Ticket is shown in [170], that consists of proposed secure pattern, landmarks, 2D barcode with document digital signature and a ticket serial number (see Fig. 4.24.b).

## 4.7 Conclusions

Thanks to their strong points the barcodes unhesitatingly enter in our daily life. Nevertheless, due to data storage limitations, robustness to P&S process and its boring view, the barcodes can not be used for designer purposes, for personal and document



**Part II**

**Contributions**





## Chapter 5

# Textured patterns used for document authentication

### 5.1 Introduction

As presented in Chapter 1, the main purpose of this thesis is to find a printable solution to protect document. This considered solution consists of an image that is sensitive to P&S process. We call this image a security element. In this chapter, this security element with embedded message is presented.

The aim is to embed a message (visible or invisible) into a security element, that is barely visible, but detectable after P&S process. This security element is used for authentication of document support, thus it ought to be sensitive to P&S process. This security element has to contain information: the pixels of different nature should be used in order to be detectable and perceptible after P&S distortions.



FIGURE 5.1: Dot shape modulation screen details (BrainBlock) [149].

The inspiration of such a security element is the Dot Shape Modulation (DSM) technique [149], that is one type of printed security patterns mentioned in Chapter 2. An example of DSM technique is BrainBlock, illustrated in Fig. 5.1, where the high density screens and low density screens differ in shape according to the gray wedge. This technique produces a uniform area in printing process. But copying systems make differentiation between these screens and the cover carried image becomes visible in a copy.

Therefore, we suppose to embed the message in security element using specific textured patterns. These textured patterns have several properties in order to give good recognition results after P&S process and to be hard reproducible due to P&S impact. The

embedded message in a legitimate image should be readable after pattern recognition process. The legitimacy of this security element can thus be verified. In this chapter, we discuss the characteristics of textured patterns in Section 5.2, present several types of security elements in Section 5.3, overview possible detection methods in Section 5.4 and show some experimental results in Section 5.5. Finally, we conclude in Section 5.6.

## 5.2 Textured pattern description

The suggested textured patterns have particular construction characteristics and specific response to P&S process. The combination of textured patterns with both features produces security elements, where the patterns are different and viewed as a uniform element. Therefore, an opinion poll has been performed in order to determine the textured pattern combinations, that create distinguishable and uniform security element.

### 5.2.1 Experimental analysis of characteristics

The P&S process impacts a lot on any printed document or image, as it is discussed in Chapter 3. The textured patterns undergo structure, form and color changes during P&S process. In this section we present some experimental results that show the distortions added by P&S process in real life.

#### P&S impact into textured patterns

In order to visualize the impact added by P&S process, we use a numerical microscope to observe two types of documents: printed document and copied (printed-scanned-printed) document. We use a numerical microscope XCSOURCE to visualize the P&S impact. These patterns have size  $12 \times 12$  dots and are printed and copied in 600 dpi resolution. Several samples are illustrated ( $22\times$  zoom) in Table 5.1.

We note that the internal textured structure of patterns is lost, the images become blurred and the colors are changed.

Remark: In this chapter, all examples of textured patterns are symmetric. However, the symmetry is not necessary and currently is not used. In addition, it is undesirable, as the symmetric structure needs less time for reconstruction. For reconstruction a fully symmetric pattern  $r \times r$  pixels, we need to reconstruct only  $r/4$  pixels of textured pattern. That means we need only  $2^{r/4}$  tries. Due to this remark the symmetric textured patterns are not used to any security element generation.

#### Change of colors

It is possible to compensate some color changes by using a color Look-Up-Table (LUT), dedicated to each printer-scanner pair. Such a LUT can be constructed by measuring the color changes after P&S process. An example of LUT is presented in Fig. 5.2.

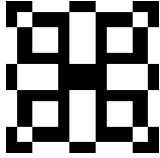
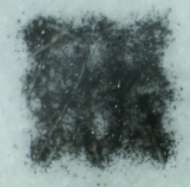
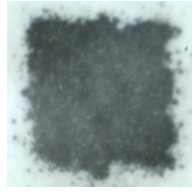
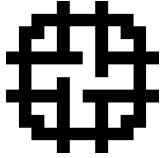
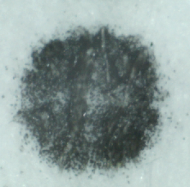
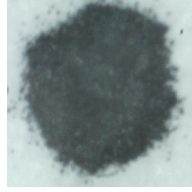
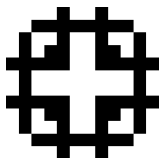
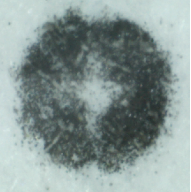
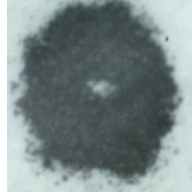
Original pattern	Printed pattern	Copied pattern
		
		
		

TABLE 5.1: Textured pattern response to printing and copying processes visualized with microscope (22 $\times$  zoom).

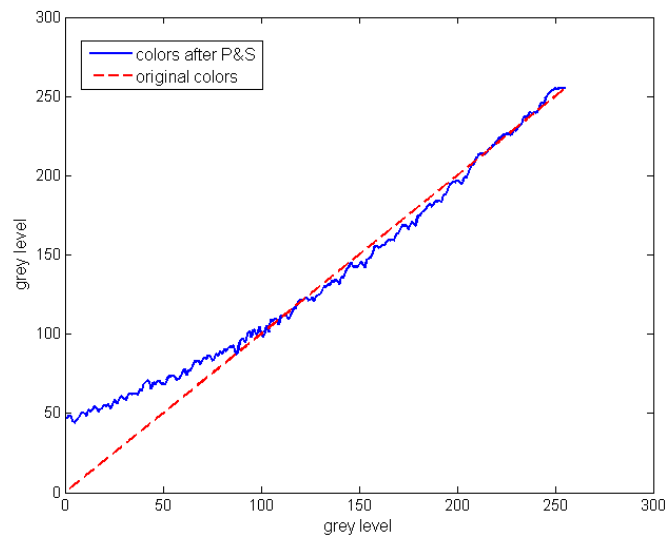


FIGURE 5.2: Example of color changes after P&S process. Red line: original colors, blue line: colors after P&S process.

We note from the Fig. 5.2 that an image loses especially dark colors after the P&S process. The LUT can correct some color defects occurred due to P&S process. However,

it is impractical to construct such a LUT for each printer-scanner pair and each type of paper.

### Grey level value randomness during P&S

In order to show the randomness of P&S process, the following experiment have been performed.

**Experiment setup:** We took more than 150 P&S samples of the same textured pattern, and compare the grey level values of the same numeric black pixel. We chose the pixels in the top, center and bottom of textured pattern, and illustrate the pixel value variation in Fig. 5.3.

This experiment shows that the same pixel value varies a lot from one print to another. In addition, using graphics in Fig. 5.3.b - Fig. 5.3.g, we note that the black pixels can be represented by gray value in the interval (60, 160). Therefore, the black pixels can sometimes be considered as white, and vice versa. Since the binarization threshold is difficult to identify, standard global thresholding methods are very hard to apply.

### Binarization of P&S sample

To consider the random distribution of black and white pixel values after P&S process, and to evaluate the ability of binarization algorithm in textured pattern recognition, we again decide to provide an experiment.

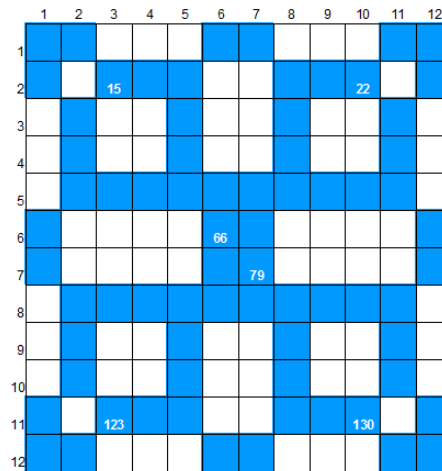
**Experiment setup:** We took P&S samples of first and third textured patterns appeared in Table 5.1.

1. We know the ratio  $b$  of black pixels.
2. The  $b$  pixels with smallest grey level values are considered as black pixels, the rest of pixels  $r^2 - b$  are considered as white pixels.

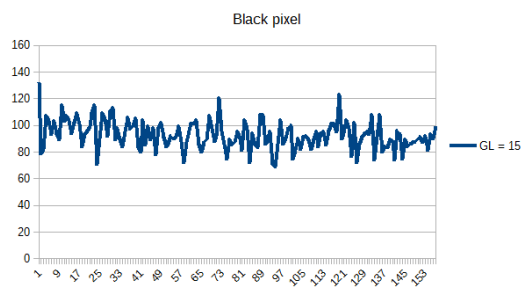
The results of such binarization method are illustrated in Fig. 5.4. We can observe the random effect of P&S process and also the lost of pattern textured structure.

Due to all these distortions, the reconstruction of such textured patterns after P&S process becomes a difficult problem. This experiment shows us that the binarization methods are not adapted for detection or reconstruction of such textured patterns.

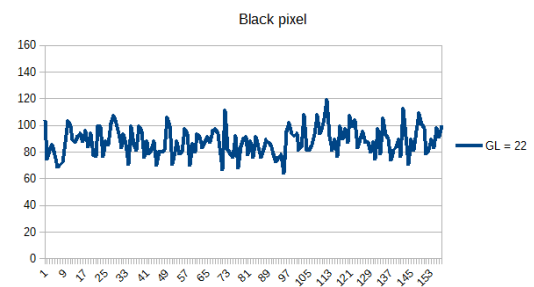
Remark: Other binarization strategies are possible, nevertheless due to high randomness impact of P&S process and tiny size of textured patterns, the probability to obtain much better results is quite small.



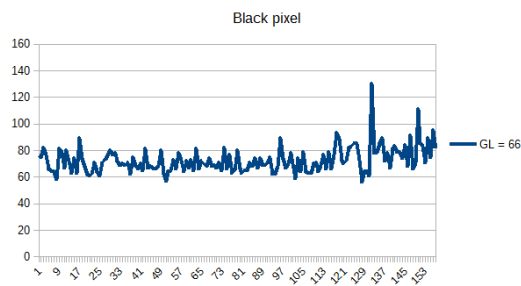
(a) Textured pattern and considered pixels



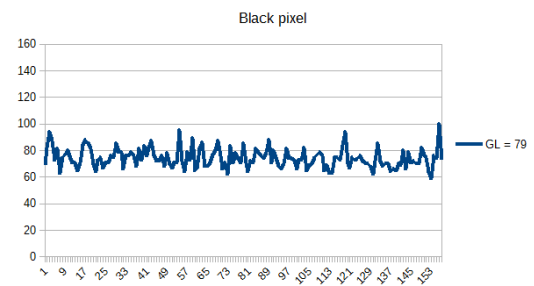
(b) Pixel number 15



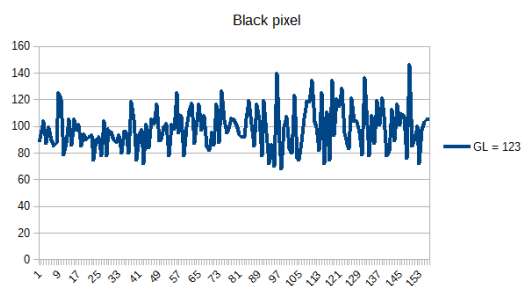
(c) Pixel number 22



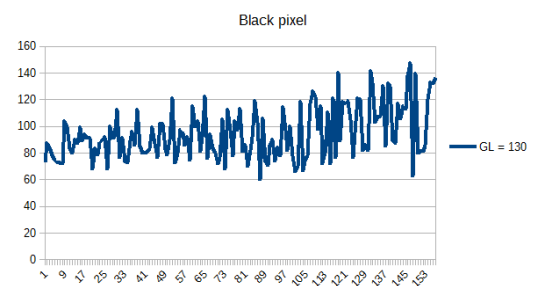
(d) Pixel number 66



(e) Pixel number 79



(f) Pixel number 123



(g) Pixel number 130

FIGURE 5.3: Distribution of radiometry value of several pixels over 150 samples.

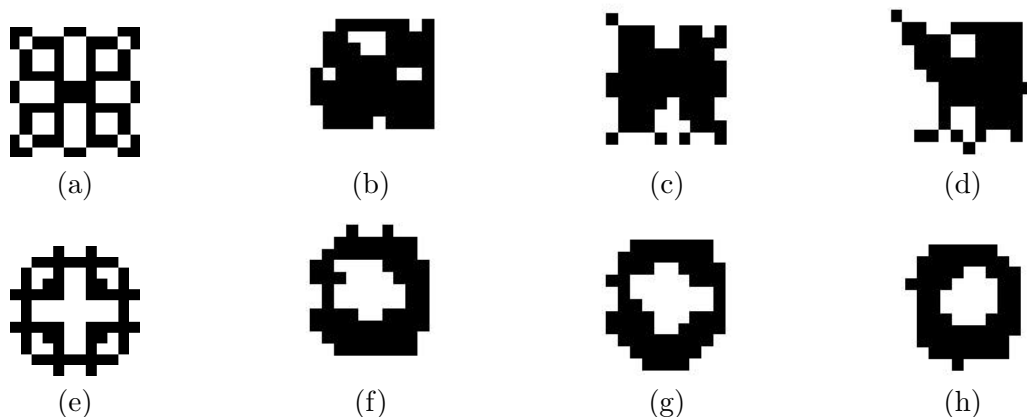


FIGURE 5.4: Examples of a) and e) Original textured patterns, b),c),d) Binarized versions of (a) and f),g),h) Binarized versions of (e).

### 5.2.2 Mean opinion score of human perception

In order to determine the combinations of textured patterns, that produce distinguishable and uniform security element for HVS, we conduct an opinion poll. This questionnaire helps us to determine several combinations of textured patterns, that could be used for security element construction. In addition, these textured pattern combinations have been analyzed then, in order to determine the criteria to identify an optimal textured pattern combinations for construction of security elements (see Section 5.2.3).

**Experiment setup:** We have prepared 120 security elements using 16 textured patterns (the combination of every two patterns creates  $16 \times 15$  possibilities, we delete the elements that were created using the same textured patterns, i.e. we obtain  $(16 \times 15)/2$  combinations). Each security element contain two textured patterns: first pattern is used for background, second pattern is used for writing a visual message (combination of letters and numbers). These security elements are placed in one sheet and printed in resolution of 600 dpi. Then, the respondents have been asked to answer two questions:

1. What characters do you recognize on each security element?
2. What is the perceptual level of each security element in the interval  $[0, 10]$ , where level 0 corresponds to "I can not recognize anything", level 10 corresponds to "I see perfectly all characters"?

An example of respondent answers is illustrated<sup>1</sup> in Fig. 5.5.

Thirty persons take part in this opinion poll in age range 18 – 50 with 25% of female respondents, who work in research, technical and marketing areas.

**Observation results:** After getting 30 questionnaires, we perform the analysis of obtained human opinion score about data perception.

First of all, we build the plot with dependence of human perception evaluation (answers

<sup>1</sup>. Security elements are shown as examples and do not respect the original requirements (page A4, printing at 600 dpi)

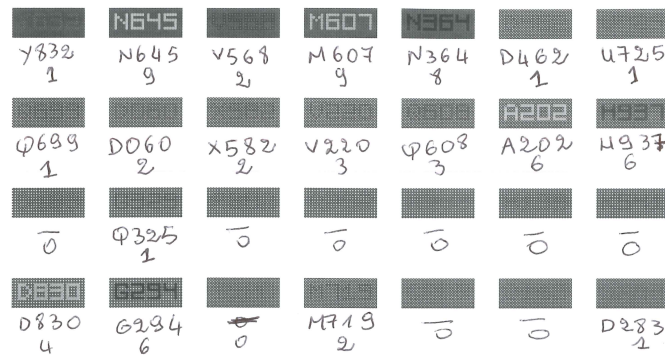


FIGURE 5.5: Example of respondent answers during opinion poll.

to question 2) and number of reading errors (that is calculated respecting the answers to question 1). This dependence is illustrated in Fig. 5.6<sup>1</sup>, where every point corresponds to mean value among all respondent answers to human perception evaluation and number of reading errors for each security element.

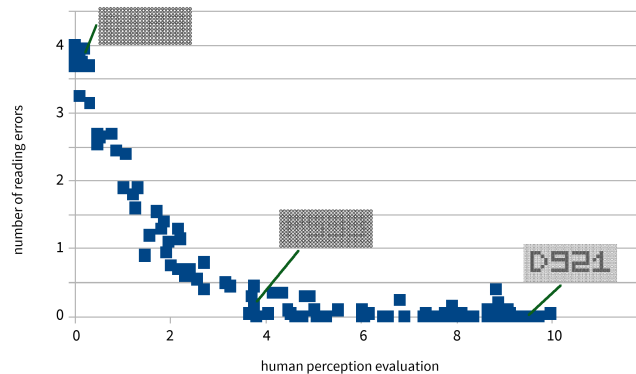


FIGURE 5.6: Dependence of human perception evaluation and number of reading errors.

The question 2 shows an evaluation of the respondent visual perception of each element. This perception evaluation can greatly vary from one person to another. These variations can introduce a bias in next statistical exploitation of such data. The question 1 produces reliable values by directly considering the number of reading errors of each respondent. The coherence of data shown in Fig. 5.6 allows us to provide the perception evaluation with accurate and reliable curve.

Additionally, this plot clearly separates the security elements into three groups: security elements with indistinguishable patterns, security elements with semi-distinguishable patterns and security elements with distinguishable patterns.

Schematically this separation can be done as shown in Fig. 5.7.

The conclusions from this mean opinion score are:

- the security elements with indistinguishable patterns correspond to our targets and can be used to embed imperceptible message. However, the textured patterns in



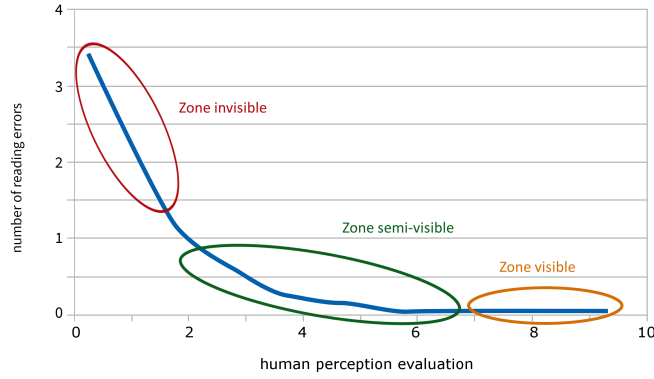


FIGURE 5.7: Zones depending on security element visibility.

these security elements are not distinguishable one from another after P&S process, therefore the legitimacy verification is impossible.

- the security elements with distinguishable patterns can be used for visible message embedding. Additionally, these elements resist P&S distortions. Nevertheless, due to facilities of perception and reconstruction, these security elements can not be used for legitimacy verification too. The complexity of reconstruction process must be studied, but the clustering of patterns is easier in distinguishable case. Thus it make easier the identification of representative candidates for each class that can be used to produce forged security elements.
- the security elements with semi-distinguishable patterns are the best candidates as we can embed difficulty perceptible message into these elements, in addition thanks to textured pattern differences, the recognition process can be effective. Therefore, the verification of security element legitimacy can be performed successfully.

In order to characterize the obtained results of human perception, we calculate the textured pattern spectra (i.e. DFT coefficients). Then, we calculate the quadratic distance between the spectra of both patterns constituting each security element (using following formula:  $\frac{1}{n^2} \sum_{i=1}^n \sum_{j=1}^n (\phi_1[i, j] - \phi_2[i, j])^2$ ). This distance is used to characterize the distinguishable capacity of pattern combinations. The dependence of human perception evaluation and distinguishable capacity is illustrated in Fig. 5.8. This figure shows that the security elements with indistinguishable patterns are obtained when the textured patterns spectra are closed, and the security elements with distinguishable patterns are obtained from the patterns with high distances between spectra.

### 5.2.3 Textured pattern combinations

Based on the opinion poll, we want now to determine the criteria for "machine perception" evaluation, that drives the behavior of the security elements after the P&S process. Additionally, we want to chose automatically the textured patterns, that can be used for security element generation.

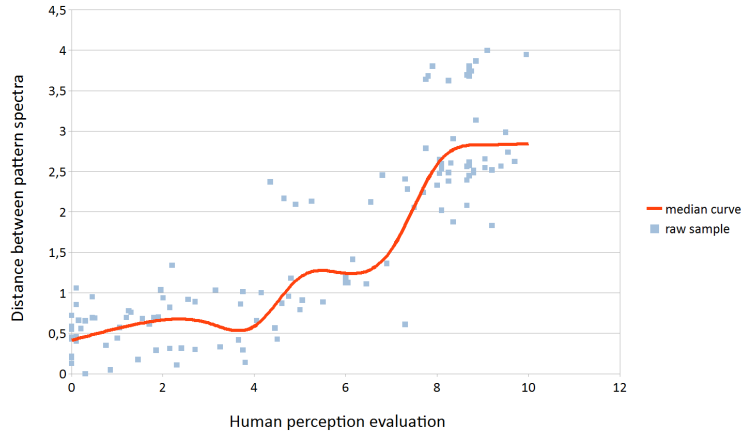


FIGURE 5.8: Dependence of human perception evaluation and pattern distinguishable capacity.

Due to the specific properties of the textured patterns, some of them, after P&S process are indistinguishable one from another. We need to choose the  $q$  patterns among  $Q$  patterns from database that can be distinguished automatically by application when they are placed on the same security element  $I$ .

In order to select  $q$  necessary patterns for textured image construction, we printed and scanned all patterns from our database. We call  $S_l$  ( $l = 1 \dots Q$ ) the patterns obtained by the P&S of patterns  $P_l$  ( $l = 1 \dots Q$ ). Then, we use the correlation measure to characterize the detectability of patterns.

The **Pearson correlation** between a pattern  $P$  and a pattern  $S$  is defined by:

$$\text{cor}(P, S) = \frac{\sum_i \sum_j (P^*(i, j))(S^*(i, j))}{\sqrt{\sum_i \sum_j (P^*(i, j))^2} \sqrt{\sum_i \sum_j (S^*(i, j))^2}}, \quad (5.1)$$

where  $P^*(i, j)$  (rsp.  $S^*(i, j)$ ) are the central values of  $P$  (rsp.  $S$ ) defined by  $P^*(i, j) = P(i, j) - \mu_P$  (rsp.  $S^*(i, j) = S(i, j) - \mu_S$ ) with  $\mu_P = \frac{1}{k} \sum_i \sum_j P(i, j)$  (rsp.  $\mu_S = \frac{1}{k} \sum_i \sum_j S(i, j)$ ).

The correlation measure has been chosen due to its sensitivity to P&S process, additionally, the correlation is often used in security pattern verification (see Section 2.4.2).

The selection of the  $q$  patterns has to satisfy the two following criteria:

1. Each pattern is better correlated with its degraded (by P&S) version than with all other degraded (by P&S) versions of selected patterns:

$$\forall l, l' \in \{1, \dots, q\}, l \neq l', \text{cor}(P_l, S_l) > \text{cor}(P_l, S_{l'}). \quad (5.2)$$

2. Degraded version of each pattern is better correlated with its original pattern than with all other original selected patterns:

$$\forall l, l' \in \{1, \dots, q\}, l \neq l', \text{cor}(P_l, S_l) > \text{cor}(P_{l'}, S_l). \quad (5.3)$$

Thus, the generation of textured image with semi-distinguishable patterns is obtained by combination of patterns satisfying the conditions (5.2) and (5.3). Remark: It could be interesting to extend these criteria considering the patterns after copy. Nevertheless, that will increase significantly the pattern database and it moves us away from our goal to make choice using only original patterns.

#### 5.2.4 Textured pattern characteristics

After all provided experiments, we conclude that the choice of textured patterns has to be done respecting two features: human perception and automatic decoding (i.e. P&S sensitivity). Respecting these two features, we can vary the security element characteristics from security element with indistinguishable patterns, that are damaged after P&S process to security element with distinguishable patterns, that survive P&S process.

The suggested security element contains of semi-distinguishable patterns, that have following characteristics:

- **Constant square size  $r \times r$  pixels.** The size constraint arises due to security element construction and facilities during recognition process.
- **Binary images.** Due to the halftoning procedure during printing process (mentioned in Chapter 3), we suppose that the production of pure black color is less random, than the production of any other gray scale color. In addition, the majority of document optical security patterns are binary.
- **Constant ratio of black pixels** or pattern density ( $b = const$ ). We set this constraint to reduce the space of possible textured patterns.
- **Spectrum related among them.** The security element contains only the textured patterns that are slightly distinguished one from another, thanks to this characteristic.
- **Combination feature.** The correlation value between original pattern and its degraded version must be high, i.e. the pattern combination satisfies the criteria (5.2) and (5.3).

### 5.3 Textured image generation

The proposed security element is called textured image. This textured image is an image  $I$  contained textured patterns that allow us to store a visual message. As it was mentioned before, the combination of textured patterns is defined experimentally. In this section, the three types of textured images are introduced.

The textured image contains the visual message and is created by using two textured patterns:  $P_1$  is used for background and  $P_2$  is used for writing a visual message (i.e.  $q = 2$ ). These patterns satisfy the criteria (5.2) and (5.3). The size of textured image  $I$  is  $r_a \times r_b$  patterns, where  $r_a$  and  $r_b$  are integers. We suggest three different types of

textured images, each of them aims to contain a visual message, that is hardly perceptible by human eyes.

### Textured image with visual message

Examples of textured images are illustrated in Fig. 5.9. At the beginning, we define the visual message (Fig. 5.9.a). Then, we choose the patterns  $P_1$  and  $P_2$  used for textured image generation. The last step consists in assembling the patterns respecting the visual message. For background we use pattern  $P_1$ , for characters we use pattern  $P_2$ , see Fig. 5.9.b. Any chosen patterns can be used for background, the other one is used for visual message.

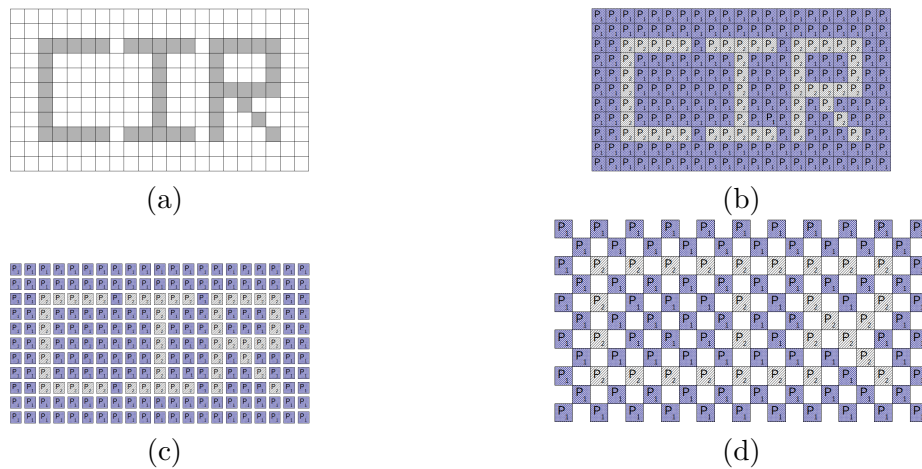


FIGURE 5.9: Generation of textured image containing the visual message: a) Binary visual message, b) Textured image containing the visual message (a), c) Textured image with white lines and columns containing the visual message (a), d) Chessboard textured image containing the visual message (a).

### Textured image with white lines and columns

This textured image is created using the same technique as "Textured image with visual message", however additionally white lines and columns are added among each two textured patterns. An example of such a textured image is illustrated in Fig. 5.9.c. The number of white lines and columns is constant and chosen by creator. The more white spaces is added, less perceptible visible message is.

### Chessboard textured image

The last proposed security element is the chessboard textured image, illustrated in Fig. 5.9.d. This textured image is created by deleting a half of textured patterns in textured image from Fig. 5.9.b respecting the chessboard order. This type of security element creates almost imperceptible visual message, even in the case when we use distinguishable combination of textured patterns.

The last two techniques make the visual message of textured images Fig. 5.9.b less perceptible due to specifics of HVS. The HVS lost the resolution, when the borders between textured patterns are not clear.

## 5.4 Pattern detection after P&S process

The original image  $I$  is made of a combination of  $q$  patterns arranged in a  $r_a \times r_b$  grid (see Fig. 5.9.b). Let  $I_s$  be the textured image obtained after P&S. Due to the random variations discussed in Section 5.2, the patterns are modified by the P&S process. Therefore, several statistical measures are used to create the representative candidates for P&S textured patterns. We suggest several classification methods to detect each pattern: 1) by correlating their representative candidates  $C_l$  with selected patches of  $I_s$ , 2) by using clustering classification, 3) by using the Hamming distances between representative candidates  $C_l$  and selected patches of  $I_s$ , and 4) by characterization Kendall correlation.

### 5.4.1 Representative candidates

First of all, we should introduce the suggested representative candidates. We experimentally estimate the representative candidates  $C_l$  ( $l = 1, \dots, q$ ) by repeating  $t$  times a P&S process of each of the  $q$  patterns, thus obtaining a set of  $t$  P&S patterns  $\mathcal{E} = \{E_l^1, \dots, E_l^t\}$ . We then propose to use as a representative candidate:

- the *mean images* obtained by averaging the  $t$  P&S patterns:

$$C_l = \text{mean}(E_l^1 \cdots E_l^t); \quad (5.4)$$

- the *median image* obtained by calculating the mean value of every  $t$  P&S patterns:

$$C_l = \text{median}(E_l^1 \cdots E_l^t); \quad (5.5)$$

- the *maximal image* obtained by calculating the maximal value of every  $t$  P&S patterns:

$$C_l = \text{max}(E_l^1 \cdots E_l^t); \quad (5.6)$$

- the *minimal image* obtained by calculating the minimal value of every  $t$  P&S patterns:

$$C_l = \text{min}(E_l^1 \cdots E_l^t); \quad (5.7)$$

- the *original image* obtained by replacing the representative candidates with original patterns:

$$C_l = P_l. \quad (5.8)$$

### 5.4.2 Pattern comparison measures

One of the most classical pattern comparison techniques is the use of correlation measure. We suggest to use one of three below mentioned correlation measures.

The **Pearson correlation** is calculated from formula (5.1).

The **Spearman rank correlation** (named also Spearman's rho) is the second most popular bivariate correlational technique [122]. Suppose we have two random variables  $P = p_1, \dots, p_r$  and  $S = s_1, \dots, s_r$ , that are converted to ranks:

$$\text{Rank}(p_1), \dots, \text{Rank}(p_r)$$

and

$$\text{Rank}(s_1), \dots, \text{Rank}(s_r).$$

The smallest element has the rank score 1, the biggest element has the rank score  $r$ . The Spearman's rho for pairs

$$[\text{Rank}(p_1), \text{Rank}(s_1)], \dots, [\text{Rank}(p_r), \text{Rank}(s_r)]$$

is computed from formula:

$$\rho = 1 - \frac{6 \sum d_i^2}{(r^3 - r)}, \quad (5.9)$$

where  $d_i = \text{Rank}(p_i) - \text{Rank}(s_i)$  is the difference between the ranks.

The **Kendall rank correlation coefficient** evaluates the degree of similarity between two sets of ranks given to the same set of objects [122]. Analogically, we have two random variables  $P = p_1, \dots, p_r$  and  $S = s_1, \dots, s_r$ . Any pair of observations  $(p_i, s_i)$  and  $(p_j, s_j)$  are called *concordant* if the ranks for both elements agree:

$$\begin{aligned} &\text{if } \text{Rank}(p_i) > \text{Rank}(p_j) \text{ and } \text{Rank}(s_i) > \text{Rank}(s_j), \\ &\text{or if } \text{Rank}(p_i) < \text{Rank}(p_j) \text{ and } \text{Rank}(s_i) < \text{Rank}(s_j). \end{aligned}$$

The pairs are called *discordant*,

$$\begin{aligned} &\text{if } \text{Rank}(p_i) > \text{Rank}(p_j) \text{ and } \text{Rank}(s_i) < \text{Rank}(s_j), \\ &\text{or if } \text{Rank}(p_i) < \text{Rank}(p_j) \text{ and } \text{Rank}(s_i) > \text{Rank}(s_j). \end{aligned}$$

If  $\text{Rank}(p_i) = \text{Rank}(p_j)$  and  $\text{Rank}(s_i) = \text{Rank}(s_j)$ , they are neither concordant, nor discordant.

The Kendall  $\tau$  rank correlation is calculated from formula:

$$\tau = \frac{N_c - N_d}{\frac{1}{2}r(r-1)}, \quad (5.10)$$

where  $N_c$  is the number of concordant pairs,  $N_d$  is the number of discordant pairs.

We can also compare the textured patterns by using the **Hamming distance**, that is the number of positions at which the corresponding elements of two sets are different. Therefore, the Hamming distance between two random variables  $P = p_1, \dots, p_r$  and  $S = s_1, \dots, s_r$  is defined:

$$\text{dist}(P, S) = \#\{i : p_i \neq s_i\}, \quad (5.11)$$

where notation  $\#E$  denotes the cardinality of the set  $E$ .

The use of introduced comparison metrics is presented in following section.

### 5.4.3 Clustering methods

In this section we introduce several pattern recognition methods based on clustering approach. First of them is based on maximization of correlation measure. Second is based on k-means clustering. The third one is based on textured pattern binarization and search of close textured patterns using Hamming distance. And the last one is based on characterization Kendall correlation values.

#### Pattern recognition method based on correlation measure

First of all we determine the pattern positions. We suppose that the size of the textured image after P&S ( $I_s$ ) is very close to the size of the original textured image  $I$ . Thus, the position of each pattern is calculated by using the first pattern position (i.e. top-left corner of  $I_s$ ) and pattern size  $r$ . The detection of the top-left corner is done by using a sliding window moved pixel by pixel. The position of the top-left corner is the first value that maximizes the correlation with  $C_l$  ( $l = 1, \dots, q$ ).

Then, the pattern recognition is performed. To detect the patterns, a sliding window is moved  $r$  pixels by  $r$  pixels. However, the size of  $I_s$  is slightly different from the size of  $I$ . Therefore, the position of the pattern is searched on a  $h \times h$  window around the nominal position. The position of the pattern is identified by the fact that it maximizes the correlation with one of representative pattern  $C_l$ . These correlation values (with  $C_l$ ) allow also the identification of the pattern.

The flowchart of this pattern recognition method is illustrated in Fig. 5.10. In this method, the correlation value  $CorVal$  could be calculated using one of presented correlation measures: Pearson correlation (5.1), Spearman correlation (5.9) or Kendall correlation (5.10).

#### K-means clustering

K-means clustering is a method of cluster analysis that aims to partition  $Q$  observations into  $q$  clusters, in which each observation belongs to the cluster with the nearest mean.

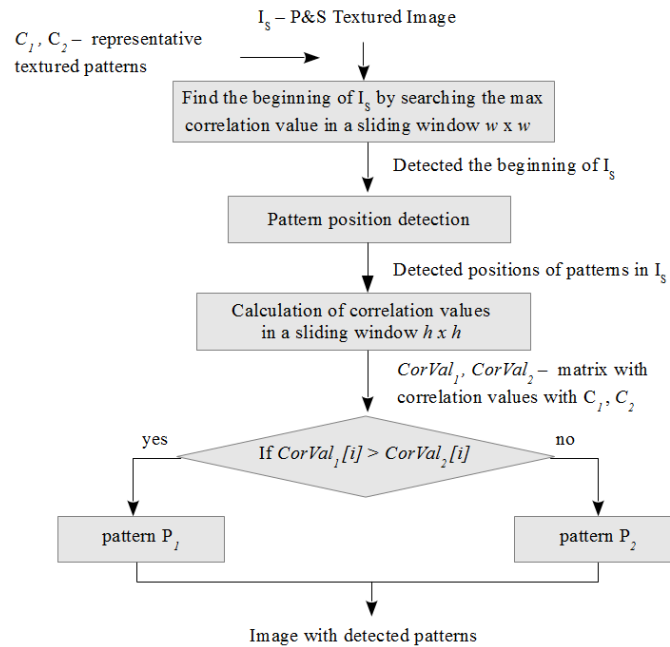


FIGURE 5.10: Textured pattern recognition using maximization of correlation measure.

Our experiment is described for particular case  $q = 2$ . As in previous detection methods, we start with definition of the pattern positions. When the pattern positions are determined, we calculate the maximal Pearson correlation values between each patch of  $I_s$  and representative candidates  $C_l$ ,  $l = 1, 2$  in the sliding window  $h \times h$ . As a result we obtain matrices of correlation values  $corVal_1$  and  $corVal_2$ , respectively. Further for each patch position in  $I_s$ , we calculate the differences between correlation values:

$$dif[i] = corVal_1[i] - corVal_2[i]. \quad (5.12)$$

After that, the k-means clustering is applied to the vector of differences  $dif$ . The classification results of k-means are used for pattern recognition. The flowchart of this recognition method is presented in Fig. 5.11.

*Remark:* Experimentally we determine that the differences between correlation values are more suitable for patterns classification using k-means clustering, than the correlation values.

### Binarization and Hamming distance

As we show in Section 5.2.1, the standard binarization methods are not effective for pattern recognition. That is why we suggest to combine the pattern detection using correlation measure with the pattern binarization using pattern density  $b$  and the use of Hamming distance for the pattern recognition.

This algorithm consists of two steps. The first step is the same as pattern detection using Pearson correlation measure. In the output of this step two classes are made: one



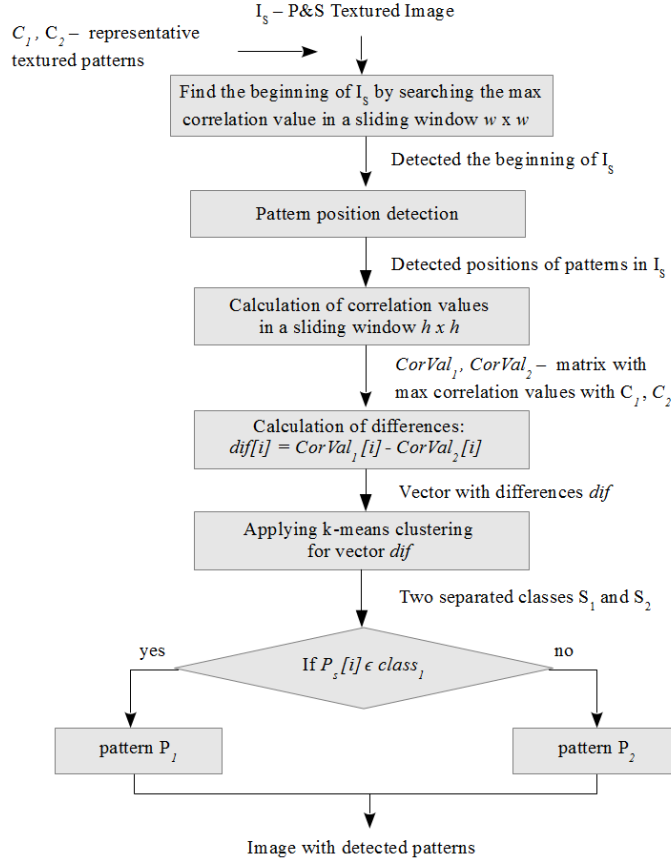


FIGURE 5.11: Textured pattern recognition using k-mean clustering.

consists of patterns detected as  $P_1$  ( $class_1$ ), other one consists of patterns detected as  $P_2$  ( $class_2$ ). The second step consists of binarization of all these patterns using the pattern density  $b$  (i.e. the number of black pixels in each pattern), we obtain the binarized patterns  $P_{bin}^i$ . This simple binarization method is described in Section 5.2.1. Then, the mean representative candidate  $C_l^{bin}$ ,  $l = 1, 2$  is constructed for each class ( $S_1$  and  $S_2$ ). After that the binarized patterns  $P_{bin}^i$  are reclassified by minimizing the Hamming distance between  $P_{bin}^i$  and  $C_l^{bin}$ . The Fig. 5.12 illustrates the flowchart of this algorithm. The main disadvantage of this algorithm is the execution time, that is higher than in previous presented methods, in addition due to binarization, this method can not refine the recognition results significantly. That is why in the rest of this manuscript we do not use this algorithm.

### Characterization Kendall correlation

We call this method characterization Kendall correlation due to pre-processing step, which can be named characterization step. This characterization step consists of calculation of probabilities  $p_1$  and  $p_2$  pixel appearances from representative set of  $T$  P&S patterns  $\mathcal{E}_l = \{E_l^1, \dots, E_l^T\}$ , where  $l = 1, \dots, q$ . Then, during the recognition step, the number of concordant and discordant pairs (see Kendall correlation (5.10)) is calculated using earlier calculated probabilities  $p_1$  and  $p_2$ . The flowchart of recognition algorithm

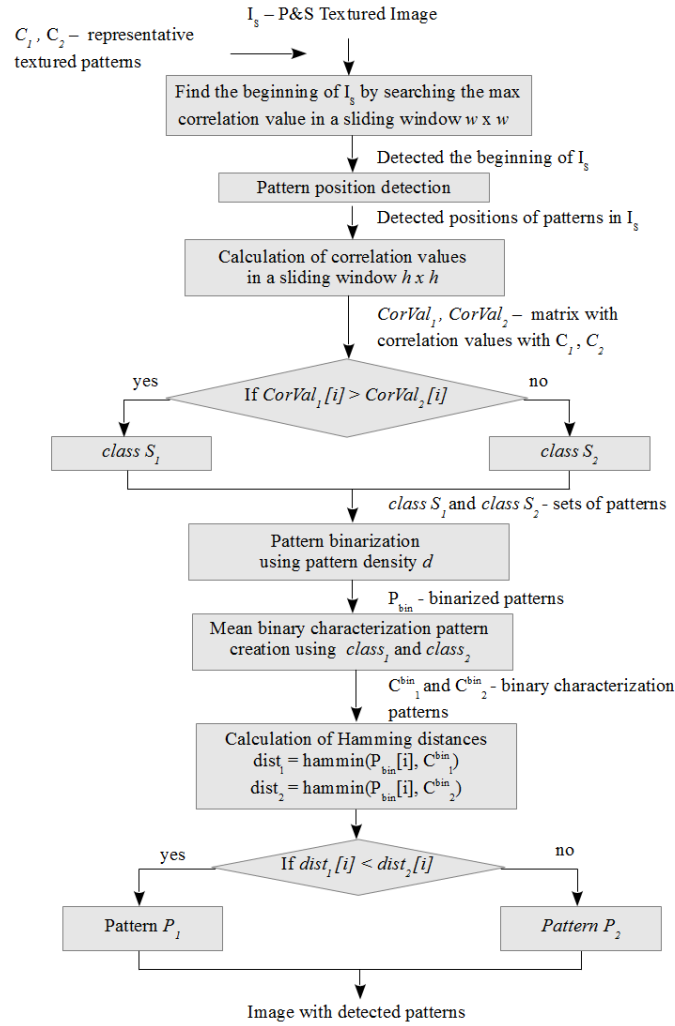


FIGURE 5.12: Textured pattern recognition using pattern binarization and Hamming distance.

is illustrated in Fig. 5.13.

We suppose that  $x_t^i$ , with  $t = 1, \dots, T$  and  $i = 1, \dots, r^2$ , is a vector of  $T$  grey level values of one pixel of representative set  $\mathcal{E}_l$ ,  $l = 1, \dots, q$ .

During the *characterization step* we calculate:

1. The probabilities that the pixel from a vector  $x_{t_1}^i$  is smaller than the pixel from a vector  $x_{t_2}^j$

$$p_1^{i,j} = p(x_{t_1}^i < x_{t_2}^j), t_1 = t_2 = 1, \dots, t, i = j = 1, \dots, r^2. \quad (5.13)$$

2. The probabilities that the pixel from a vector  $x_{t_1}^i$  is bigger than the pixel from a vector  $x_{t_2}^j$

$$p_2^{i,j} = p(x_{t_1}^i > x_{t_2}^j), t_1 = t_2 = 1, \dots, t, i = j = 1, \dots, r^2. \quad (5.14)$$

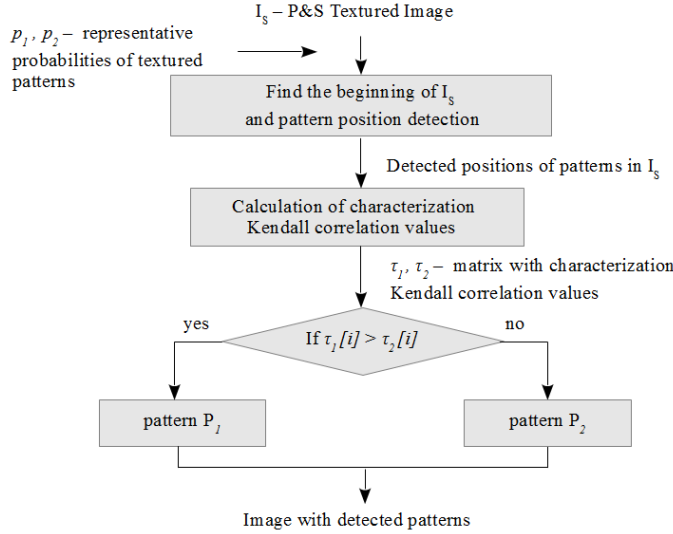


FIGURE 5.13: Textured pattern recognition using characterization Kendall correlation.

The characterization step could be done once, during creation of security element.

During the *recognition step*, for each patch  $P_s = y_1, \dots, y_{r^2}$  from  $I_s$  we calculate:

- the number of concordant pairs

$$N_c = \sum_i \sum_j p_1^{i,j} \times num(y_i < y_j) + \sum_i \sum_j p_2^{i,j} \times num(y_i > y_j), \quad (5.15)$$

- the number of discordant pairs

$$N_d = \sum_i \sum_j p_2^{i,j} \times num(y_i < y_j) + \sum_i \sum_j p_1^{i,j} \times num(y_i > y_j), \quad (5.16)$$

where function  $num(\varphi)$  corresponds to number of pixels that satisfy the condition  $\varphi$ .

Finally, we calculate the Kendall rank correlation coefficient using formula (5.10). We calculate the characterization Kendall correlation  $\tau_l$ ,  $l = 1, \dots, q$  with every representative set. In discussed case we have representative set  $\mathcal{E}_1$  for pattern  $P_1$  and representative set  $\mathcal{E}_2$  for pattern  $P_2$ , i.e  $q = 2$ . Therefore, the maximal characterization Kendall correlation value corresponds to pattern type:

$$\begin{aligned} &\text{if } \tau_1 > \tau_2, \text{ the pattern is recognized as } P_1, \\ &\text{otherwise it is recognized as } P_2. \end{aligned}$$

All these recognition methods have been tested during our experiments, and several experimental results are presented in following section.

## 5.5 Experimental results

For evaluation of suggested detection methods, we have used the security element with visual message (see Fig. 5.9.b). First of all, we chose two textured patterns from database of  $Q = 100$  textured patterns, that maximize criteria (5.2) and (5.3). The pattern size

was set to  $12 \times 12$  pixels. The number of black pixels in each pattern was set to  $b = 64$  (that corresponds to 45% of black pixels). The selected two textured patterns are shown in Fig. 5.14.



FIGURE 5.14: The pattern combination: a) Pattern 1 and b) Pattern 2.

We printed and scanned 200 times each textured pattern, in order to define the characterization patterns. We generated mean, median, maximum and minimum characterization patterns from these 200 samples. An example of characterization patterns for textured pattern  $P_2$  is illustrated in Fig. 5.15.

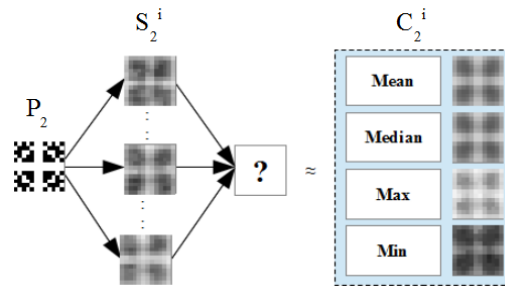


FIGURE 5.15: Examples of pattern  $P_2$  (Fig. 5.14.b) changes during P&S process.

We generated a textured image  $I$  of  $10 \times 21$  patterns. Examples of binary visual message and textured image with this message are presented in Fig. 5.16.a and Fig. 5.16.b. Then, we created, from the defined textured image, a printable at 600 dpi version, that corresponds to  $10.5 \times 5mm^2$ . We printed and scanned this image at 600 dpi using the printer Brother HL-4150CDN and the scanner Canon LIDE210. The resulting image  $I_s$  after P&S process is presented in Fig. 5.16.c - Fig. 5.16.d. Note that the image is quite blurred and the distribution of grey levels has been changed by the P&S process.



FIGURE 5.16: Example of textured images containing a visual message: a) Binary visual message, b) Textured image containing the visual message (a), c) P&S textured image (b), d) Zoom of central part of P&S textured image (c).

In our database, we have had 30 P&S samples of textured image Fig. 5.16.b. After experimenting with more than 10,000 samples, we conclude that the database of 30

samples can perfectly represent the random impact of P&S process. Therefore, we use these 30 samples to evaluate the effectiveness of suggested recognition methods.

The Table 5.2 presents the error detection results for all six suggested pattern recognition methods using different characterization patterns.

Pattern detection method	Type of characterization pattern				
	original	mean	median	max	min
Detection using correlation measure					
Pearson correlation	<b>0.63%</b>	29.37%	30.67%	37.19%	23.30%
Spearman correlation	<b>0.78%</b>	38.83%	43.89%	35.03%	31.03%
Kendall correlation	<b>0.78%</b>	40.08%	45.38%	35.76%	31.92%
Detection using alternative methods					
k-means clustering	<b>1.71%</b>	36.40%	37.81%	34.35%	31.16%
Pattern binarization and Hamming distance	<b>5.63%</b>	35.53%	29.01%	41.10%	36.67%
Characterization Kendall correlation	-	<b>2.53%</b>			

TABLE 5.2: Error probability detection of textured patterns in P&S textured image by using different characterization patterns for suggested detection methods.

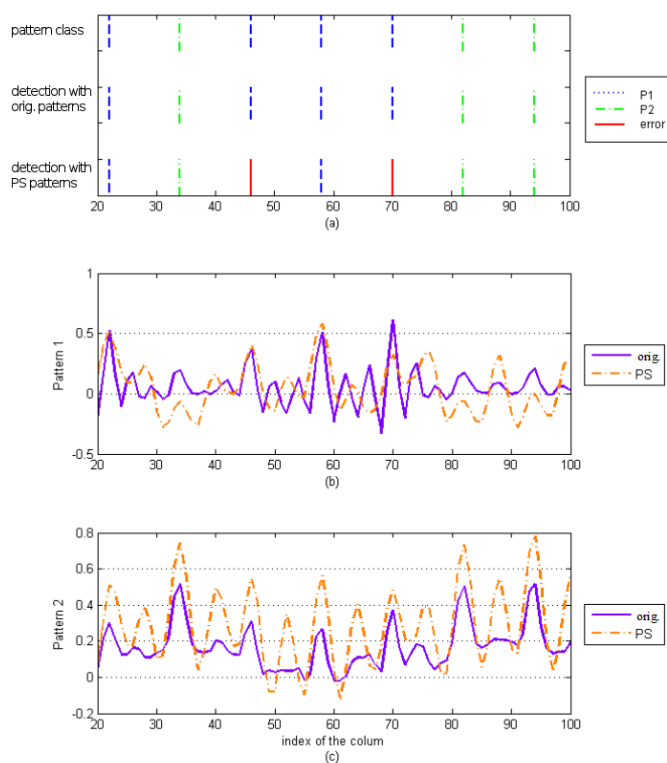


FIGURE 5.17: Correlation values of one line of textured image for pixels in interval  $[20 \cdots 100]$ .

In correlation based methods, the best pattern detection results were obtained with original patterns as representative candidates. Indeed, as shown in Fig. 5.17 on one of lines of the textured image, correlations with characterization patterns are generally higher than the correlations with the original patterns. Fig. 5.17.a shows seven textured

patterns in sub-sample line (the blue lines correspond to the pattern  $P_1$ , the green lines correspond to the pattern  $P_2$ , the red lines to errors in detection). Fig. 5.17.b (respectively Fig. 5.17.c) shows 1) the correlation values among the P&S patterns and the original pattern  $P_1$  (respectively  $P_2$ ) using purple line, 2) the correlation values among the P&S patterns and the median pattern  $P_1$  (respectively  $P_2$ ) using orange line. We show the correlation values with median patterns as they gave the best detection rate among all representative patterns. In Fig. 5.17, the detection peaks correspond to pattern locations. While the correlation value is higher with median pattern than with the original pattern, it induces more errors in pattern detection. An explanation that we could give to this phenomenon is that both patterns  $P_1$  and  $P_2$  were selected to optimize the correlation among scanned patterns and original patterns and not to maximize the correlation among scanned patterns and characterization patterns.

## 5.6 Conclusions

In this chapter, we present a security element for identifying the legitimacy of documents after P&S process. This security element is constructed by using specific textured patterns, which are sensitive to P&S process. We experimentally determine three types of security elements: with distinguishable patterns, semi-distinguishable patterns and indistinguishable patterns. The security element with semi-distinguishable patterns is chosen for identification of legitimate documents after P&S process.

Different pattern detection methods are suggested to recognize the embedded message. The methods, that maximize the correlation value between original patterns and P&S patches of security element, perform the best pattern recognition rates.

The contributions of this chapter were published in international conference [143].



## Chapter 6

# High density barcodes

### 6.1 Introduction

The high density barcodes are developing field as was mentioned in Chapter 4. The amount of information stored increases, but the barcode size decreases. The storage capacity/code size trade off is the current main problem. Several standard barcodes were developed to store a big amount of information, for example the QR code. However, the standard dense versions of QR code have several reading problems due to frequent changes and small sizes of black and white modules. The barcode binarization and tilt correction are the important reading problems for flatbed scanner and smartphone QR code applications, despite the presence of error correction codes.

The pattern detection described in Chapter 5 shows us that central pixels statistically are less changed. This observation gives us an idea to improve the module binarization results: suggested measure, based on centrality bias of each module, is called Weighted Mean Square Error (WMSE). This WMSE measure is used for a new classification method and for improvement of standard binarization methods.

In our approach, the high density QR codes are used in multimedia management applications, such as automatic tracking and tracing of printed documents (tax forms, invoices). We consider an application scenario: the administrative information is stored in the high density QR code, which is inserted into the document. This document is then printed using an office printer. During the automatic tracking and tracing, this document is scanned using an office scanner by the administrative office. The information stored in QR code is then automatically read by application.

In the following sections, we present the QR code reading process and several binarization methods (in Section 6.2), the proposed centrality bias measure (in Section 6.3) and binarization methods using this measure (in Section 6.4). Then the numerous experiments are illustrated to highlight the effectiveness of this measure in Section 6.5. Finally, we conclude this chapter in Section 6.6.



## 6.2 High density QR code

The high density QR codes are the QR codes of version 35 – 40. These versions have size from  $157 \times 157$  to  $177 \times 177$  modules and can store between 18,448 and 23,648 data bits in low error correction level.



FIGURE 6.1: QR code evolution: a) QR code V10 with  $57 \times 57$  modules, b) QR code V40 with  $177 \times 177$  modules.

If we compare the QR code V10 and QR code V40 in Fig. 6.1, we can note the high increasing of module density in QR code V40. The module density affects a lot the reading process of QR code.

### 6.2.1 Reading process

As it was introduced in Section 4.2.2, the QR code recognition algorithm consists of three steps: 1) pre-processing; 2) binarization; and 3) decoding algorithm. We should note that in [4], pre-processing is done after binarization. The main steps are presented in Fig. 6.2. A grayscale image with rotated QR code is an input for reading application. First, the position pattern localization is applied. Then we apply a re-sampling process in order to set the exact QR code size and correct its orientation. After this, the recognition of white and black modules should be completed, before applying the standard decoding algorithm. The output is the extracted message.

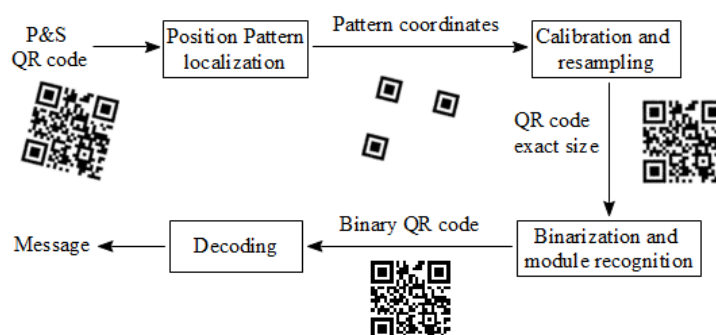


FIGURE 6.2: The main steps of QR code recognition algorithm.

*Pre-processing.* During this step we perform the image geometric correction. First of all,

we localize the position patterns. The standard process [4] is based on a search of the areas with 1:1:3:1:1 dark-light ratio. The authors in [137] use the Canny edge detector and look for square contours for position pattern localization. Thanks to the regular structure of the QR code, the Hough transform for QR code detection and localization can be used [90]. The Hough transform is used to detect the four corners of the 2D barcode region in [157]. In [100], the authors combine the Hough transform with adaptive threshold and texture analysis to locate 2D color barcodes correctly. The authors in [139] also use the Hough transform and the property of 2D barcodes with a regular distribution of edge gradients to localize the QR codes in high-resolution images.

After localization the QR code is resampled using the determined position pattern coordinates (for example, by applying a nearest or bilinear interpolation method). The authors in [48] suggest to use bilinear interpolation method because of the quality and clarity of the image results. In [30], a geometry-based corner searching algorithm and boundary modeling using the parabola equation are suggested for correction of 3D distortion. The result of this operation creates a QR code of an exact size with the correct orientation.

*Binarization.* In this step we pass from a grayscale image to a black-and-white image. This process is discussed profoundly in the following section.

*Decoding algorithm.* During the last step of the QR code recognition algorithm, the standard QR code decoder [4] is applied to the re-sampled and binarized QR code image.

### 6.2.2 Binarization methods

Various binarization methods have been proposed for document binarization [125]. Some of these methods have been adapted for QR code binarization like the Otsu method, Niblack method or Sauvola document binarization method. We can categorize the existing QR code binarization methods into three large groups: the global thresholding methods, local thresholding methods and the mixed (global and local) thresholding methods.

The standard QR code binarization algorithm is based on *global thresholding*. The global image threshold is calculated as the middle value between the maximum and minimum reflectance in the image [4]. This method is very simple but it is inefficient under non-uniform lighting conditions [84]. An example of QR code binarization under non-uniform lighting conditions is shown in Fig. 6.3. Another popular global thresholding method is the Otsu's algorithm [98]. The threshold for this algorithm is chosen by minimizing within-group variance. Both binarization methods cannot perform effective binarization results for images that were taken in variable lighting conditions and for low resolution scanned QR symbols.



FIGURE 6.3: An example of: a) QR code under non-uniform lighting conditions, b) binarization results using standard global thresholding method.

In the case of variable lighting conditions, *local thresholding methods* perform better. The experimental results in [145] show that Niblack's method using post-processing appears to be the most suitable. Correct window size, elimination of block effects and reduction of the execution time are the main local thresholding method problems. The authors in [99] use a modified version of Niblack algorithm where a threshold for a given point is set as the average gray-level of point neighborhood.

The following two methods are *mixed binarization methods*, because we calculate the global threshold by using local thresholds. In [97], the authors propose to calculate global threshold using nine  $60 \times 60$  pixel parts that are selected nearby the center of the QR code image. These parts are considered as sampling points to define the threshold between the black and white luminance levels. The local threshold of each part is calculated as the median luminance value. Then, the global threshold is defined as the minimum value among all calculated local thresholds.

The authors in [84] suggest an adaptive multilevel thresholding method, which integrates local with global threshold. They propose to filter the gray-level image histogram and analyze the histogram peak features. For a bimodal distribution of filtered histogram, the global threshold is equal to the lowest trough or the middle value of the flat trough. For a single peak histogram, they adopt an iterative thresholding method, which calculates threshold as a mean value among old global threshold and the center of the dark or light areas. For a multipeak distribution histogram, the local threshold algorithm is used.

The improved background gray-level based binarization algorithm is introduced in [168], this calculates the gray-level value for each sub-block and, then, uses the joint interpolation algorithm to build the gray-level image, which is finally binarized by Otsu's algorithm. This improved binarization algorithm can effectively correct the QR codes under nonuniform lighting conditions.

The effective Sauvola's adaptive document binarization method [123] is modified to solve the QR code image binarization problem. The authors in [169, 71] used the Sauvola's threshold formula for local binarization process of the QR code. Then, the previous binarization results are improved in [82] by slightly changing the threshold formula.

All presented binarization methods use the image histogram for global or local threshold definition. The global threshold binarization methods are not very effective for image illumination changes and for frequent changes in pixel gray-levels such as in high density

QR codes.

In document binarization methods there are also multi-pass algorithms, these are named locally adaptive thresholding methods [125]. In [96], the authors set lower and higher threshold values, and then the pixels, that are situated between these thresholds, are binarized by using indicator kriging. The authors in [65] use the global threshold during the first stage and a local refinement process during the second stage.

The other type of document binarization methods are based on a machine learning approach. The authors in [29] use the support vector machine approach for binarization of document images produced by camera. Nevertheless, this method needs a training process, which is produced by human labeling of a training set.

The performance of these methods is verified using the small QR code versions V1-V15. Thus, we can not compare these methods with the one proposed in this chapter, as our methods are used for reading high density QR code versions.

We suggest a new two-pass QR code binarization method, which uses the second stage for binarization refinement. At the same time, the proposed method is based on an unsupervised machine learning approach, as we use the results, obtained during first stage, for second stage binarization.

### 6.3 Proposed centrality bias measure

We focus on high density QR code reading amelioration by binarization improvements. We propose the Weighted Mean Square Error (*WMSE*) measure, which takes into account the fact that the module borders influence each other and change the pixel values due to P&S distortion. This means that the significance of central pixels increases and the significance of border pixels for each module decreases. We suggest the use of *WMSE* measure for module classification to increase the QR code recognition rate. In addition, the *WMSE* measure can be used for improving the ISO binarization method and Otsu's method.

As it was presented experimentally, the gray-levels of neighboring pixels influence the scanned gray-level pixel values [6]. That is why the suggested measure is based on the fact that the central pixels of modules are more informative than pixels placed on the module border. Indeed, the central pixels are better to represent the module. Moreover, the border of one module can influence a border of another module, so the border pixel values do not represent the module correctly. For example, a black module surrounded by white modules, as illustrated in Fig. 6.4.a, after the P&S process tends to make lighter pixels of the black module, as illustrated in Fig. 6.4.b. And a white module surrounded by black modules, as illustrated in Fig. 6.4.c, after the P&S process tends to make darker pixels of the white module, as illustrated in Fig. 6.4.d.

Therefore we propose to increase the significance of the central pixels of each module and to decrease the significance of pixels on the borders by using the *WMSE* measure.

Consider the module  $M$  as a square block of  $(r + 1)^2$  pixels, where  $r$  is an even positive integer. The structure of module  $M$  is illustrated in Fig. 6.5.



FIGURE 6.4: Examples of a) Black module surrounded by white modules, b) Changes in a black module surrounded by white modules after the P&S process, c) White module surrounded by black modules, d) White module surrounded by black modules after the P&S process.

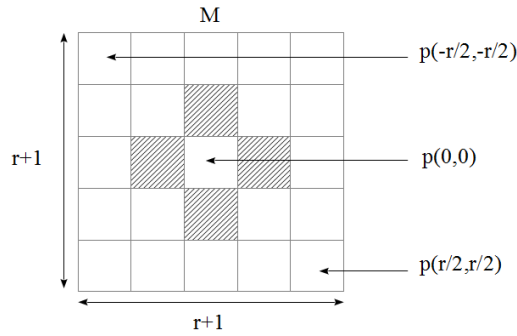


FIGURE 6.5: Module structure composed of  $(r + 1)^2$  pixels. The hatch pixels represent 4-connected neighbors.

In the local reference system, the upper left pixel has position  $p(-\frac{r}{2}, -\frac{r}{2})$ , the lower right pixel has position  $p(\frac{r}{2}, \frac{r}{2})$  and the central pixel is  $p(0, 0)$ . The  $WMSE$  between an original module  $M$  (black module  $M_B$  or white module  $M_W$ ) and a P&S module  $M'$  (with pixel  $p'(i, j)$ ) is:

$$WMSE(M, M') = \frac{1}{T} \sum_{i=-\frac{r}{2}}^{\frac{r}{2}} \sum_{j=-\frac{r}{2}}^{\frac{r}{2}} w(i, j) D(p(i, j), p'(i, j)), \quad (6.1)$$

where

$$D(p(i, j), p'(i, j)) = (p(i, j) - p'(i, j))^2, \quad (6.2)$$

and  $w(i, j)$  is a function which gives more weight to the central pixels,  $T$  is the total number of weights used in these calculations.

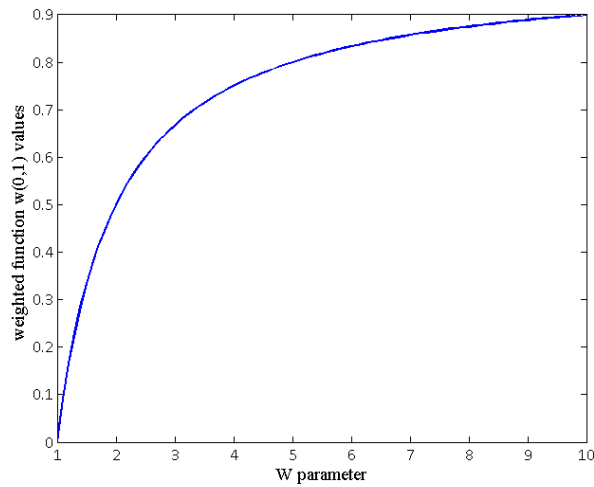


FIGURE 6.6: The values of weight function  $f$  depending on weight parameter  $W$  for 4-connected pixels ( $|i| + |j| = 1$ ).

We propose to use:

$$w(i, j) = \begin{cases} 1 - \frac{|i|+|j|}{W}, & \text{if } 1 - \frac{|i|+|j|}{W} \geq 0 \\ 0, & \text{otherwise} \end{cases}, \quad (6.3)$$

where  $W$  is the weight, which is given to the central value of the module, this weight is a real number and respects the condition  $W \geq 1$ . Consequently, for  $T$  we have:

$$T = W(r+1)^2 + \frac{r}{2}(r+1)(r+2). \quad (6.4)$$

We precise several noticeable values for weight function  $w(i, j)$ :

– the central pixel  $p(0, 0)$  has the same weight function value:

$$\forall W : w(0, 0) = 1. \quad (6.5)$$

– the weight function for 4-connected pixels is equal to zero, when the weight parameter  $W = 1$ :

$$W = 1 : w(i, j) = 0 \quad \forall (i, j) \in \{(-1, 0), (1, 0), (0, -1), (0, 1)\}. \quad (6.6)$$

In Fig. 6.6, we plot the dependence of weight function  $w(i, j)$  from parameter  $W$  for 4-connected pixels, for example for pixel  $p(0, 1)$ . We state that the weight given to each 4-connected pixel is approximately equal to 1 with  $W = 10$ , i.e. when increasing the value of  $W$  the importance of the central pixel decreases.

*Remark:* The weight function  $w(i, j)$  may be defined differently. For example, in our first experiments, we have used the weight function  $w(i, j) = W - |i| - |j|$ , with  $W \geq r$ . This weight function produces also acceptable recognition results.

## 6.4 Binarization using WMSE measure

This section aims to present our proposed module binarization methods using the *WMSE* measure. In following sections we present a new module binarization method using classification and improvements of global binarization methods: ISO standard method and Otsu's method.

### 6.4.1 Classification binarization method

In this section we propose a recognition method based on the *WMSE* measure and module classification. This method aims to increase the QR code recognition rate. For the first step we classify the modules into two classes ( $C_W$  - white module class,  $C_B$  - black module class). For the second step we calculate characterized modules for black and white classes ( $M_{C_B}$  - black characterized module,  $M_{C_W}$  - white characterized module). Then, in the second step, we compare each module with the characterized modules ( $M_{C_B}$  and  $M_{C_W}$ ). Fig. 6.7 presents the scheme of the proposed method. During the first step we classify our modules into two classes by minimizing the *WMSE* measure between original modules and P&S modules. During the second step the classification is made by minimizing the *WMSE* measure between characterized modules and P&S modules. In Fig. 6.7, the  $M_{C_W}, M_{C_B}$  are the characterized modules of the classes  $C_W$  and  $C_B$ , respectively.

**First step - Module classification.** We classify all modules into  $C_W$  and  $C_B$  classes. For this we calculate *WMSE* measure among modules in the P&S QR code, and black ( $M_B$ ) and white ( $M_W$ ) modules using equation (6.1).

The  $WMSE_B$  (rsp.  $WMSE_W$ ) is the *WMSE* value calculated among black module  $M_B$  (rsp. white module  $M_W$ ) and module  $M'$  (P&S module). We create two classes ( $C_W$  - white module class,  $C_B$  - black module class) by choosing the minimum value between  $WMSE_B$  and  $WMSE_W$ . Module  $M'$  belongs to black class  $C_B$  if the  $WMSE_B$  value is smaller than the  $WMSE_W$  value, otherwise  $M'$  belongs to white class  $C_W$ :

$$M' \in \begin{cases} C_B, & \text{if } WMSE_B < WMSE_W, \\ C_W, & \text{otherwise.} \end{cases} \quad (6.7)$$

The output values of this step are composed of two sets of modules,  $C_B$  (black module class) and  $C_W$  (white module class), and the first binary QR code, which is a QR code candidate for decoding. We could stop the recognition method after the first step.

**Second step - Recognition based on module characterization.** As illustrated in Fig. 6.7, we can improve the recognition results obtained during the first step by creating a characterization module  $M_{C_B}$  (rsp.  $M_{C_W}$ ) for black class  $C_B$  (rsp. for white class  $C_W$ ) obtained during the first step. The characterization modules  $M_{C_B}$  and  $M_{C_W}$

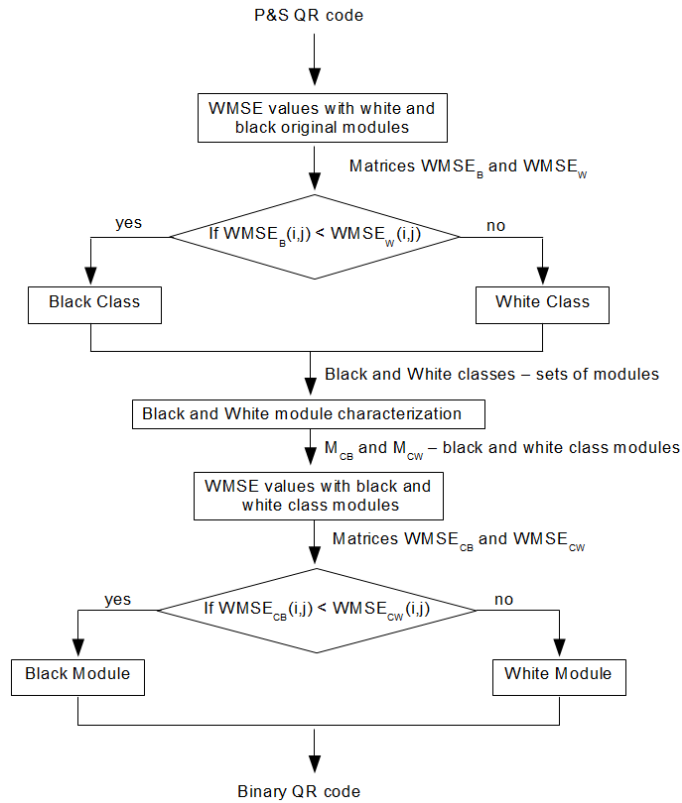


FIGURE 6.7: An overview of the proposed recognition method based on two steps.

represent the black and white classes, respectively. These characterized modules will thus be used for QR code module recognition. The image changes randomly after the P&S process, so we need to have a set of  $t$  P&S samples  $\mathcal{E} = \{E^1, \dots, E^t\}$  for each module (black or white module after the P&S process). We conduct module characterization using statistical measurements:

- *Mean image*: make the image with a mean value of each pixel of images in  $\mathcal{E}$  as it was defined in equation (5.4);
- *Median image*: make the image with a median value of each pixel of images in  $\mathcal{E}$  as it was defined in equation (5.5).

The result of the module characterization is two characterized modules  $M_{CB}$  and  $M_{CW}$ , which are the mean (or median) images of  $C_B$  and  $C_W$  classes.

Then, as in the first step, we calculate the  $WMSE$  values using equation (6.1), but we replace the black and white reference modules  $M_B$  and  $M_W$  by the characterization modules  $M_{CB}$  and  $M_{CW}$ . Therefore, the  $WMSE_{CB}$  (rsp.  $WMSE_{CW}$ ) is the  $WMSE$  value calculated between the black characterization module  $M_{CB}$  (rsp. white characterization module  $M_{CW}$ ) and P&S QR code module  $M'$ . Then, analogically to the first step, we choose the minimum value between  $WMSE_{CB}$  and  $WMSE_{CW}$  to recognize the QR code modules:

$$M' \in \begin{cases} 0, & \text{if } WMSE_{CB} < WMSE_{CW}, \\ 1, & \text{otherwise.} \end{cases} \quad (6.8)$$



The output from this step is the binary QR code with a higher recognition rate in comparison to the binary QR code after the first step classification.

### 6.4.2 Global thresholding methods vs. WMSE classification

The global thresholding methods perform the QR code binarization in two steps:

- Image binarization. This step can be viewed as pixel classification into black and white classes (identification of a global threshold).
- Module color decision. This one determines the module color by using majority vote for each module. The number of white and black pixels in each module are calculated. The biggest number corresponds to the color (white or black) of the module. This method does not ensure high recognition rates for high density QR codes.

The **standard ISO binarization method** [4] calculates the global threshold as the middle value between the maximum and minimum reflectance of the image.

The **Otsu's threshold method** [98] chooses the threshold that minimizes within group variance.

After binarization using both thresholding methods, the module color decision using the majority vote has to be done.

To illustrate the advantages of the proposed WMSE classification method, we select a white module  $M'$  surrounded by black modules and provide the module binarization using our method and both global thresholding binarization methods. This example is shown in Fig. 6.8. The calculated global image threshold for the ISO method is equal to 137 for this example, Fig. 6.8.b. If we apply binarization by the standard ISO method with a threshold of 137, we obtain 14 black pixels and 11 white pixels. After the majority vote it corresponds to a black module. Analogically, we calculate the image threshold using Otsu's method (Fig. 6.8.d), which is equal to 134 for this example. By applying the Otsu's binarization method, we obtain 13 black pixels and 12 white pixels. After the majority vote, this module is recognized as a black module.

However, the proposed classification method recognizes this module as a white module, Fig. 6.8.a. We fix the weight parameter  $W = 3$  and we calculate the  $WMSE$  value with black  $WMSE(M_B, M')$  (rsp. with white  $WMSE(M_W, M')$ ) module. We get  $WMSE(M_B, M') = 1,292$  and  $WMSE(M_W, M') = 469$ . Because the minimal value corresponds to  $WMSE(M_W, M')$ , the module is recognized as a white module.

### 6.4.3 Weighted global binarization methods

In order to improve the recognition rate of the global binarization methods, we propose to replace the majority vote by the  $WMSE$  measure. The module color decision, in this case, is made by using the  $WMSE$  measure. The  $WMSE$  measure is calculated between the binarized modules and referenced black and white modules ( $M_B$  and  $M_W$ ).

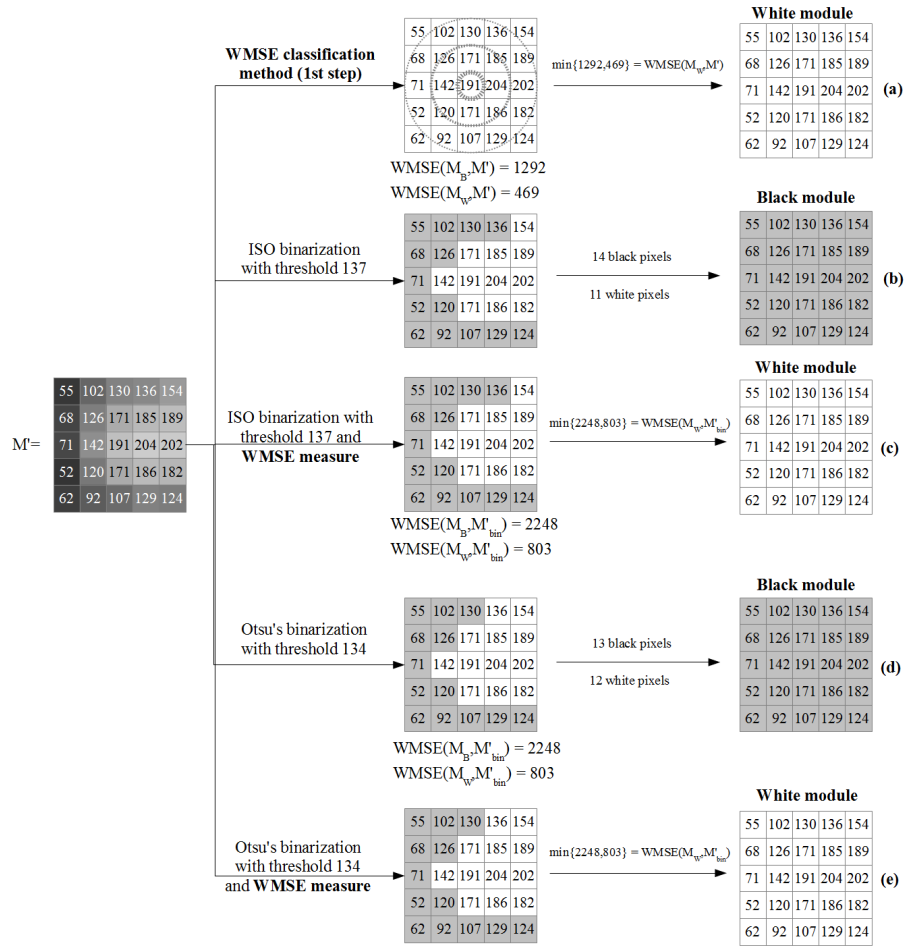


FIGURE 6.8: Comparison of white module recognition using ISO standard method with majority vote and with WMSE measure, using Otsu's threshold method with majority vote and with WMSE measure and using the proposed classification method.

We use the same white module  $M'$  surrounded by black modules to illustrate the advantage of the  $WMSE$  measure in comparison with majority vote (results of Fig. 6.8.c and Fig. 6.8.e vs. results of Fig. 6.8.b and Fig. 6.8.d).

The referenced module  $M'$  is recognized as white, when we use the proposed  $WMSE$  measure instead of using the majority vote in ISO standard binarization (in Fig. 6.8.c referenced as ISO  $WMSE$ ). Let  $M'_{bin}$  be the binarized ISO method module. We calculate the  $WMSE$  values between the black module  $M_B$  and binarized module  $M'_{bin}$  ( $WMSE(M_B, M'_{bin})$ ) and between the white module  $M_W$  and binarized module  $M'_{bin}$  ( $WMSE(M_W, M'_{bin})$ ). With  $W = 3$ , we get  $WMSE(M_B, M'_{bin}) = 2,248$  and  $WMSE(M_W, M'_{bin}) = 803$ . Since the minimal  $WMSE$  value corresponds to  $WMSE(M_W, M'_{bin})$ , the module is recognized as a white module.

Analogically, the referenced module  $M'$  is recognized as a white module when we use the  $WMSE$  measure in Otsu's binarization method (referenced as Otsu's  $WMSE$  in Fig. 6.8.e). Let  $M'_{bin}$  be binarized by Otsu's method module. The  $WMSE$  value between the black module  $M_B$  and binarized module  $M'_{bin}$  (with weight parameter  $W = 3$ )

is equal to  $WMSE(M_B, M'_{bin}) = 2,248$  and the  $WMSE$  value between the white module  $M_W$  and binarized module  $M'_{bin}$  is equal to  $WMSE(M_W, M'_{bin}) = 803$ . The module is recognized as a white module because the minimal  $WMSE$  value corresponds to  $WMSE(M_W, M'_{bin})$ .

In this example both of global binarization methods perform better when use the  $WMSE$  measure. In Section 6.5.2 more exhaustive results and comparisons between all methods are presented.

## 6.5 Experimental results

In this section we aim to present several experimental results. The described experiments have been made using a big database of P&S QR codes V40 that contains more than 10,000 images. We describe this database in Section 6.5.1. The analysis and comparison of proposed and referenced binarization methods is provided in Section 6.5.2. The additional experiments are done in order to optimize the weight parameter  $W$  in Section 6.5.3. Finally, several supplementary experiments to simulate the camera capture of QR codes are performed in Section 6.5.4.

### 6.5.1 Database description

For our experiments a QR code version 40 was used as illustrated in Fig. 6.9.a. The QR code version 40 is currently the largest available, due to its high density the reading process is often failed.

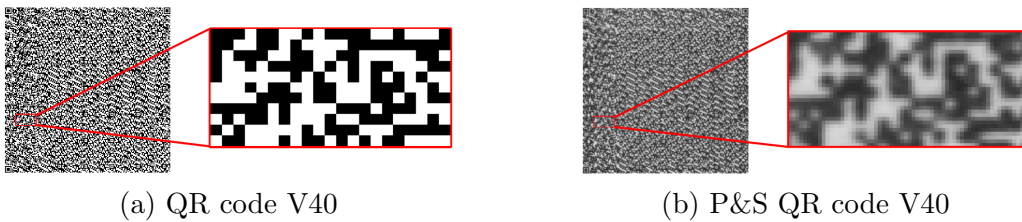


FIGURE 6.9: a) Original QR code version 40 and a zoom of a selected part, b) The same QR code and a zoom after the P&S process.

We set the module size at  $3 \times 3$  pixels which defines the tiny modules. That is why the QR code image size is  $531 \times 531$  pixels, this corresponds to  $22 \times 22 \text{ mm}^2$  at 600 dpi resolution. The P&S version of this QR code is illustrated in Fig. 6.9.b. Note that the P&S QR code is blurred and non-binary, as illustrated in the zoom in Fig. 6.9.b.

For our experiments 3 printers have been used: a HP LaserJet Pro CM1415 printer-scanner, a HP LaserJet P4015 printer and a Brother HL-4150CDN printer. We have also used 3 scanners: a HP LaserJet Pro CM1415 printer-scanner, a Canon LIDE210 scanner and a Toshiba e-studio 455 scanner. Therefore, we had 9 printer-scanner pairs. As

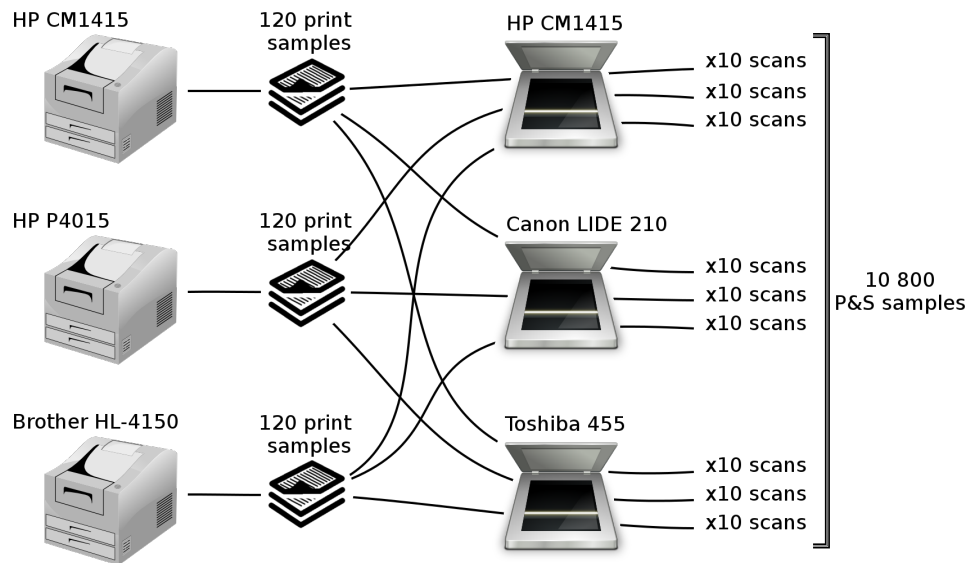


FIGURE 6.10: Schematic description of our image database.

additional information, the specific characteristics of printers and scanners are presented in Table 6.1. Both printers and scanners have been used in 600 dpi resolution. Fig. 6.10 describes schematically our image database. We printed the same QR code 120 times using each printer, then each printed QR code was scanned 30 times (10 times using each scanner). That is why for each pair printer-scanner we had 1,200 samples. In total, we had 10,800 samples of P&S QR codes.

Printer characteristics		
Printer name	Maximum grayscale resolution from manufacturer	Color mode
HP CM1415 (P1)	600 dpi	yes
HP P4015 (P2)	1200 dpi	no
Brother HL4150 (P3)	600 dpi (quality 2400 dpi)	yes
Scanner characteristics		
Scanner name	Maximum grayscale resolution from manufacturer	Photo mode
HP CM1415 (S1)	1200 dpi	no
Canon LIDE210 (S2)	1200 dpi	no
Toshiba 455 (S3)	600 dpi	yes

TABLE 6.1: Characteristics of the printers and scanners.

### 6.5.2 Detection results

After the P&S process we corrected the orientation and geometrical distortions of the QR code samples. The bilinear and nearest interpolations have been used. The recognition

results after bilinear interpolation were better than after nearest interpolation. That is why in our tests we applied the bilinear interpolation for all our database samples. After the pre-processing steps we concluded that our QR codes were correctly rotated and had the right orientation.

We have decided to analyze the changes in the recognition rate for QR code with module size  $3 \times 3$  because of using different printers and scanners. Since the ISO standard method and the Otsu's method are used in majority of existing QR code readers, we compare the proposed recognition methods only with these two methods.

First, the ISO standard method and Otsu's binarization method were applied to all database samples. After these two methods the majority vote was applied for each  $3 \times 3$  module in order to recognize the color value of the module (white or black). At the end, the mean recognition rate for ISO standard method was equal to 89.50%. The mean module recognition rate for Otsu's method was 90.78%.

An example of the module recognition using ISO standard method is illustrated in Fig. 6.11.a-Fig. 6.11.b. In the zoomed part of the QR code illustrated in Fig. 6.9.a, we can see some isolated white modules surrounded by black modules. We show the binarization results with ISO standard method in Fig. 6.11.b. In Fig. 6.11.b1 the results of the majority vote are illustrated. At the end, the comparison of module recognition image with original QR code is illustrated in Fig. 6.11.b2. In this figure, there are 33 errors (red squares) in a  $21 \times 11$  part of the QR code.

The analogical results can be obtained for Otsu's method. The number of errors in the same  $21 \times 11$  part of the QR code is equal to 28.

Then, we applied the proposed classification method with the weight parameter  $W = 2$ . The recognition rate after the first classification step was equal to 93.41%, after the second step, we had increased the recognition rate to 93.93%. These results improved the average overall recognition rate by up to 4.43% in comparison with the ISO standard method.

The last experiments have been made by using the ISO *WMSE* method and Otsu's *WMSE* method (the weight parameter  $W = 2$ ). The recognition rate for the ISO *WMSE* method was equal to 92.28%, which improves the recognition rate based on the ISO standard method by up to 2.77%. The recognition rate for Otsu's *WMSE* method was equal to 93.39%, which had improved the recognition rate of Otsu's method by up to 2.61%.

An example of the module recognition using the ISO *WMSE* method is illustrated in Fig. 6.11.c. We took the same zoomed part of P&S QR code illustrated in Fig. 6.9.b. The results of the binarization with the ISO *WMSE* method are illustrated in Fig. 6.11.c1. The comparison of module recognition of the zoomed part of P&S QR code with the zoomed part of the original QR code is illustrated in Fig. 6.11.c2. In this figure there are 20 errors (red squares) in a  $21 \times 11$  part of the QR code, instead of 33 errors as presented in Fig. 6.11.b2.

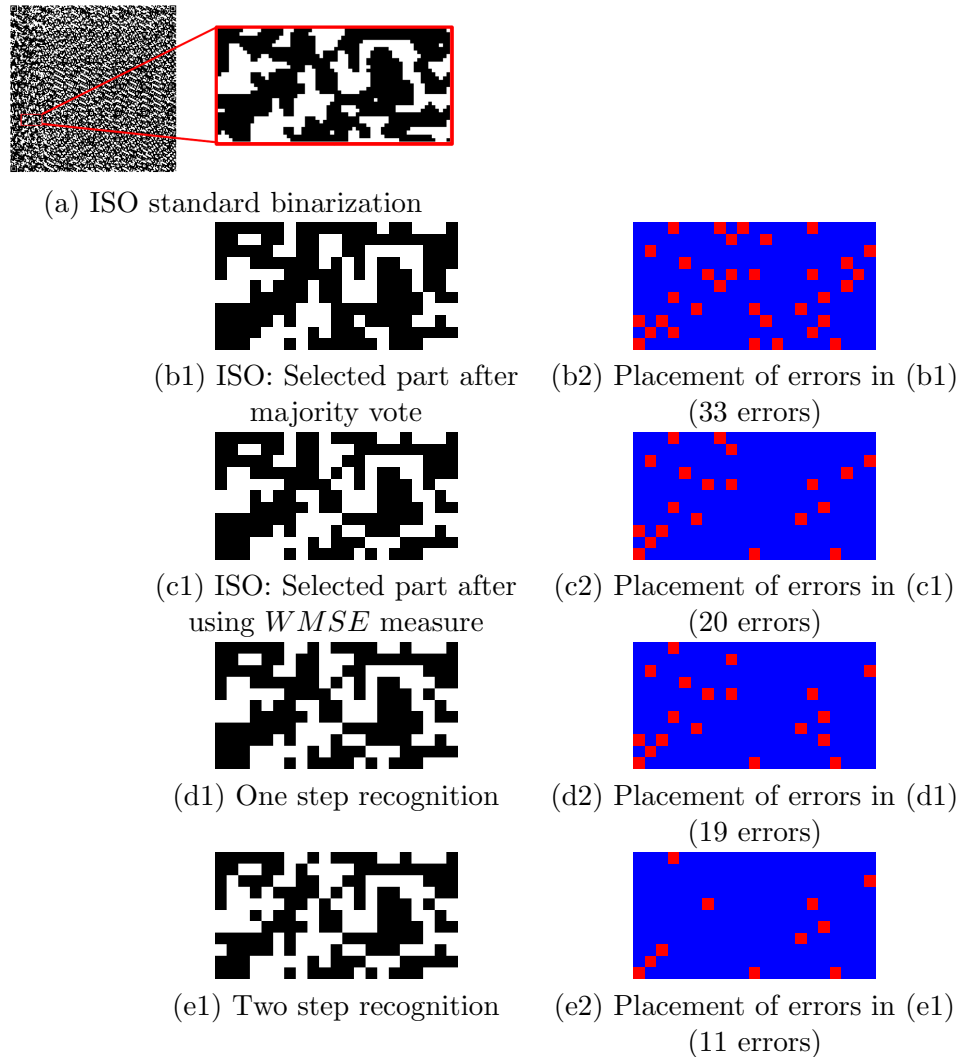


FIGURE 6.11: Module recognition results with ISO standard method using majority vote (b1-b2) and the *WMSE* measure (c1-c2), one step classification method (d1-d2) and two step classification method (e1-e2) for binarization of the QR code of Fig. 6.9.b.

The analogical results can be obtained for Otsu's *WMSE* method. The number of errors in the same  $21 \times 11$  part of the QR code is equal to 11 (that is the best score for this zoomed part). These results show that the *WMSE* measure improves significantly the recognition rate of high density QR codes.

In Fig. 6.11.d and Fig. 6.11.e, an example of the module recognition using proposed classification method is presented. Fig. 6.11.d1 and Fig. 6.11.e1 illustrate the same zoomed part of QR code from Fig. 6.9.b after the first step classification and second step classification, respectively. We note that there are better recognition results for white isolated modules by using second step classification, Fig. 6.11.e. The number of errors (red squares in Fig. 6.11.d2) after the first step classification is equal to 19 in comparison to the original QR code. And the number of errors (red squares in Fig. 6.11.e2) after the second step classification is equal to 11 in comparison to the original QR code.

The recognition rates for each printer-scanner pair are presented in Table 6.2. The column *ISO* (rsp. *Otsu's*) corresponds to the recognition results obtained by the ISO standard method (rsp. by Otsu's method). The column *ISO WMSE* (rsp. *Otsu's WMSE*) corresponds to the recognition results obtained by using the ISO standard method (rsp. Otsu's method) improved by using the *WMSE* measure. Finally, the column *WMSE* corresponds to the recognition results obtained by first step of proposed classification method, the column *WMSE mean* (rsp. *median*) corresponds to recognition results obtained by second step classification method with *mean* (rsp. *median*) characterization modules. We can observe that the recognition rate changes due to the printer and scanner used. The HP CM1415 printer-scanner pair gives the worst recognition results. The best recognition rate is obtained by using the Brother HL4150 printer and the Toshiba 455 scanner. These results can be explained by the quality of the printer (according to the data-sheet of the Brother HL4150, the printer is able to work with 2400 dpi resolution) and by the quality of the scanner (the Toshiba 455 scan images in Photo Mode). The results of this table show that the use of *WMSE* measure always improves the module recognition rate.

P&S pair	ISO	ISO <i>WMSE</i>	Otsu's	Otsu's <i>WMSE</i>	<i>WMSE</i>	<i>WMSE</i> mean	<i>WMSE</i> median
P1-S1	81.61%	85.13%	86.42%	89.62%	90.83%	90.95%	<b>91.24%</b>
P1-S2	85.47%	89.61%	87.69%	91.74%	90.87%	92.27%	<b>92.39%</b>
P1-S3	93.04%	94.97%	93.38%	95.20%	94.82%	<b>95.66%</b>	95.53%
P2-S1	87.76%	90.06%	88.13%	90.63%	89.82%	<b>90.69%</b>	90.58%
P2-S2	89.42%	92.24%	89.58%	<b>92.47%</b>	92.02%	92.37%	92.14%
P2-S3	92.34%	93.94%	92.35%	93.91%	93.62%	<b>94.08%</b>	93.84%
P3-S1	89.05%	92.67%	91.56%	94.37%	<b>95.82%</b>	95.79%	95.49%
P3-S2	91.94%	95.98%	93.04%	96.66%	97.11%	<b>97.39%</b>	97.06%
P3-S3	94.91%	95.89%	94.86%	95.86%	95.82%	<b>96.20%</b>	95.96%
Mean	89.50%	92.28%	90.78%	93.39%	93.41%	<b>93.93%</b>	93.80%
Std	±3.86%	±4.05%	±3.87%	±4.16%	±4.34%	±4.28%	±4.31%

TABLE 6.2: The module recognition rate comparison for different recognition methods and different printer-scanner pairs.

In Table 6.2 we can also see the mean module recognition rates and standard deviation for 10,800 P&S QR code samples. We affirm that the proposed classification methods (*WMSE*, *WMSE mean* and *WMSE median* in the table) have the recognition rate between 90% and 97% and it always improves the recognition results in comparison with the threshold methods. We also note that both the median and the mean characterization modules give the best recognition rate. In addition, we can state that the usage of proposed the *WMSE* measure improves classical thresholding methods by up to 4%. As the mean recognition rate of proposed methods is bigger than 93%, that allows to read correctly the QR code V40 in Low correction level. This fact is important for all approaches where the error correction bits are used for QR code enrichment.

In Table 6.3 we show the improvements between the first step classification method



( $WMSE$  in the table) and the ISO method, and between the second step classification method with mean characterization modules ( $WMSE$  mean in the table) and the ISO method, between the ISO method improved by using the  $WMSE$  measure (ISO  $WMSE$  in the table) and the ISO method, and between Otsu's improved method using the  $WMSE$  measure (Otsu's  $WMSE$  in the table) and Otsu's method. We can conclude that using of the proposed  $WMSE$  measure always improves the recognition rate at least by up to 2.61%. That is bigger than the improvement done by the Otsu's method in comparison with ISO standard method.

P&S pair	$WMSE/$ ISO	$WMSE$ mean/ ISO	ISO $WMSE/$ ISO	Otsu's $WMSE/$ Otsu's
P1-S1	9.22%	9.34%	3.52%	3.20%
P1-S2	5.41%	6.81%	4.15%	4.05%
P1-S3	1.78%	2.62%	1.93%	1.82%
P2-S1	2.06%	2.93%	2.30%	2.51%
P2-S2	2.59%	2.95%	2.82%	2.89%
P2-S3	1.28%	1.74%	1.60%	1.57%
P3-S1	6.77%	6.74%	3.62%	2.81%
P3-S2	5.17%	5.45%	4.04%	3.62%
P3-S3	0.90%	1.28%	0.97%	1.00%
Mean	3.91%	4.43%	2.77%	2.61%

TABLE 6.3: The improvements of module recognition rate in comparison with ISO and Otsu's methods for different printer-scanner pairs.

### 6.5.3 Weight parameter optimization

In this section we aim to optimize the weight parameter  $W$  used in equation (6.3). In equation (6.3) we define the weight parameter  $W \geq 1$ . That is why for the modules with a size of  $3 \times 3$  pixels, since we have  $r = 2$ , there are four particular cases:

- $W = 1$ . In this case only the central pixel is taken into account. This case can not be used because the module is not represented by one single central pixel due to the randomness of the P&S process. Furthermore, if the geometrical correction is not completed correctly the recognition step will be performed incorrectly, and therefore, the QR code will be decoded incorrectly. That is why, as mentioned in Section 6.3, the weight parameter  $W$  should satisfy the condition  $W > 1$ .
- $W \rightarrow \infty$ . In this case the recognition method tends to provide the same recognition results as global thresholding methods. We loose the principal idea of the  $WMSE$  measure because of the difference between the weight given to central pixels and the weight given to border pixels is tiny.
- $W \in ]1, 2[$ . In this case, the weight 4 diagonal neighbor pixels to the central pixel  $p(0, 0)$  is equal to 0. And the binarization have been completed using the weighted values of a central pixel and 4-connected pixels. The recognition results of this case are presented below.



- $W \in [2, \infty[$ . In this case, the weighted values of all pixels are used for module binarization. Experimentally, we show that increasing the weight parameter  $W$  decreases the recognition results, see Fig. 6.13-6.15.

In Fig. 6.12 the weight changes depending on different  $W$  values are presented. We note that the particular values  $W = 1$  and  $W = 10$  can not be used as mentioned below. That is why the weight parameter  $W$  should be chosen in interval  $]1, 10[$ .

W = 1	W = 1.5	W = 2	W = 10																																				
<table border="1" style="border-collapse: collapse; width: 60px; height: 60px;"> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>0</td></tr> </table>	0	0	0	0	1	0	0	0	0	<table border="1" style="border-collapse: collapse; width: 60px; height: 60px;"> <tr><td>0</td><td>0.33</td><td>0</td></tr> <tr><td>0.33</td><td>1</td><td>0.33</td></tr> <tr><td>0</td><td>0.33</td><td>0</td></tr> </table>	0	0.33	0	0.33	1	0.33	0	0.33	0	<table border="1" style="border-collapse: collapse; width: 60px; height: 60px;"> <tr><td>0</td><td>0.5</td><td>0</td></tr> <tr><td>0.5</td><td>1</td><td>0.5</td></tr> <tr><td>0</td><td>0.5</td><td>0</td></tr> </table>	0	0.5	0	0.5	1	0.5	0	0.5	0	<table border="1" style="border-collapse: collapse; width: 60px; height: 60px;"> <tr><td>0.8</td><td>0.9</td><td>0.8</td></tr> <tr><td>0.9</td><td>1</td><td>0.9</td></tr> <tr><td>0.8</td><td>0.9</td><td>0.8</td></tr> </table>	0.8	0.9	0.8	0.9	1	0.9	0.8	0.9	0.8
0	0	0																																					
0	1	0																																					
0	0	0																																					
0	0.33	0																																					
0.33	1	0.33																																					
0	0.33	0																																					
0	0.5	0																																					
0.5	1	0.5																																					
0	0.5	0																																					
0.8	0.9	0.8																																					
0.9	1	0.9																																					
0.8	0.9	0.8																																					

FIGURE 6.12: Weight function  $w(i, j)$  coefficients for 8-connected pixels depending on  $W$  values for modules with size  $3 \times 3$  pixels.

We tested all mentioned methods with several values for parameter  $W$  between 1 and 100. In Fig. 6.13 the recognition results are presented for the printer-scanner pair HP CM1415 - HP CM1415. This printer-scanner pair gave the worst recognition results, both for standard and for proposed binarization methods. We see that the methods using the  $WMSE$  measure are better than the global thresholding methods. In addition, the recognition results for  $W \in ]1, 2[$  do not change significantly, but from value  $W = 2$  the recognition results of the ISO  $WMSE$  method and Otsu's  $WMSE$  method tend to recognition rates of standard ISO and Otsu's method respectively. That is why for this printer-scanner pair the best weight value is  $W \in ]1, 2[$ .

Analogically, the recognition results for printer-scanner pair HP P4015 - Toshiba 455 are presented in Fig. 6.14. This printer-scanner pair (quite good printer and the best scanner) achieves neither worst nor better recognition results. We see that the recognition results of the ISO standard method and Otsu's method are the same in this case. The ISO  $WMSE$  and Otsu's  $WMSE$  methods gave the best recognition results for  $W \in ]1, 2[$ . But the recognition rates for these methods start to decrease and from  $W = 5$  the results are the same when using the standard thresholding methods. At the same time the proposed classification methods are more stable and this improves the recognition results even for  $W = 100$ .

In the end the recognition results for the best printer Brother HL-4150 and the worst scanner HP CM1415 are illustrated in Fig. 6.15. The recognition results of the Otsu's method are better than the results of the ISO standard method. The first step classification method gives the best recognition results. Analogically to two previous figures the best recognition results correspond to  $W \in ]1, 2[$  and the recognition rate of the ISO  $WMSE$  method and Otsu's  $WMSE$  method tend to recognition rates of standard ISO and Otsu's methods by increasing the  $W$  value.

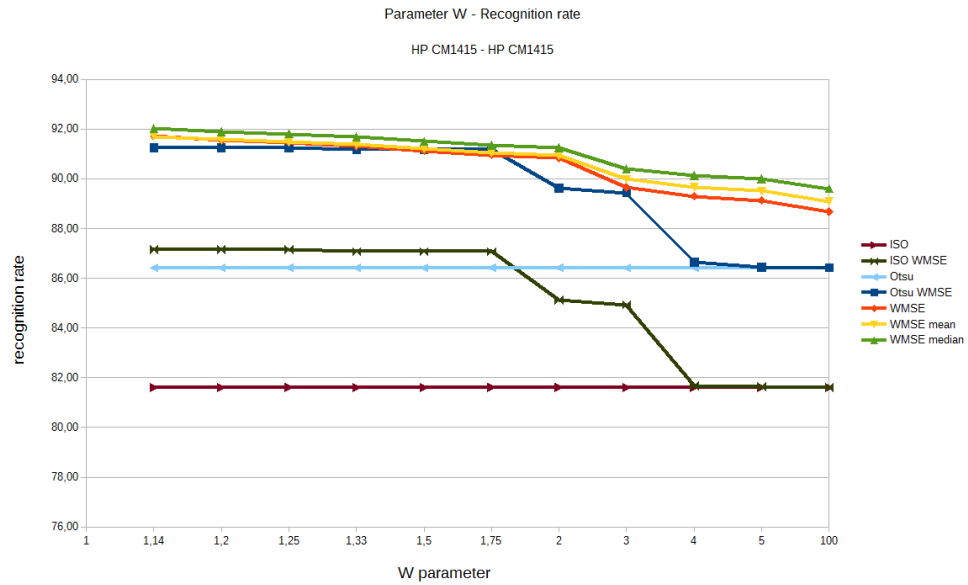


FIGURE 6.13: Recognition rate changes depending on weight parameter  $W$  value for printer-scanner pair HP CM1415 - HP CM1415. The axis  $X$  corresponds to  $W$  value, axis  $Y$  corresponds to recognition rate (%).

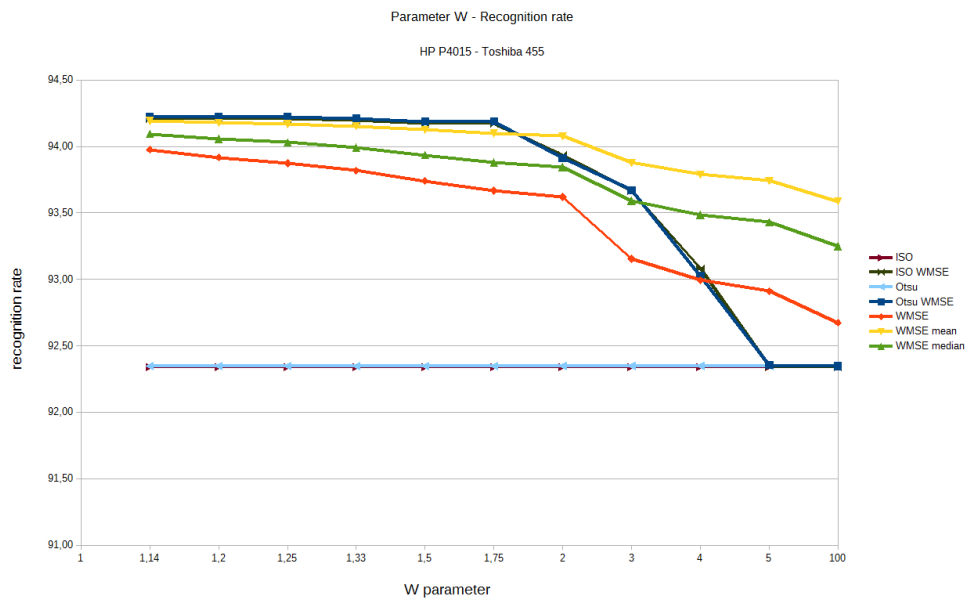


FIGURE 6.14: Recognition rate changes depending on weight parameter  $W$  value for printer-scanner pair HP P4015 - Toshiba 455. The axis  $X$  corresponds to  $W$  value, axis  $Y$  corresponds to recognition rate (%).

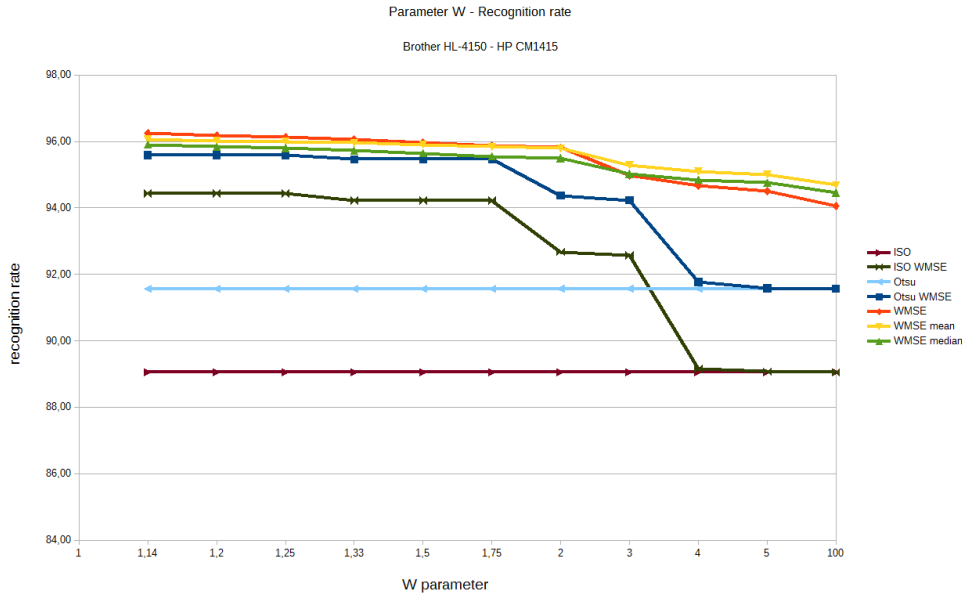


FIGURE 6.15: Recognition rate changes depending on weight parameter  $W$  value for printer-scanner pair Brother HL-4150 - HP CM1415. The axis  $X$  corresponds to  $W$  value, axis  $Y$  corresponds to recognition rate (%).

### 6.5.4 Simulation of camera capture

In all our experiments we use the P&S process due to the application scenario introduced in Section 6.1. In order to simulate the variability of QR code capture with a smartphone camera, we generate several samples with different number of pixels per module ( $7 \times 7$  pixels,  $5 \times 5$  pixels and  $3 \times 3$  pixels). We print and scan these samples using the same resolution (600 dpi). The module recognition results are presented in Table 6.4. We note that the proposed methods are more effective when the module size is tiny. In addition the methods that use the WMSE measure have at least a minor advantage in all experiments.

Module size	ISO	ISO $WMSE$	Otsu's	Otsu's $WMSE$	$WMSE$	$WMSE$ mean	$WMSE$ median
$7 \times 7$	99.61%	99.91%	99.64%	99.91%	<b>99.92%</b>	<b>99.92%</b>	99.91%
$5 \times 5$	98.22%	99.26%	98.37%	99.30%	99.37%	<b>99.44%</b>	99.34%
$3 \times 3$	90.18%	95.24%	92.49%	97.04%	<b>98.38%</b>	98.29%	97.95%

TABLE 6.4: The module recognition rate comparison for different recognition methods and different module sizes.

In addition we added Gaussian noise into the P&S QR codes in order to verify the performance of the first step of our proposed method depending on variance  $\sigma^2$  of Gaussian noise and depending on module size. The results are illustrated in Fig. 6.16. We note that the recognition rates for  $5 \times 5$  and  $7 \times 7$  module sizes have consistently high values.

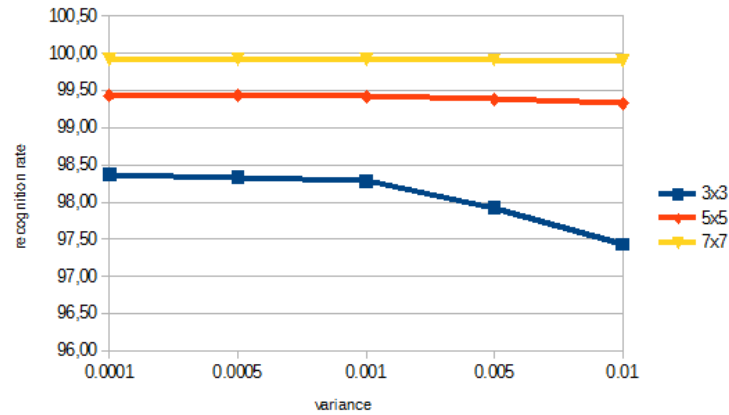


FIGURE 6.16: Recognition rates depend on variance  $\sigma^2$  of Gaussian noise for module sizes  $3 \times 3$ ,  $5 \times 5$  and  $7 \times 7$  using the first step of the proposed classification method.

The QR codes with  $3 \times 3$  module size are more sensitive to Gaussian noise but the recognition rate remains high.

## 6.6 Conclusions

The high density barcodes and more precisely high density QR codes have binarization problems due to frequent changes of black and white modules and impact of P&S process. Most of these problems deteriorate the binarization and the reading process of QR codes. In order to refine the binarization results the Weighted Mean Square Error (WMSE) measure is introduced.

This *WMSE* measure is based on module centrality bias and it increases the significance of central pixels and decreases the significance of border pixels of each module. The *WMSE* can be used for the proposed two step classification method and also for improving the recognition results of standard ISO method or Otsu's global thresholding method.

All performed experiments show that the proposed methods improve the recognition results by up to 5%. The minimal recognition rate with our methods is 93% that means that we can use the Low error correction level (which restores 7% of codewords) and therefore we can store more information in small bar code area.

The contributions of this chapter are published in international conference [144]. Additionally, the extended version is accepted for publication in the international journal "Signal Processing: Image communication".



## Chapter 7

# Textured patterns used for barcodes

### 7.1 Introduction

In our discussion about rich barcodes in Chapter 4, we highlight the popularity of barcodes and the advancement of rich barcodes design, thanks to a huge number of application scenario and their strong features:

- robustness to the copying process,
- easy reading process by any device and any user,
- high encoding capacity enhanced by error correction facilities,
- small size,
- robustness to geometrical distortions.

However, those undeniable advantages also have their counterparts:

1. Information encoded in a barcode is always accessible to everyone, even if it is ciphered and therefore is only legible to authorized users (the difference between "see" and "understand").
2. It is impossible to distinguish an originally printed barcode from its copy due to their insensitivity to the P&S process.

In this chapter, we propose to overcome these shortcomings by enriching the standard barcode encoding capacity. This enrichment is obtained by replacing its black modules by specific textured patterns. Besides the gain of storage capacity (thanks to increasing the alphabet size), these patterns can be designed to be sensitive to distortions of the P&S process. These patterns, that do not introduce disruption in the standard reading process, are always perceived as black modules by any barcode reader.

Therefore we obtain a two level barcode: a first (public) level accessible for any standard barcode reader, therefore it keeps the strong characteristics of the barcode; and a second (private) level that improves the capacities and characteristics of the initial barcode.

The information in the second level is encoded by using a  $q$ -ary ( $q \geq 2$ ) code with error correction capacities. That is the main difference of the proposed two level barcode in comparison with other barcodes with supplementary level (see Section 4.5). This information is invisible to standard barcode readers because they perceive the textured patterns as black modules. Therefore, the second level can be used for private message sharing. Additionally, thanks to the pattern sensitivity we envisage the application to printed document authentication.

In this chapter, all descriptions are done enhancing the capacities of a classic QR code, proposing as example a new two level QR (2LQR) code. Nevertheless, the described procedures can be easily extended to any standard barcode cited in Section 4.2.

This chapter presents two application scenario for proposed 2LQR code. First application scenario is storage of information in two levels, that is also called message sharing scenario. The generation process, storage capacity and reading process for this scenario are presented in Section 7.2. The experimental results of 2LQR code for message sharing scenario are shown in Section 7.3. Second application scenario is document support authentication using 2LQR code. Section 7.4 introduces authentication system, reading and authentication process. Then, Section 7.5 shows the experimental results of authentication test and several simple attacks performed over 2LQR code. Then, we overview both scenarios in Section 7.6. And we conclude in Section 7.7.

## 7.2 New rich QR with two storage levels

In Section 4.2.2, we describe the specific structure of the standard QR code. Like the standard QR code, the 2LQR code has the same specific structure, which consists of position tags, alignment patterns, timing patterns, version and format patterns. However, in the standard QR code, we have white and black modules and in the 2LQR code we have white modules and textured modules instead of black modules. This replacement of black modules by textured modules does not disrupt the standard QR code reading process. But it allows us to have a second storage level, which is invisible to the standard QR code reader. This second level contains the private message, encoded with a  $q$ -ary ( $q \geq 2$ ) code with error correction capacity. The textured modules are the textured patterns described in Section 5.2. As it was discussed, these textured patterns have specific features and are used for private message  $M_{priv}$  storage in the proposed 2LQR code.

The comparison of standard QR code with the proposed 2LQR is illustrated in Fig. 7.1. The standard QR code is illustrated in Fig. 7.1.a. Likewise the proposed 2LQR code for private message sharing is illustrated in Fig. 7.1.b. We can note the replacement of black modules in the standard QR code by textured patterns in the 2LQR code. The

alphabet dimension of this 2LQR code example is equal to  $q = 3$  (i.e. 3 different textured patterns are used for private level generation). Nevertheless, the alphabet can be extended by rising the pattern number used.

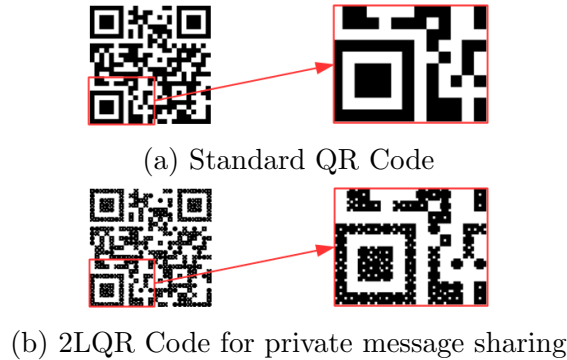


FIGURE 7.1: A comparison of a) Standard QR code, with b) Proposed 2LQR code for private message sharing scenario.

We store the samples of used textured patterns in the position tags according to two targets: to store the textured patterns used and to use these stored samples for representative patterns creation. These textured patterns are mixed by using permutation  $\sigma$  in order to:

- enhance the representativity of textured samples and P&S impact to each of them;
- reduce the visual difference among the patterns, that makes it more difficult for attackers to rebuild;
- break the texture effect in position patterns.

The black modules in the position tags are replaced respecting the permutation  $\sigma$ . These textured patterns are used for pattern detection. In this case, we can decode the private message  $M_{priv}$ , if we know the permutation used  $\sigma$  and the Error Correction Code (ECC) algorithm used for message encoding. For this, we have to reconstruct the classes of textured patterns by using the permutation  $\sigma$ , then create the characterization patterns for each used textured pattern, and compare the characterization patterns with the textured patterns used in 2LQR code in order to recognize the patterns.

### 7.2.1 Generation scheme

We define the public  $M_{pub}$  and the private  $M_{priv}$  messages. The public message is encoded into the QR code using the standard algorithm. In the same time, the private message is encoded using the chosen ECC algorithm. Finally, both QR code with public message  $M_{pub}$  and encoded private message are consolidated in the 2LQR code with two stored messages. The 2LQR code generation steps are illustrated in Fig. 7.2.

**Public message  $M_{pub}$  storage** (Fig. 7.2, block 1). The public message  $M_{pub}$  is stored



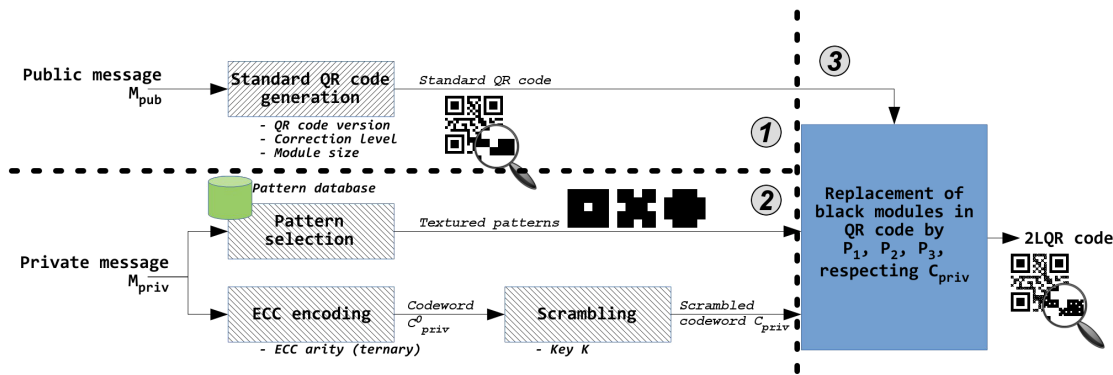


FIGURE 7.2: Overview of 2LQR code generation steps.

in the standard QR code, using the classical generation method described in [4]. The standard QR code generation algorithm includes the following steps. First of all, we analyze the message, select the most optimal mode (numeric, alphanumeric, byte or Kanji) and encode the message  $M_{pub}$  using the shortest possible string of bits. This string of bits is split up into 8 bit long data codewords. Then, we choose the error correction level and generate the error correction codewords using the Reed-Solomon code. After that, the data and the error correction codewords are arranged in the right order. Then, we apply the best (for our data) mask pattern, in order to be sure that the generated QR code can be read correctly. After this manipulation, the codewords are placed in a matrix respecting a zigzag pattern, starting from the bottom-right corner. The final step is to add the function patterns (position tags, alignment, timing, format and version patterns) into the QR code.

**Private message  $M_{priv}$  encoding** (Fig. 7.2, block 2). We have the private raw-bit string to encode, we suggest using ECC to ensure the message error correction after the P&S operation. We use the block codes, and more precisely cyclic codes (or polynomial-generated codes) such as Golay code [132] or Reed-Solomon code, for message encoding. Cyclic codes can be defined in matrix form and polynomial form. Any cyclic code  $C$  is defined by  $[n, k, d]$  parameters, where  $n$  is the length of the codeword,  $k$  is the number of information digits in the codeword,  $d$  is the minimum distance between distinct codewords. The  $n - k$  digits in the codeword are called parity-check digits, and in ECCs these digits are used for error detection and correction. The minimum distance  $d$  of code  $C$  ensures that up to  $(d - 1)/2$  errors can be corrected by the code  $C$ .

Let  $R = A[x]/(x^n - 1)$  be a polynomial ring over a Galois field  $A = GF(q)$ . The cyclic code  $C$  elements are defined with polynomials in  $R$  so that the codeword  $(c_0, c_1, \dots, c_{n-1})$  maps to the polynomial  $c_0 + c_1x + \dots + c_{n-1}x^{n-1}$ , and the multiplication by  $x$  corresponds to a cyclic shift. The code  $C$  is generated by a generator polynomial  $g(x)$ , which is the code polynomial of the minimum degree in a  $(n, k)$  cyclic code  $C$ . Therefore, the generator polynomial  $g(x)$  is a factor of polynomial  $x^n - 1$ .

Let  $k$  informative digits of message be represented by a polynomial  $m(x)$ , of degree, at

most  $k - 1$ . Then, the codeword  $c(x)$  is the polynomial of the form:

$$c(x) = m(x)g(x), \quad (7.1)$$

$$c(x) = c_0 + c_1x + \cdots + c_{n-1}x^{n-1}. \quad (7.2)$$

Therefore, the encoded informative digits are  $(c_0, c_1, \dots, c_{n-1})$ . We encode the private message  $M_{priv} = (m_1^{priv}, \dots, m_k^{priv})$  using ECC  $[n, k]$ . First of all, we represent the  $M_{priv}$  in polynomial form  $m_{priv}(x)$ . Then, we calculate the polynomial form of the codeword  $c_{priv}(x) = m_{priv}(x)g(x)$ , and we obtain the codeword  $C_{priv}^0$ .

After that, as illustrated in Fig. 7.2, we scramble the codeword  $C_{priv}^0$  using the key  $K$ . Thus, the scrambled codeword digits are:

$$C_{priv} = (c_0^{priv}, c_1^{priv}, \dots, c_{n-1}^{priv}). \quad (7.3)$$

**Black module replacement** (Fig. 7.2, block 3). We insert the codeword  $C_{priv}$  in standard QR code by replacing the black modules with textured patterns  $P_1, \dots, P_q$  respecting the codeword  $C_{priv}$ , starting from the bottom-right corner. Then, we replace the black modules in the position tags by textured patterns with respect to the chosen permutation  $\sigma$ , see Fig. 7.1.b.

## 7.2.2 Textured pattern selection

The textured patterns  $P_i, i = 1, \dots, q$  are images of size  $r \times r$  pixels introduced in Section 5.2. We choose  $q$  patterns from a database of  $Q \gg q$  textured patterns, which respect all mentioned characteristics in Section 5.2.4: the patterns are binary, have the same density (ratio of black pixels), have related spectra and respect criteria (5.2) and (5.3). The reading capacity of private level depends on pattern density: a large density value can disable the reading process of private level.

The two criteria (5.2) and (5.3) can be rewritten in the form:

$$\begin{aligned} \forall l \in \{1, \dots, q\}, cor(P_l, S_l) &= \max_{\forall l' \in \{1, \dots, q\}} (cor(P_l, S_{l'})) \\ &= \max_{\forall l' \in \{1, \dots, q\}} (cor(P_{l'}, S_l)). \end{aligned} \quad (7.4)$$

The condition (7.4) is valid for all values of  $l'$  ( $\forall l' \in \{1, \dots, q\}$ ), but if we exclude the value  $l' = l$ , the condition (7.4) can be rewritten in the form:

$$\begin{aligned} \forall l, l' \in \{1, \dots, q\}, cor(P_l, S_l) - \max_{l' \neq l} (cor(P_l, S_{l'})) &\geq \varepsilon, \\ cor(P_l, S_l) - \max_{l' \neq l'} (cor(P_{l'}, S_l)) &\geq \varepsilon, \end{aligned} \quad (7.5)$$

where  $\varepsilon > 0$ .

The condition (7.5) represents a new criteria which is the minimum distance between the best correlation score and the second best one. This distance should be greater than a given  $\varepsilon$  threshold.

Therefore, only the textured patterns which respect the condition (7.5) can be combined and used for 2LQR code generation.

### 7.2.3 Storage capacity study

In this section we want to discuss the storage capacities of proposed 2LQR code. The schematic calculation of 2LQR code storage capacity is illustrated in Fig. 7.3. Let  $N^2$  be the number of modules in a standard QR code. As QR code construction aims to have an approximately equal number of black and white modules, we can suppose that  $N^2/2$  is approximately the number of black modules in standard QR code. We have three position tags, each tag has 33 black modules. That is why, there are approximately  $(N^2/2 - 3 \times 33)$  black modules in QR code, that could be replaced by textured patterns.

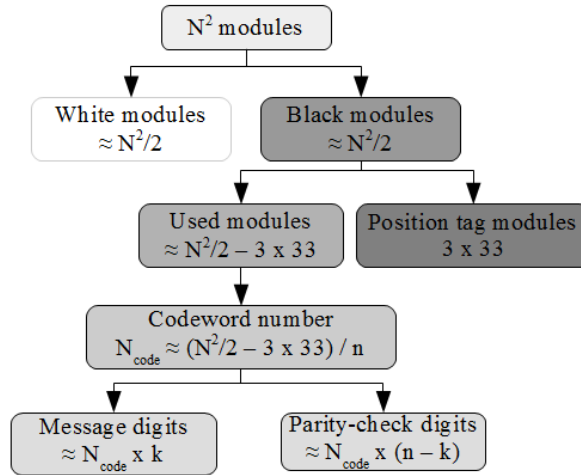


FIGURE 7.3: Storage capacity of the 2LQR for QR code of size  $N \times N$  modules, where  $n$  is the codeword length,  $k$  is the number of information digits,  $n - k$  is the number of parity-check digits.

We remind that  $n$  is the total number of digits in a codeword,  $k$  is the number of message digits,  $n - k$  is the number of error correction bits in a codeword. Therefore, the number of codewords, that could be inserted in the second level of a 2LQR code, is approximately equal to  $N_{code} \approx (N^2/2 - 3 \times 33)/n$ . And the number of message digits is approximately equal to  $N_{code} \times k$ , that is why the length of message (in bits) is approximately equal to  $\log_2(q) \times N_{code} \times k$ , where  $q$  is the alphabet dimension.

These formulas allow us to calculate the maximal storage capacity of the 2LQR for a fixed module number  $N^2$  i.e. for a fixed QR code version. At the same time, we can define the version of the QR code, which can be used for insertion into a fixed message digit number.

We calculate the storage capacity for several QR code versions in Table 7.1. The use of second storage level increases the storage capacity of QR code up to 30% – 310%.

Version	Module number	Number of black modules	ECC level	Public message (bits)	Private message (bits)		
					$q = 2$	$q = 3$	$q = 8$
					G [23, 12]	G [11, 6]	RS [7, 3]
2	$25 \times 25$	213	L	272	108	180	270
			H	128			
5	$37 \times 37$	585	L	864	300	504	747
			H	368			
8	$49 \times 49$	1, 101	L	1, 552	564	950	1, 413
			H	688			
40	$177 \times 177$	15, 565	L	23, 648	8, 112	13, 456	20, 007
			H	10, 208			

TABLE 7.1: Storage capacity for several QR code versions.

*Example:* If we take the Golay binary code [23, 12, 7], i.e.  $n = 23, k = 12, n - k = 11, q = 2$ , and we take the standard QR code version V2 :  $N^2 = 25 \times 25 = 625$ . The number of codewords is approximately equal to  $N_{code} = (625/2 - 99)/23 \approx 9$ . Therefore, we can insert  $\log_2(2) \times N_{code} \times k = 9 \times 12 = 108$  binary digits (bits) of private message.  $\square$

#### 7.2.4 Reading process

We focus on printed QR codes that imply the use of the printing process and the scanning process. As it was discussed in Chapter 3, the P&S process is a stochastic process, which blurs and modifies the output image. We recall that  $S_l, l = 1, \dots, q$  is the P&S degraded version of the textured pattern  $P_l, l = 1, \dots, q$ .

The overview of the 2LQR code reading process is illustrated in Fig. 7.4. First, the geometrical distortion of P&S 2LQR code has to be corrected during the **pre-processing step**. The position tags are localized by the standard process [4] to determine the corner coordinates. Linear interpolation is used to re-sample the P&S 2LQR code. Therefore, at the end of this step, the 2LQR code has the correct orientation and original size  $N \times N$  pixels.

The second step is the **module classification** performed by any threshold method. We use global threshold, which is calculated as a mean value of the whole P&S 2LQR code. Then, if the mean value of the block  $r \times r$  pixels is smaller than global threshold, this block belongs to the black class ( $BC$ ). Otherwise, this block belongs to the white class ( $WC$ ). The result of this step is two classes of modules.

In the next step, we complete the **standard QR code reading process**, that corrects the module classification errors (i.e. the  $BC$  class contains only textured patterns) and decodes the public message  $M_{pub}$ .

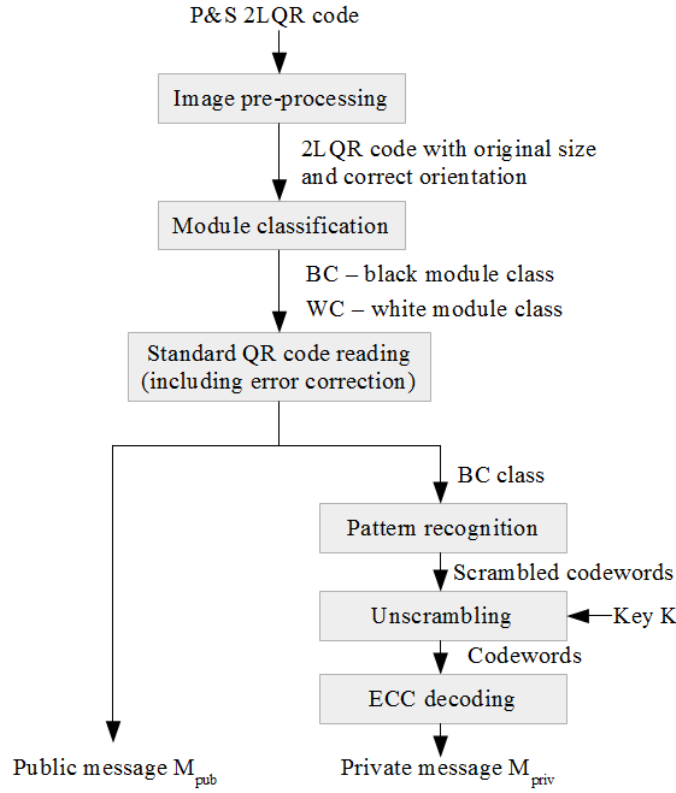


FIGURE 7.4: Overview of 2LQR code reading process.

Finally, we use the BC class for **pattern recognition** of the textured pattern in P&S 2LQR code. The class  $BC$  contains the textured patterns  $BP_i, i = 1, \dots, N_{code} \times n$ , where  $N_{code} \times n$  is the total number of codeword digits,  $N_{code}$  is the number of codewords,  $n$  is the number of digits in the codeword. Therefore, we have  $N_{code} \times n$  textured patterns, which belong to  $q$  classes.

In the proposed pattern detection method, we compare the P&S patterns ( $BP_i, i = 1, \dots, N_{code} \times n$ ) with characterization patterns by using the Pearson correlation (5.1). The chosen patterns  $P_1, \dots, P_q$  are stored in position tags, in order to create the mean and median characterization patterns (see Section 5.4.1). We have used the permutation  $\sigma$  to mix the textured patterns in position tags. Now, we apply the inverse permutation  $\sigma^{-1}$  and obtain  $q$  sets (as  $q$  is the dimension of our alphabet) of the P&S representative patterns. Each position tag consists of 33 black modules (for any QR code version), in total we have 99 representative patterns. That is why the dimension of each representative set is equal to an integer  $v = \lceil 99/q \rceil$ . We calculate the characterization patterns as mean images (5.4) and median images (5.5).

The flowchart of the proposed pattern recognition method is presented in Fig. 7.5. In the class BC, we have  $N_{code} \times n$  textured patterns  $BP_i, i = 1, \dots, N_{code} \times n$ . For each textured pattern  $BP_i, i = 1, \dots, N_{code} \times n$ , we look for a maximum correlation value  $cor_l$  with characterization patterns  $CP_1, \dots, CP_q$ , i.e.  $cor_l = \max_{j=1, \dots, q} \{cor(BP_i, CP_j)\}$ . Then, the codeword digit is equal to  $l$ :  $c'_i = l$ . So, in the output of this method we obtain the

codeword  $C'_{priv} = (c'_1, \dots, c'_{N_{code} \times n})$ .

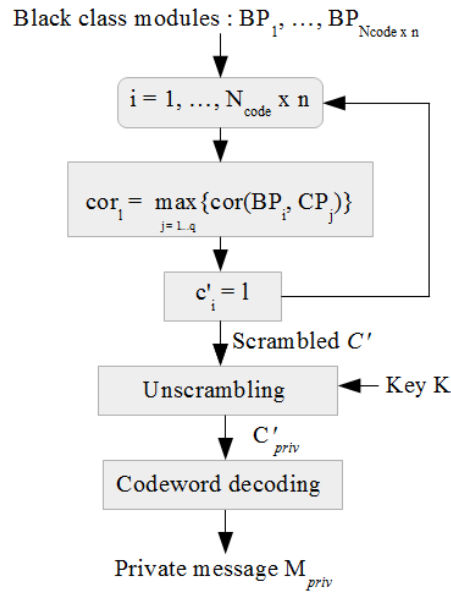


FIGURE 7.5: The pattern recognition method scheme by using characterization patterns  $CP_1, \dots, CP_q$ .

The last steps of the 2LQR code reading process are **unscrambling** using key  $K$  and **ECC decoding** of the obtained codeword  $C'_{priv}$ . We use the parity-check digits for error detection and correction. For error correction and decoding, we use one of the classical ECC decoding algorithms (i.e. error syndrome decoding, maximum likelihood decoding algorithms), as a result we have the restored private message  $M_{priv}$ .

### 7.3 Experiments with 2LQR code

We illustrate both the generation steps of the 2LQR code in Section 7.3.1 and the message extraction steps in Section 7.3.2. Then, we discuss the storage capacities of the 2LQR code in Section 7.3.3. After, the reading capacities of public and private levels are presented in Section 7.3.4.

**Application scenario.** For example, we propose to store the Surname, First name, Date of Birth and Place of Birth of a person in the public level. Then, the secret information, which is the number of his bank account, is encoded in the private level.

**Setup.** In these experiments, the version V2 of the QR code in Low error correction level is used. This version has  $25 \times 25$  module size and can store 272 bits of data.

**Error correction code.** We encode the private information with the ternary Golay code [11, 6, 5]. Using the calculation strategy described in Section 7.2.3, we determine the storage capacities of public and private levels for this example in Table 7.2.

	Public level (Standard QR code V2 Low)	Private level (2LQR code V2 with Golay [11, 6, 5])
Total number of modules	625	312
Supporting information number of modules	324	213
Payload in digits	–	114
Message bits	272	180

TABLE 7.2: Storage capacity information of public and private levels of the 2LQR code version V2.

### 7.3.1 2LQR code generation

The 2LQR generation, as was previously mentioned in Section 7.2.1, consists of four steps: standard QR code generation, codeword generation, pattern selection and replacement of black modules in QR code.

**Standard QR code generation.** The standard QR code with public message  $M_{pub}$  "John Doe - 13/05/1958 - New York" is generated by using a free online QR code generator. The generated standard QR, version V2, is illustrated in Fig. 7.6. The actual size of this QR code is  $1.2 \times 1.2 \text{ cm}^2$ .

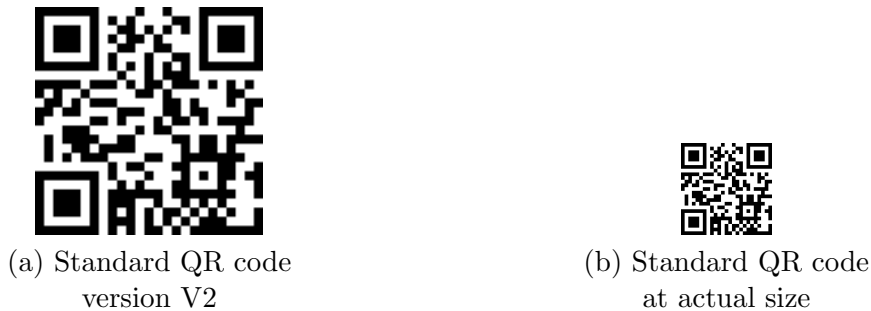


FIGURE 7.6: The example of a) Standard QR code with public message  $M_{pub}$ , b) Standard QR code at actual size defined at 600 dpi, ( $1.2 \times 1.2 \text{ cm}^2$ ).

**Codeword generation.** The private message  $M_{priv}$  with a 180 bit length, which corresponds to 114 ternary digits, is defined. We encode this message with ternary Golay code [11, 6, 5]. The codeword  $C_{priv}^0$  has 209 digit length. The last 4 accessible digits are defined randomly. Then, we use the key  $K$  for scrambling of the codeword  $C_{priv}^0$ , and we obtain the codeword  $C_{priv}$ .

**Pattern selection.** We chose three textured patterns  $P_1$ ,  $P_2$  and  $P_3$ , which are illustrated in Fig. 7.7. These patterns have some particular characteristics:

- they have a size of  $12 \times 12$  pixels;
- they are binary;
- they have the same number of black pixels  $b = 60$  (that corresponds to nearly 42%);
- they have spectra related between them.

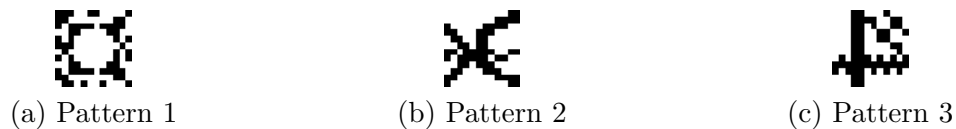


FIGURE 7.7: The three textured patterns used for private level generation: a) Pattern 1, b) Pattern 2, c) Pattern 3.

The original and the P&S degraded versions of these patterns satisfy the condition (7.5) with  $\varepsilon = 0.60$ . The correlation values as well as the minimal distance ( $\min\{cor(P_l, S_l) - \max_{l' \neq l} (cor(P_l, S_{l'}))\}$ ) are illustrated in Table 7.3. We note that the diagonal correlation values  $cor(P_l, S_l), l = 1, \dots, 3$ , which are equal to  $cor(P_1, S_1) = 0.6787$ ,  $cor(P_2, S_2) = 0.7624$  and  $cor(P_3, S_3) = 0.6404$ , are the maximum in line and in column. Then, we calculate the distances among  $cor(P_l, S_l)$  and  $cor(P_l, S_{l'}), l, l' \in \{1, 2, 3\}, l' \neq l$ , i.e.  $cor(P_l, S_l) - cor(P_l, S_{l'})$  and  $cor(P_l, S_l) - cor(P_{l'}, S_l), l, l' \in \{1, 2, 3\}, l' \neq l$ . We notice, that all distances are bigger, then threshold  $\varepsilon = 0.6$ . Therefore, the condition (7.5) is respected. We can use these patterns for second level generation.

	$P_1$	$P_2$	$P_3$
$S_1$	<b>0.6787</b>	-0.1014	0.0244
$S_2$	-0.0910	<b>0.7624</b>	-0.0114
$S_3$	-0.0469	-0.0088	<b>0.6404</b>
minimal distance	0.6543	0.7738	0.6161

TABLE 7.3: The correlation values for the original patterns  $P_1, P_2, P_3$  and its P&S degraded versions  $S_1, S_2, S_3$ , the minimal distances among diagonal correlation values and all other correlation values in the same line and the same column  $\min\{cor(P_l, S_l) - \max_{l' \neq l} (cor(P_l, S_{l'}))\}$ .

**The replacement of black modules in the QR code.** The private level of the 2LQR code is constructed by replacement of black modules by textured patterns illustrated in Fig. 7.7.a-c with respect to the codeword  $C_{priv}$ . We replace the black modules starting from the bottom-right corner of the QR code.

The black modules in position tags are replaced by patterns  $P_1, P_2$  and  $P_3$  with respect to permutation  $\sigma$ . The patterns stored in position tags are used to build the characterization patterns  $CP_l, l = 1, 2, 3$ . The example of the 2LQR suitable for the *private message sharing scenario* is illustrated in Fig. 7.8. The 2LQR size is equal to  $1.2 \times 1.2 \text{ cm}^2$ . The public level of this 2LQR code is readable by any QR code reader.

### 7.3.2 Message extraction

For the pattern detection experiments, we have printed the same 2LQR code 1,000 times in 600 dpi using Brother HL-4150 printer. Then, we scanned each printed 2LQR code in 600 dpi using Canon LIDE 210 scanner. The 2LQR code after one P&S operation is called printed 2LQR code. Fig. 7.9 illustrates an example of the P&S 2LQR code. In comparison with the original 2LQR code (Fig. 7.8), these images (Fig. 7.9.a-b) are



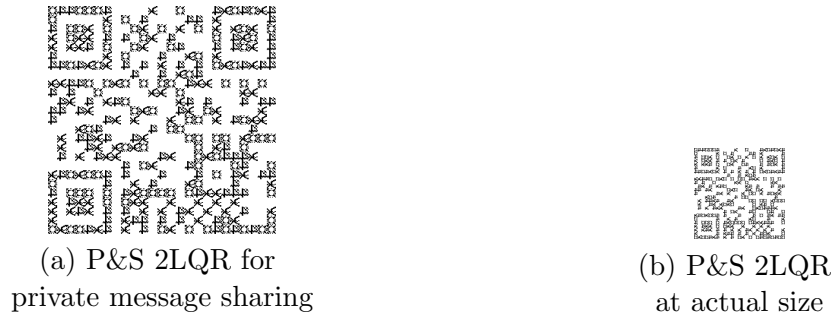


FIGURE 7.8: The example of a) 2LQR code for private message sharing. The alphabet dimension is equal to  $q = 3$  and b) 2LQR code at actual size.

blurred and in gray-level (instead of being binary). In addition the printed 2LQR code samples are copied (using Copy Machines (CM): the Toshiba e355 in standard mode and with maximal contrast (CM1 and CM2 respectively), the RICOH Aficio MP C2050 (CM3), the Toshiba e456 (CM4) and the Toshiba e256 (CM5)) and rescanned (using the same Canon LIDE 210 scanner). The samples after two P&S operations are called copy 2LQR codes.

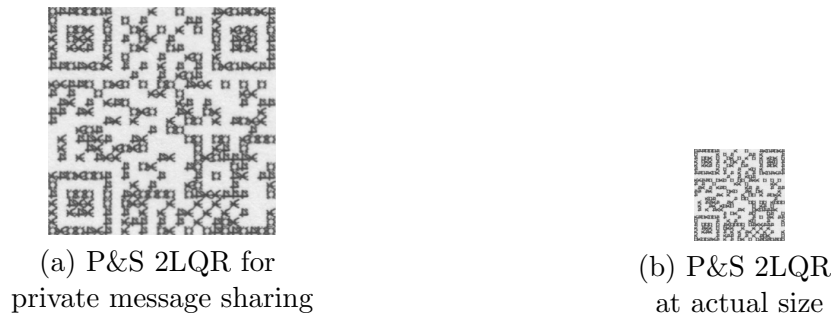


FIGURE 7.9: The example of a) P&S 2LQR code for private message sharing and b) P&S 2LQR code at actual size defined at 600 dpi.

For printed and copied 2LQR codes, we apply the proposed detection method with characterization patterns (mean and median) and with original patterns (as the original patterns can be secretly transmitted to the receiver). The detection results are presented in Table. 7.4. The error probability of pattern detection by using both characterization patterns and original patterns is equal to 0.00%. However, the error probability of pattern detection after copying process is equal to 0.00% – 1.07%. The best pattern detection results are obtained using CM1, the error recognition rate is equal to 0.05%. The worst pattern detection results obtained by use of the same copy machine with maximal contrast (CM2), and the error probability of pattern detection is equal to 0.66%. We highlight the best detection results by using bold font. We note that the mean characterization patterns perform better detection results in copy 2LQR codes.

We apply the unscrambling operation using key  $K$  to the sequence of numbers, which corresponds to detection patterns. Since our private message was encoded using ternary Golay ECC, we apply the error correction and decoding algorithm and get the private

		Printed 2LQR code	Copy 2LQR codes using				
			CM1	CM2	CM3	CM4	CM5
Original	% of $P_1$	0.00%	0.00%	0.00%	0.03%	0.06%	0.12%
	% of $P_2$	0.00%	0.17%	0.69%	0.17%	0.29%	0.32%
	% of $P_3$	0.00%	0.19%	2.35%	0.41%	0.11%	0.22%
	<b>Total</b>	0.00%	0.13%	1.07%	0.21%	0.15%	0.22%
Mean	% of $P_1$	0.00%	0.00%	0.00%	0.00%	0.06%	0.12%
	% of $P_2$	0.00%	0.17%	1.56%	0.41%	0.26%	0.38%
	% of $P_3$	0.00%	0.00%	0.38%	0.05%	0.00%	0.00%
	<b>Total</b>	0.00%	0.06%	<b>0.66%</b>	<b>0.15%</b>	<b>0.11%</b>	<b>0.16%</b>
Median	% of $P_1$	0.00%	0.00%	0.03%	0.00%	0.03%	0.12%
	% of $P_2$	0.00%	0.14%	1.62%	0.43%	0.32%	0.41%
	% of $P_3$	0.00%	0.00%	0.46%	0.05%	0.00%	0.00%
	<b>Total</b>	0.00%	<b>0.05%</b>	0.71%	0.16%	0.12%	0.17%

TABLE 7.4: Pattern detection results after P&S process in the 2LQR code with  $q = 3$ ,  $\varepsilon = 0.60$ .

message  $M_{priv}$ . The error probabilities of incorrect digit decoding are presented in Table 7.5. We conclude that all errors caused by incorrect pattern detection are corrected by Golay error correction algorithm: we have 99% correctly detected 2LQR codes. We always can extract both public and private messages from the 2LQR code.

		Error probability of digit decoding after ECC	Use of characterization patterns		
			Original	Mean	Median
Printed 2LQR code		0.00%	0.00%	0.00%	
Copy 2LQR code using	CM1	0.29%	0.29%	0.29%	
	CM2	0.73%	0.55%	0.57%	
	CM3	0.05%	0.22%	0.22%	
	CM4	0.11%	0.00%	0.00%	
	CM5	0.22%	0.00%	0.00%	

TABLE 7.5: Error probability of message decoding after Golay error correction algorithm for  $q = 3$ ,  $\varepsilon = 0.60$ .

### 7.3.3 Storage capacity analysis

In this section we aim to study the storage capacity of the 2LQR code, using a fixed surface equal to  $1.2 \times 1.2 \text{ cm}^2$  and a fixed pattern density equal to approximately 42%. We can increase the storage capacity of the 2LQR both by:

- increasing the value of  $q$ , which is the number of digits and textured patterns;
- increasing the number of modules (raise the QR code version and reduce the size  $r \times r$  pixels of the textured patterns).

In both cases, we found some problems in pattern detection. Therefore, we have to find a pattern size and a pattern number trade-off.

**In the first experiment**, we have increased the storage capacity of the 2LQR code by increasing the dimension  $q$  of the alphabet set. We have used the version V2 of QR code for storage of the public message. Therefore, the length of the public message  $M_{pub}$  is constant and equal to 272 bits. The number of message bits is increased by increasing the dimension  $q$  of the alphabet, see Table. 7.6. For this experiment, we chose three different alphabet dimensions with three different ECC: binary Golay [23, 12] ( $q = 2$ ), ternary Golay [11, 6] ( $q = 3$ ) and 8-ary Reed-Solomon [7, 3] ( $q = 8$ ). Then, we use ECC for the encoding of the private message  $M_{priv}$ . For example the use of binary alphabet  $q = 2$  can increase the storage capacity from 272 bits up to 380 bits, the use of ternary alphabet  $q = 3$  increases the storage capacity up to 452 bits and the use of 8-ary alphabet  $q = 8$  increases the storage capacity up to 1,082 bits.

Alphabet dimension $q$	Message bit number in		Total number of message bits
	public level	private level	
$q = 2$	272	108	380
$q = 3$	272	180	452
$q = 8$	272	810	1,082

TABLE 7.6: The storage capacity changes in respect of  $q$  ( $q = 2, 3, 8$ ) alphabet in the 2LQR code.

We have had 1,000 P&S samples of each 2LQR code (i.e. the 2LQR codes with the number of textured patterns equal to  $q = 2$ ,  $q = 3$  and  $q = 8$ ). We applied the proposed pattern detection method and the ECC decoding algorithms to each sample. Then, we compared the error probability of pattern detection and digit decoding in Table 7.7. We can see that the number of incorrect pattern detection increases when we increase the alphabet dimension  $q$ , but these errors are successfully corrected by ECC. We note that the error probability of pattern detection is between 0% and 0.51%, that is still acceptable, as the message is correctly extracted after ECC. From this experiment, we conclude that the increasing of alphabet dimension  $q$  is a valid solution to expand the 2LQR code storage capacity.

Alphabet dimension $q$	Error probability of					
	pattern detection			digit decoding		
	Mean	Median	Original	Mean	Median	Original
$q = 2$	0.00%	0.00%	0.00%	0.00%		
$q = 3$	0.06%	0.08%	0.02%	0.00%		
$q = 8$	0.39%	0.51%	0.11%	0.00%		

TABLE 7.7: The error probability of pattern detection and digit decoding after ECC algorithm for different alphabet dimensions ( $q = 2, 3, 8$ ).

**In the second experiment**, we have increased the storage capacity of the 2LQR code by decreasing the size of the textured patterns. The patterns in Section 7.3.1 have a size of  $12 \times 12$  pixels, and the real size of QR code is  $300 \times 300$  pixels or  $1.2 \times 1.2 \text{ cm}^2$ . In this experiment, we fix the size of the QR code approximately equal to  $1.2 \times 1.2 \text{ cm}^2$

and the alphabet dimension  $q = 3$ , but we change the size of patterns and the version of the QR code. In our comparison, we use a pattern size of  $12 \times 12$  pixels and QR code version V2, a pattern size of  $8 \times 8$  pixels and QR code version V5, and a pattern size of  $6 \times 6$  pixels and QR code version V8. With this approach we can significantly increase the storage capacity of the 2LQR code, as illustrated in Table 7.8. We increase the storage capacity of QR code V2 from 272 bits up to 452 bits. The smaller patterns increase the storage capacity of QR V8 from 1,552 bits to 2,502 bits.

Pattern size (pixels)	QR code		Message bit number		Total message bit number
	version	size	public	private	
$12 \times 12$	V2	$300 \times 300$	272	180	452
$8 \times 8$	V5	$296 \times 296$	864	504	1,368
$6 \times 6$	V8	$294 \times 294$	1,552	950	2,502

TABLE 7.8: The storage capacity changes in respect of pattern size decreasing in 2LQR code with  $q = 3$ .

Pattern size (pixels)	Error probability of					
	pattern detection			digit decoding		
	Mean	Median	Original	Mean	Median	Original
$12 \times 12$	0.06%	0.08%	0.02%	0.00%	0.00%	0.00%
$8 \times 8$	3.14%	2.95%	0.23%	2.33%	2.31%	0.00%
$6 \times 6$	6.45%	6.32%	1.06%	6.47%	5.98%	0.08%

TABLE 7.9: The error probability of incorrect bit decoding after ECC algorithm for different sizes of textured patterns with  $q = 3$ .

Similarly to the first experiment, we have had 1,000 P&S samples of each 2LQR code type (i.e. the 2LQR codes with textured patterns of size  $6 \times 6$ ,  $8 \times 8$  and  $12 \times 12$  pixels). We applied the proposed pattern detection method and the ECC decoding algorithms for each sample and we compared the error probability of each pattern detection and digit decoding in Table 7.9. We note that the decreasing of pattern size slightly decreases the pattern detection and the successful decoding of the message. Nevertheless, we can correctly decode the message encoded in the 2LQR code by using  $8 \times 8$  size textured patterns. The error probability of the second level digit decoding by using  $6 \times 6$  size patterns is equal to 0.08%, that means the decoding process would require several tentative captures to be completed. The error probability of pattern detection varies from 0.02% until 6.45% depending on the size of the textured patterns.

From this experiments, we can see that storage capacity increases with decreasing the pattern size (here we reduce the pattern size in twice), nevertheless that introduces more difficulties to read and restore the private message. However, we should note, that the detection using the original patterns is always better, than the use of characterization patterns.

### 7.3.4 Reading capacity of first and second levels

In this section we study the reading capacity of the first and second levels of the 2LQR depending either on pattern density or on pattern size.

Firstly, we change the **pattern density** from 10% to 90%, but we fix the pattern size to  $8 \times 8$  pixels and the actual size of the 2LQR code to  $1.2 \times 1.2 \text{ cm}^2$ . The pattern detection results are illustrated in Fig. 7.10. We note that the public level (the information stored in the standard QR code) is always readable, even when the density is low. At the same time, the number of textured pattern detection errors increases when we increase the density in the private level. Nevertheless, thanks to ECC, we can extract the correct private message for patterns even with a density of 80%.

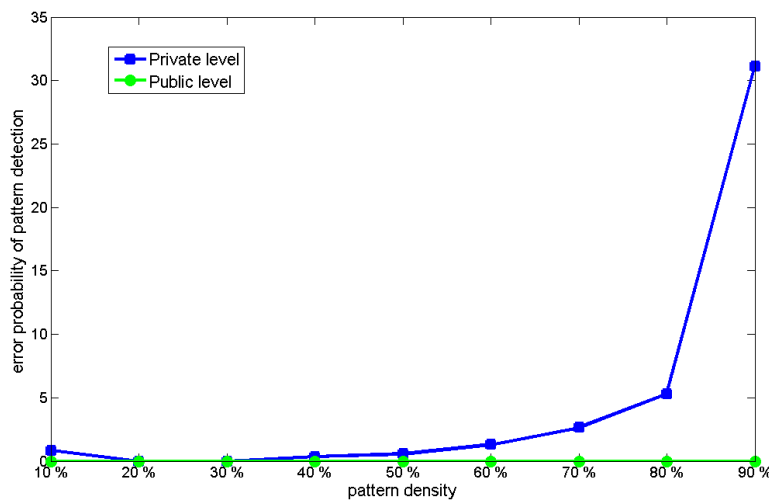


FIGURE 7.10: The error probability of module detection for public level (green lines) and of textured pattern detection for private level (blue line) depending on pattern density (from 10% to 90%).

Secondly, we vary the **pattern size** from  $3 \times 3$  pixels to  $12 \times 12$  pixels, but we fix the pattern density equal to approximately 40 – 42% and the 2LQR actual size equal to  $1.2 \times 1.2 \text{ cm}^2$ . The pattern detection results are illustrated in Fig. 7.11. Here, the first level is readable for patterns that have a size from  $4 \times 4$  pixels to  $12 \times 12$  pixels. The second level is readable from a  $5 \times 5$  pixel pattern size, using the error correction capacity.

Experimental results show that for the correct reading process of public and private levels:

- The pattern size should be from  $5 \times 5$  pixels to  $12 \times 12$  pixels (the larger pattern size can significantly increase the QR code printing surface).
- The pattern density can vary from 20% to 80%. But we have to note that the image contrast is too weak, when the pattern density is less than 40%. And when the pattern density is larger than 70%, the distance among the correlation values is quite weak.

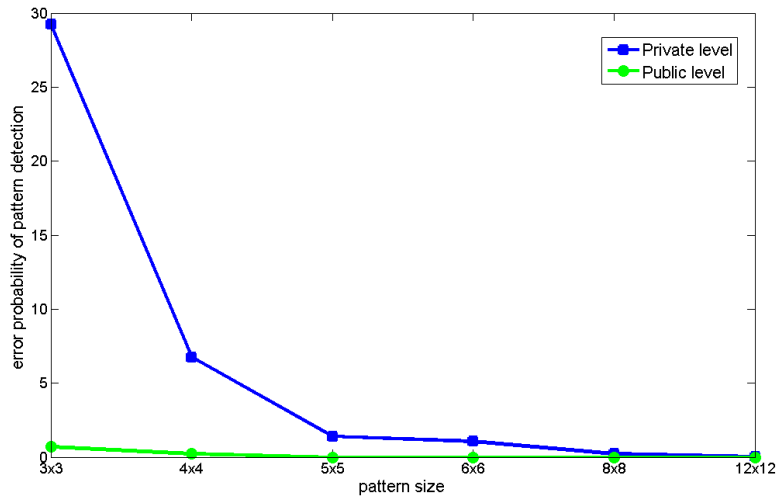


FIGURE 7.11: The error probability of modules detection for public level (green lines) and of textured pattern detection for private level (blue line) depending on pattern size (from  $3 \times 3$  pixels to  $12 \times 12$  pixels).

Code name	Storage capacity ( <i>bits/inch<sup>2</sup></i> )			Color printing	Copy sensitivity
	public	private	total		
HCC2D code [119]	15,048	-	15,048	Yes	No
Multilevel 2D barcode [151]	11,224	-	11,224	No	No
Graphical code for authentication <sup>1</sup> [106]	NI	-	NI	No	Yes
QR code with hidden message <sup>2</sup> [20]	7,548	3,102	10,650	No	No
Proposed 2LQR code <sup>2</sup>	7,548	6,386	13,934	No	Yes

TABLE 7.10: Rich graphical barcode comparison.

- The alphabet dimension  $q$  can be increased up to  $q = 8$ . The larger alphabet dimensions could disrupt the textured pattern detection due to the P&S impact.

To conclude, the proposed 2LQR code evaluation, we compare it with several existing rich graphical barcodes in Table 7.10. We note that only two (including the proposed 2LQR) of the mentioned graphical codes are sensitive to copying process, as well as this, only two have the private storage level capability. The maximum storage capacity of QR code with hidden message [20] is equal to 9,720 bits using QR code V40. If we use an 8-ary Reed-Solomon error correction code in a QR code V40, we can increase the hidden message length up to 20,000 bits.

1. The possibility to store a message is mentioned, but the length of message is never indicated (NI).  
 2. Storage capacity calculated for QR code V40 fixing the barcode size to  $3.1329 \text{ inch}^2$ .

## 7.4 Two level QR code for document authentication

The textured patterns illustrated in previous section are tolerant to P&S process due to high  $\varepsilon$  value ( $\varepsilon = 0.60$ ). The proposed method is built to allow the choice of textured patterns that can be more sensitive to P&S process, see Section 5.2. Therefore, in this section we introduce other application of 2LQR codes: the 2LQR codes used for printed document authentication.

Due to the specific characteristics of the textured patterns used, the original 2LQR code can be distinguished from one of its copies to ensure document authentication. This functionality has been performed due to the impact of the P&S process, that can be considered as physically unclonable because of both the deficiencies of the physical process and the stochastic nature of the matter [106].

### 7.4.1 Printed document authentication system

As was mentioned in Section 4.6.2, we can differ two types of printed document authentication: document content authentication and document support authentication. Let explain these difference using passport authentication. The document content authentication allows us to check the authenticity of data written in the passport (first and last name, date of births, nationality of passport owner). However, the passport content can be authentic (a person with such identification data exists), but the passport could be made by falsifiers. In this case, we need to use the document support authentication. The proposed 2LQR code can be used in both applications.

When we suppose to use the *2LQR code in document content authentication*, its storage capacity is highlighted and the following systems is provided. The valuable document generation center computes document own hash and stores it into the private level of a generated 2LQR code. An overview of considered authentication system is presented in Fig. 7.12.

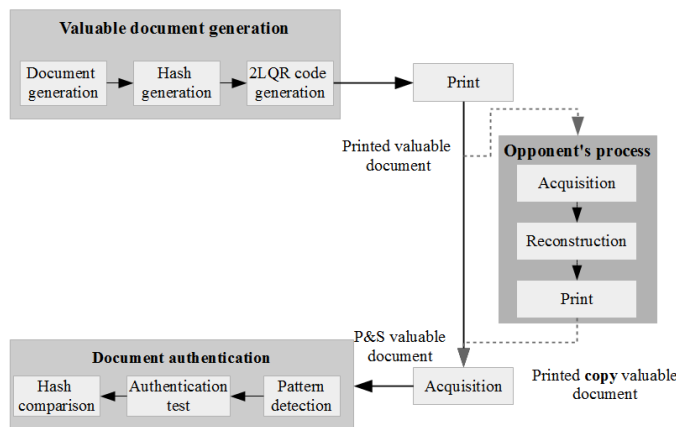


FIGURE 7.12: Considered authentication system for valuable documents.

After generation, the valuable protected document is printed using authorized printer. Further, during legitimate document verification, the printed document is scanned using authorized device. The P&S image of 2LQR code is used for pattern detection. If all detected patterns have positive notes during the authentication test, then the hash function is restored from the private level and is compared with the calculated document hash. Nevertheless, this thesis does not focus on document content authentication.

*Remark:* Sometimes, we need to protect only particular parts of document. In this case, the document tamper proofing scenario has been proposed [150] for local document content authentication. The document is divided into parts and the local hash is computed for each part. Then, these hashes can be stored in 2D barcodes.

When the *document support authentication using 2LQR code* is needed, the valuable document generation center stores the secret information into the private level of a generated 2LQR code. Then, during reading, the authority center can produce the authentication test, before reading of private storage level.

#### 7.4.2 Generation of authenticating 2LQR code

Unlike the secret message sharing scenario, in *document authentication*, we have to check the authenticity of the textured patterns used. Therefore, we have to compare the textured patterns used in 2LQR code with the original patterns. That is why, we do not need to change the black modules in the position tags of standard QR code. The comparison of 2LQR code used for document authentication with standard QR code is illustrated in Fig. 7.13.

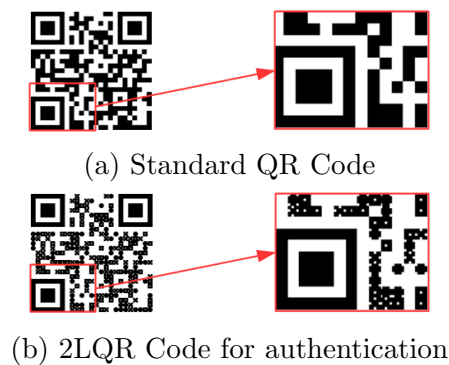


FIGURE 7.13: A comparison of a) Standard QR code, with b) Proposed 2LQR code for authentication scenario. The position patterns are black in authenticating 2LQR code.

The generation process of 2LQR code used for document authentication is the same as generation process presented in Section 7.2.1. However, the textured patterns must be chosen more carefully. Experimentally it is shown, that for document authentication, the distance threshold  $\varepsilon$  must be smaller:

$$\varepsilon < 0.40. \quad (7.6)$$



This condition ensures the textured patterns high sensitivity to P&S and copying processes.

### 7.4.3 Reading process of authenticating 2LQR code

The reading process of authenticating 2LQR code is almost the same as presented in Section 7.2.4. However, since we have not replaced the black modules in position tags by textured patterns (see Fig. 7.13.b), we compare the P&S patterns with the original patterns. That is why the characterization patterns are replaced by original images from equation (5.8):  $CP_l = P_l, l = 1, \dots, q$ .

All steps of 2LQR code reading process are illustrated in Fig. 7.14. Analogically to 2LQR code used for private message sharing, the image pre-processing step and the standard QR code reading process are applied in the beginning, in order to retrieve the public message  $M_{pub}$  and construct a black class  $BC$ . The textured modules from the  $BC$  class are used for the private message  $M_{priv}$  extraction and authentication test.

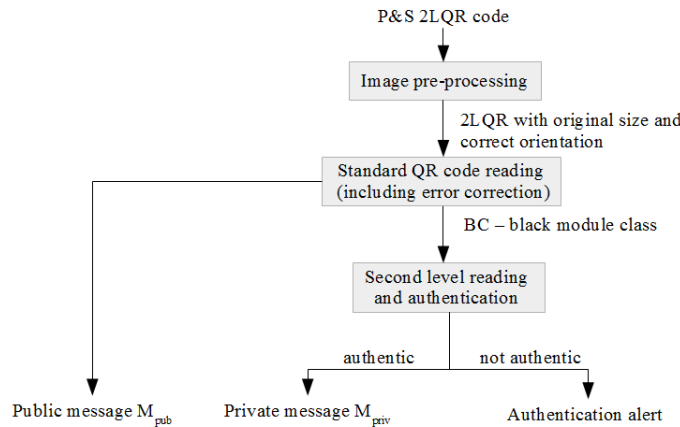


FIGURE 7.14: Overview of authenticating 2LQR code reading process.

The proposed pattern recognition and authentication tests are illustrated in the flowchart Fig. 7.15. As an input we have the textured modules  $BP_i, i = 1, \dots, N_{code} \times n$  from black module class  $BC$ . For each textured module  $BP_i$ , we calculate the correlation values with original textured modules  $P_l, l = 1, \dots, q$ :  $cor_l^i = cor(BP_i, P_l), i = 1, \dots, N_{code} \times n$ . The maximum correlation value  $cor_l = \max_{j=1, \dots, q} \{cor_j^i\}$  defines the bit value:  $c'_i = l$ . In the end of this process, we obtain the scrambled codeword  $C'$ . Before unscrambling and decoding operations, the authentication test has to be performed. The proposed authentication test is described in Section 7.4.4. If the authentication test approves the 2LQR code authenticity, the unscrambling with key  $K$  and ECC decoding process is applied for private message  $M_{priv}$  extraction. The parity-check bits of ECC are used for error detection and correction.

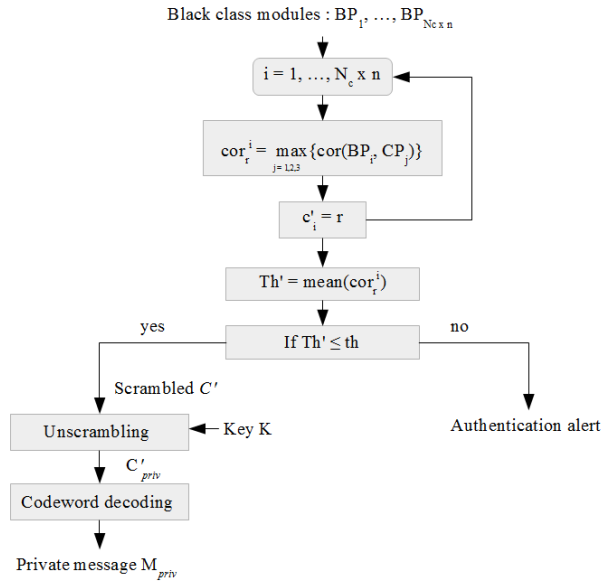


FIGURE 7.15: The pattern recognition method scheme by using characterization patterns  $CP_i$  for authenticating 2LQR code.

### 7.4.4 Authentication process

For authentication of printed document, we have to check the authenticity of textured patterns used in 2LQR code. That is why, during the recognition process, the P&S textured patterns are compared with the original numerical patterns using the Pearson correlation (5.1). The maximum correlation values  $cor_i^i$  is also used for authentication test.

#### Authentication test

The authentication threshold  $Th$  is evaluated on an experimental step. The document authentication is based on sensitivity of textured patterns to P&S process.

A copy attack is a reproduction chain that implies two successive P&S processes. The modifications, that induces on the document, are cumulative. The authentication step aims at evaluating the pattern degradation in order to acknowledge for successive P&S functions applied to the original pattern. We propose to use the Pearson correlation coefficients for measuring the degradation. The Fig. 7.16 shows this evaluation process.

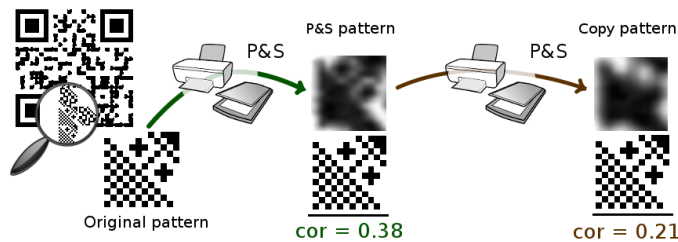


FIGURE 7.16: An example of textured pattern changes during copying process.

The authentication test is performed by comparison of mean value of  $cor_l^i, i = 1, \dots, N_{code} \times n$  with pre-calculated threshold value  $Th$ . The document is authentic, if this mean value is bigger than the threshold  $Th$ . The authentication test is presented in Algorithm 1.

---

**Algorithm 1** Authentication test
 

---

**Require:**  $cor_l^i, i = 1, \dots, N_{code} \times n$

- 1: **if**  $mean(cor_l^1, \dots, cor_l^{N_{code} \times n}) \geq Th$  **then**
- 2: Document is authentic
- 3: **else**
- 4: Authentication alert
- 5: **end if**

---

## 7.5 Experiments with authenticating 2LQR code

In this section, we present the printed document authentication scenario, describe the database used and show the authentication and module recognition results.

The *application scenario* is as follow. The authority center creates the valuable document. The public and private information is stored in the first and second levels of the 2LQR code, respectively. In the end, the generated 2LQR code is inserted into the document, and the document is printed using desktop printing device.

In the verification step, the valuable document is scanned using desktop scanner device, the first level reading is performed, the verification of 2LQR code authenticity is then applied. If the 2LQR code is authentic, the private information can be easily restored.

In these experiments, a version V3 QR code in Low error correction level is used. This version has  $29 \times 29$  module size and can store 440 bits of a message. The private message is encoded using the ternary Golay code [11, 6, 5], where each of  $N_{code}$  codewords has  $n = 11$  digit length, with  $k = 6$  informative digits. With this chosen ECC we can store 174 ternary digits on the private level of the 2LQR code, that corresponds to nearly 275 message bits.

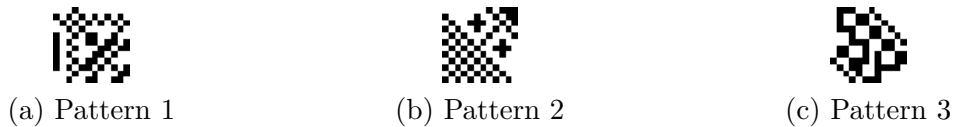


FIGURE 7.17: The three textured patterns used for private level generation of authenticating 2LQR code: a) Pattern 1, b) Pattern 2, c) Pattern 3.

For experiments with authenticating 2LQR code, we chose three textured patterns (i.e.  $q = 3$ ) illustrated in Fig. 7.17, that are sensitive to P&S process. The original and the P&S degraded versions of these patterns satisfy conditions (7.5). We set  $\varepsilon = 0.25$ . The correlation values as well as the minimal distance ( $\min\{cor(P_l, S_l) - \max_{l \neq l'}(cor(P_l, S_{l'}))\}$ ) are illustrated in Table 7.11. We note that the diagonal correlation values  $cor(P_l, S_l), l = 1, 2, 3$ , which are equal to  $cor(P_1, S_1) = 0.4432, cor(P_2, S_2) =$

0.2754 and  $cor(P_3, S_3) = 0.4752$ , are the maximum in line and in the column. Then, we calculate the distances among  $cor(P_l, S_l)$  and  $cor(P_l, S_{l'})$ ,  $l, l' \in \{1, 2, 3\}$ ,  $l' \neq l$ , i.e.  $cor(P_l, S_l) - cor(P_l, S_{l'})$  and  $cor(P_l, S_l) - cor(P_{l'}, S_l)$ ,  $l, l' \in \{1, 2, 3\}$ ,  $l' \neq l$ . We notice that all distances are bigger, then threshold  $\varepsilon = 0.25$ . That is why, the condition (7.5) is respected. Therefore, we can use these patterns for second level generation.

	$P_1$	$P_2$	$P_3$
$S_1$	<b>0.4432</b>	-0.0072	0.1016
$S_2$	-0.0054	<b>0.2754</b>	-0.0076
$S_3$	0.0293	-0.0860	<b>0.4752</b>
minimal distance	0.3416	0.2830	0.3737

TABLE 7.11: The correlation values for the original patterns  $P_1, P_2, P_3$  and its P&S degraded versions  $S_1, S_2, S_3$ , the minimal distances among diagonal correlation values and all other correlation values in the same line and the same column  $\min\{cor(P_l, S_l) - \max_{l' \neq l}(cor(P_l, S_{l'}))\}$ .

### 7.5.1 Pattern degradation for authentication

As we mentioned in Chapter 3, any P&S process adds specific changes in each image. These modifications can be provided by ink dispersion (in the paper or onto the device output), non homogeneous luminosity conditions during the scanning process, inherent re-sampling of the P&S process or variable speed during the acquisition process [9]. The modifications which the textured patterns go through are discussed in Section 5.2. Due to the P&S impact, it is difficult to model the P&S degraded versions of proposed textured patterns.

We decided to measure the difference between the original printed document (after one P&S process) and a copy of the document (after two P&S processes). A numeric original pattern is called an original pattern, a pattern after the P&S process is called a P&S pattern, and a pattern after two P&S processes is called a copy pattern, see Fig. 7.16.

For this experiment, we generate a 2LQR code with binary alphabet  $q = 2$ , i.e. we use only two textured patterns  $P_1, P_2$  for private level generation. This 2LQR code was printed and scanned once (we obtain a P&S document) and then, this P&S document was printed and scanned a second time (we obtained a copy document). Therefore, we had three documents: original document, P&S document and copy of the P&S document. We compared the original patterns with the P&S patterns, and then, the original patterns with the copy patterns by using distances between correlation values  $\varepsilon$ :

$$\varepsilon_o = |cor(P_1, S_i) - cor(P_2, S_i)|,$$

$$\varepsilon_c = |cor(P_1, S_i^c) - cor(P_2, S_i^c)|,$$

where  $S_i$  is the P&S version of  $P_i$ ,  $S_i^c$  is the copy version of  $P_i$ ,  $cor(P_1, S_i)$  (resp.  $cor(P_2, S_i)$ ) is the Pearson correlation value between the original  $P_1$  ( $P_2$ ) textured patterns and the

P&S version of textured pattern,  $cor(P_1, S_i^c)$  (resp.  $cor(P_2, S_i^c)$ ) is the Pearson correlation value between the original  $P_1$  ( $P_2$ ) textured patterns and the copy version of the textured pattern,  $i = 1, 2$ . We use the differences in order to show the convergence of correlation values after two consecutive P&S processes. These correlation value differences are illustrated in Fig. 7.18. The blue squares represent the  $\varepsilon_o$  values calculated among the original patterns and the P&S patterns. The red dots represent the  $\varepsilon_c$  values calculated among the original patterns and the copy patterns. We can notice that the mean value of  $\varepsilon_o$  calculated for the P&S patterns (blue line) are twice as larger as the mean value of  $\varepsilon_c$  calculated for the copy patterns (red line). Therefore, we can conclude that the suggested textured patterns are sensitive to the copy process and can be used to distinguish the original printed document from its copy.

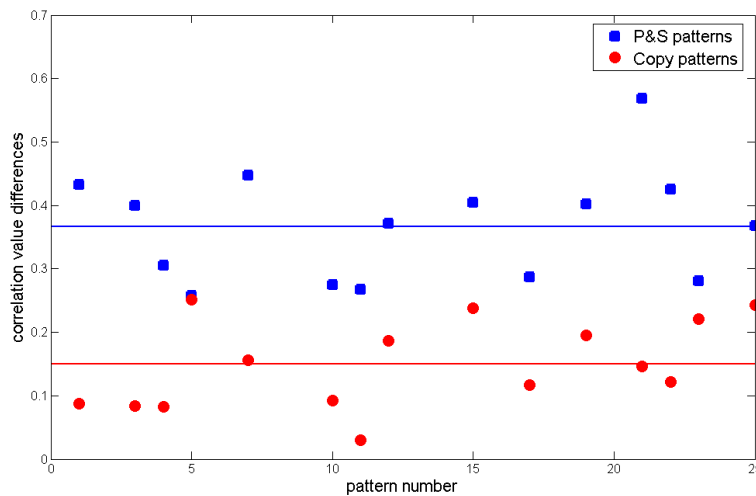


FIGURE 7.18: Degradation of textured patterns by the P&S process: the values  $\varepsilon_o$  are differences of correlation values with original patterns  $P_1, P_2$  and P&S patterns  $S_1, S_2$ , the values  $\varepsilon_c$  are differences of correlation values with original patterns  $P_1, P_2$  and copy patterns  $S_1^c, S_2^c$ . The blue (red) line is the mean value of differences  $\varepsilon_o$  ( $\varepsilon_c$ ).

### 7.5.2 Message extraction from authenticating 2LQR code

In this section, we aim to extract the message from printed and copied 2LQR codes. Therefore, we apply the reading process over printed and copied 2LQR codes.

In the *database used*, we have P&S 2LQR codes printed with the Brother HL-4150CDN printer and scanned with the Canon LIDE210 scanner. The copy of printed 2LQR code are made with the copy machine Toshiba e355 in standard mode and with maximal contrast (CM1 and CM2 respectively), the RICOH Aficio MP C2050 (CM3), the Toshiba e456 (CM4) and the Toshiba e256 (CM5). The printing, scanning and copying processes have been performed using 600 dpi resolution. Note that we want to highlight the effects of document copy process: we fix the production chain and test several copy machines. Of course, the production and control chain can be improved (for example

up to 1200 dpi) to increase the protection of the system. An example of the 2LQR code as well as an example of P&S and copy 2LQR codes are illustrated in Fig. 7.19.

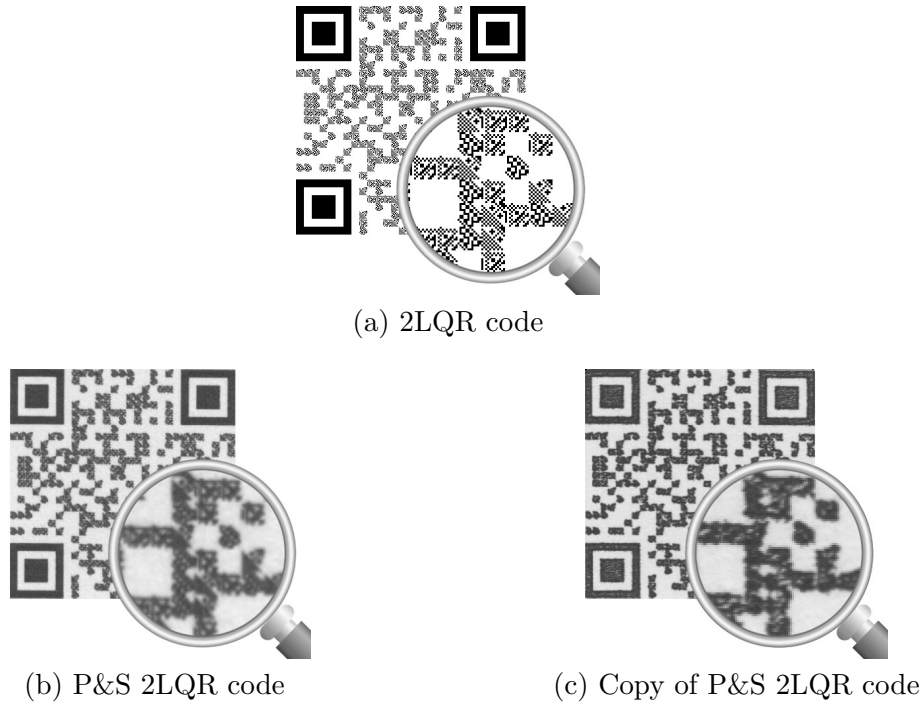


FIGURE 7.19: An example of a) 2LQR code, b) P&S 2LQR code and c) Copy of P&S 2LQR code.

In total we have 120 P&S 2LQR codes with different public and private messages, and 120 copy P&S 2LQR codes using each copy machine (i.e. 600 copy samples in a total). This choice of database allows to have diversity of pattern combinations.

		Error probability of	
		pattern detection	digit decoding after ECC
Printed 2LQR code		1.03%	0.30%
Copied 2LQR code using	CM1	20.30%	21.32%
	CM2	26.29%	26.24%
	CM3	23.92%	22.48%
	CM4	20.12%	15.77%
	CM5	20.76%	17.01%
	Mean	22.28%	20.56%

TABLE 7.12: Error probability of pattern detection and error probability of message decoding after ternary Golay error correction algorithm.

Table 7.12 gathers all obtained results: the first line shows the results obtained to printed 2LQR codes, the lines from second to sixth represent the results obtained to copies using different copy machines and, the last line illustrates the mean values of lines from 2 to 6 (copies results). For each printed 2LQR code and each copy, we apply the proposed module recognition method. In the first column of Table 7.12, we note, that the probability of a wrong pattern detection equals to 1.03% for printed 2LQR codes (original).

Nevertheless, the mean probability of a wrong pattern detection of copies using different copy machines equals to 22.28%.

After the unscrambling using key  $K$ , error detection and correction algorithm, the private message  $M_{priv}$  is retrieved. The error probabilities of incorrect digit decoding are presented in the second column of Table 7.12. We can see that the private message could not always be decoded in the copy samples, as the pattern detection operation is not performed successfully.

### 7.5.3 Authentication test

In the end, we evaluate the *authentication test*. In Fig. 7.20, we can see the mean correlation values for printed 2LQR codes (blue line) and for copy 2LQR codes using different copy machines (orange, yellow, green, vinous and light blue lines). This figure shows that the threshold between original and copy 2LQR code can be calculated experimentally, and then this threshold can be used for authentication test performance.

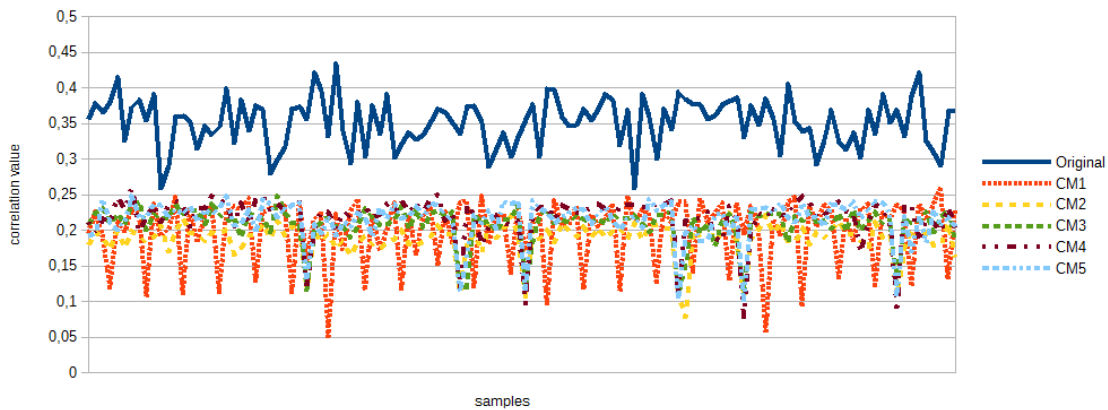


FIGURE 7.20: The mean correlation values for P&S 2LQR codes (blue line) and for copy P&S 2LQR codes using different printers and scanners (orange, yellow, green, vinous and light blue lines).

We evaluate the authentication test with authentication threshold  $Th = 0.3$ . The Table 7.13 shows the probability of code authentication. All copy codes and less than 2% of original 2LQR codes do not pass the authentication test. Our goal is to detect all unauthorized duplication of documents (false positive rate has to be zero) even if false negative rate is non-zero.

	Original	CM1	CM2	CM3	CM4	CM5
Positive authentication test ( $Th = 0.3$ )	98.33%	0.00%	0.00%	0.00%	0.00%	0.00%

TABLE 7.13: Positive authentication test results using authentication threshold  $Th = 0.3$

In reality, all documents, which failed the authentication test, undergo a deeper analysis (high time consuming process), that will distinguish the false negative from real fakes. Therefore, we suppose that the 2% false positive error is acceptable, when the false negative error is equal to 0%.

The pattern detection results (presented in Table 7.12) show that when the authentication test failed, we cannot extract the correct private message in most cases and cannot verify the authenticity of document content. Contrary, after the positive answer of authentication test, both level information can be extracted.

#### 7.5.4 Simple attacks

We perform two simple attacks to verify the robustness of our 2LQR code. The first attack consists of two consecutive P&S operations (named "2P&S attack" in Table 7.14). This attack can be performed by naive attacker. The second attack consists also of two consecutive P&S operations, but in this case the image histogram equalization of printed and scanned 2LQR code is performed before re-printing (named "Equalization histogram attack" in Table 7.14). A scheme of this attack is illustrated in Fig. 7.21. This attack is also simple, but requests several knowledge of image processing and experience in graphics tools.

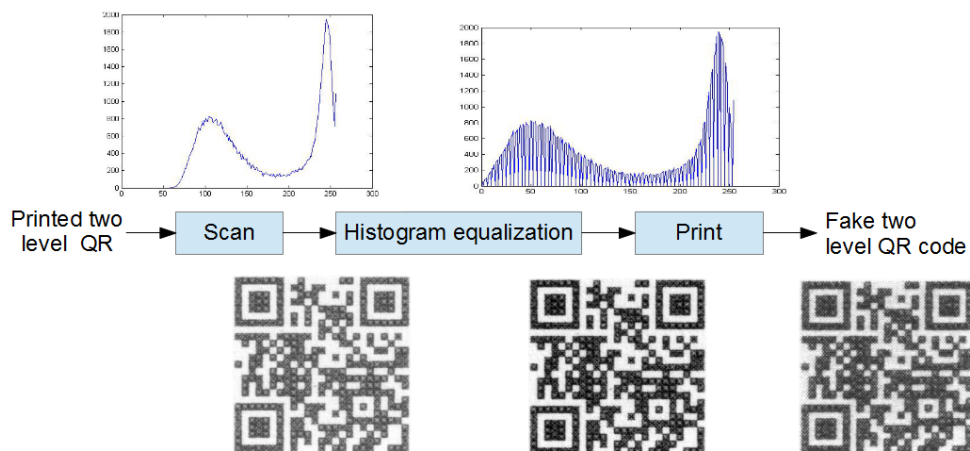


FIGURE 7.21: Histogram equalization attack of 2LQR code.

In our database, we collected the images printed and scanned with two Printer-Scanner (PS) pairs: a HP LaserJet Pro CM1415 printer-scanner (PS1) and a HP LaserJet Pro CM1415 printer with a Canon LIDE210 scanner (PS2). We have applied the proposed pattern detection method to retrieve the inserted message. Table 7.14 shows the pattern detection results for both attacks, and both printer-scanner pairs. The error probability of pattern detection is nearly 20% for both attacks. Thus, the private message can not be decoded correctly.



	Error probability of pattern detection	
	PS1	PS2
2P&S attack	22.12%	39.23%
Equalization histogram attack	19.76%	13.57%

TABLE 7.14: Error probability of pattern detection after simple attacks.

The results of Section 7.5.3 and Section 7.5.4 show that the proposed 2LQR code resists simple P&S attacks as well as duplication attacks using copy machines.

## 7.6 Message sharing vs authentication scenario

In this section we conclude by highlighting the differences between private message sharing scenario and authentication scenario. For this, we fix the alphabet dimension  $q = 3$ , the QR code version V2, textured pattern size  $12 \times 12$  pixels. And we vary the threshold  $\varepsilon$  from criteria (7.5). In Table 7.15, the error probabilities of pattern detection and digit decoding are presented for  $\varepsilon = 0.60$  and  $\varepsilon = 0.25$ . The results of this table show that the choice of threshold  $\varepsilon$  change the application scenario for proposed 2LQR code. So, if we want to embed a huge number of information, to be robust to copy process, and we do not need to assure the 2LQR code authenticity, we use the threshold  $\varepsilon \in [0.40, 0.70]$ . Otherwise, if we do not want only to embed a huge number of information, but also verify the authenticity of 2LQR code, we should chose  $\varepsilon \in (0.20, 0.40)$ . In the second scenario, we cannot be robust to copy process, as only the printed 2LQR code (after one P&S operation) is considered as authentic.

		Error probability of			
		pattern detection		digit decoding after ECC	
		$\varepsilon = 0.60$	$\varepsilon = 0.25$	$\varepsilon = 0.60$	$\varepsilon = 0.25$
Printed 2LQR code		0.00%	1.03%	0.00%	0.30%
Copy 2LQR code using	CM1	0.13%	20.30%	0.29%	21.32%
	CM2	1.07%	26.29%	0.73%	26.24%
	CM3	0.21%	23.92%	0.05%	22.48%
	CM4	0.15%	20.12%	0.11%	15.77%
	CM5	0.22%	20.76%	0.22%	17.01%

TABLE 7.15: Error probability of pattern detection and error probability of message decoding after ternary Golay error correction algorithm for different textured patterns, depending on  $\varepsilon$  value.

## 7.7 Conclusions

In this chapter a new two level barcode is introduced. This barcode stores the information in public and private levels. The public information is readable by standard barcode reader, the private information can be accessible only for authorized users who possess the specific reading application.

Two application scenarios are presented: private message sharing scenario using blind pattern detection, and printed document authentication scenario, where the authentication test must succeed before decoding private message.

The additional storage level as well as the authentication capacity are offered by specific textured patterns, that are distinguished one from another after P&S process and can be more/less sensitive to coping process (depending on textured pattern choice).

The contribution of 2LQR code for message sharing is published in international conference [142]. Additionally, the extended version is accepted for publication in the international journal "Transaction on Information Forensics and Security". The paper that introduces the 2LQR code for document authentication is submitted to the International Conference on Acoustics, Speech and Signal Processing (ICASSP 2016).



## Chapter 8

# Experimental study of P&S impact

### 8.1 Introduction

The degradation of information due to Print-and-Scan (P&S) processes is a major issue in digital forensics and printed document as well as in image authentication. This degradation is usually considered as being a stochastic process that can be modeled by an additive or a multiplicative normal or lognormal distribution [10, 164]. These models include a random noise due to ink dispersion on the paper during the printing as well as the illumination conditions during the scanning process. It is generally acknowledged that these two degradations cannot be separated.

In this chapter, we aim at experimentally determine the nature of the printer and the scanner noise, as well as identify the distribution of white and black pixels after the P&S process.

One of the main goals of this paper is to propose a way to characterize the nature of a stochastic process in image processing and particularly answer the following questions:

1. Does this process follow a given statistical distribution (e.g. normal or Laplace)?
2. Can we consider the noise as being additive?
3. Can we consider the noise as being stationary?
4. Can we consider the noise as being ergodic?
5. Can we consider the noise as being white?

To answer those questions, we propose a series of statistical tests. We experiment those tests on a large image database that contains black-and-white images of our proposed 2LQR code (see Chapter 7) that have been collected after numerous printing and scanning operations. We use these images since they have a very high contrasted structure made of black and white patterns. All statistical tests presented in this chapter are

performed with real data.

In Section 8.2, we introduce some statistical definitions and explain the statistical tests we use. The proposed methodologies to verify process stationarity and ergodicity are explained in Section 8.3. The experiments and their outcomes are presented in Section 8.4. The numerous experimental results we obtain on color changes after P&S process are discussed in Section 8.5. Finally, we conclude in Section 8.6.

## 8.2 Statistical definitions and tests

### 8.2.1 Random process characteristics

A *random variable*,  $X(s)$ , is a single-valued real function that assigns a real number, called the value of  $X(s)$ , to each sample point  $s \in S$  [61].

A *random process* can be defined as a family of random variables  $\{X(t, s) | t \in T, s \in S\}$  defined over a given probability space and indexed by the time parameter  $t$  [61]. The  $X(t, s)$  is a collection of time functions, one for each sample point  $s$  (see Fig. 8.1).

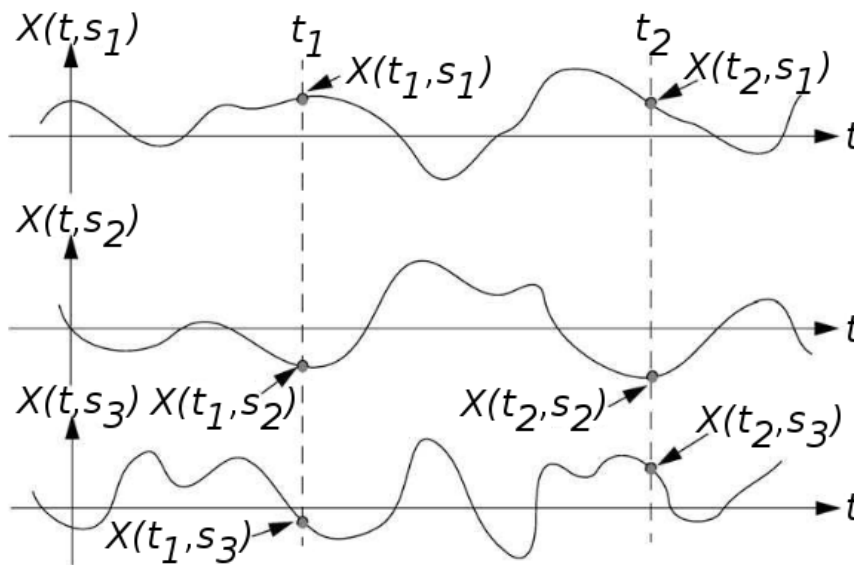


FIGURE 8.1: A sample random process [61].

A parameter is called *statistical*, if it is calculated using  $X(t, s)$  with a fixed value of  $t$ . A parameter is called *spatial*, if it is calculated using  $X(t, s)$  with a fixed value of  $s$ .

**Stationary process.** A random process is called a *strict sense stationary process* if its Cumulative Distribution Function (CDF)  $F_X$  is invariant to a shift in the time origin [61]. That means that  $X(t, s)$  is strict sense stationary if its CDF is identical to the CDF of  $X(t + \epsilon, s)$  for any arbitrary shift  $\epsilon$ :

$$F_X(X(t, s)) = F_X(X(t + \epsilon, s)). \quad (8.1)$$

When the CDF is differentiable, the equivalent condition for strict sense stationarity is that the Probability Distribution Function (PDF)  $f_X$  is invariant to a shift  $\epsilon$  in the time origin:

$$f_X(X(t, s)) = f_X(X(t + \epsilon, s)). \quad (8.2)$$

In practice, we often work only with the mean and the autocovariance functions of a random process. A random process in which the mean and autocovariance function does not depend on absolute time is called a *wide-sense stationary* process. Thus, for a wide-sense stationary process  $X(t)$ , we have:

$$E[X(t, s)] = \mu_X, \quad \text{constant}, \quad (8.3)$$

$$E(X(t, s), X(t + \tau, s)) = R_{XX}(\tau), \forall t \in T. \quad (8.4)$$

**Ergodic process.** The time average of random process  $X(t, s)$  is calculated at a fixed sample point  $s$  (i.e. calculated over  $X(t_1, s), \dots, X(t_n, s)$ ). Considering a random process  $X(t, s)$  whose observed sample is  $x(t)$ , the time average of the function  $x(t)$  is defined by:

$$\bar{x} = \lim_{\tau \rightarrow \infty} \frac{1}{2\tau} \int_{-\tau}^{\tau} x(t) dt. \quad (8.5)$$

A stationary random process  $X(t, s)$  is said to be *ergodic* if every member of the set exhibits the same statistical behavior as the set. This implies that it is possible to determine the statistical behavior of the set by examining only one typical sample function [61]. An ergodic process can be represented by only one stochastic process realization. As for stationarity, ergodicity is often restricted to its two first orders.

A random process  $X(t, s)$  is defined as being *first order ergodic* (or ergodic in the mean) if time mean function has the same distribution as statistical mean function:

$$\bar{x} = \mu_X. \quad (8.6)$$

A random process  $X(t, s)$  is defined to be *second order ergodic* if time autocovariance function has the same distribution as statistical covariance function:

$$R_{XX} = Cov_{XX}. \quad (8.7)$$

**White-noise process.** The random process  $X(t, s)$  is a white noise if its autocovariance function is close to a Dirac impulse:

$$R_{XX}(\tau) = \frac{\sigma_X^2}{2} \delta(\tau), \quad (8.8)$$

where  $\sigma_X^2$  is the variance of the random process  $X(t, s)$  and  $\delta(\tau)$  is the impulse function. This functions is illustrated in Fig. 8.2.

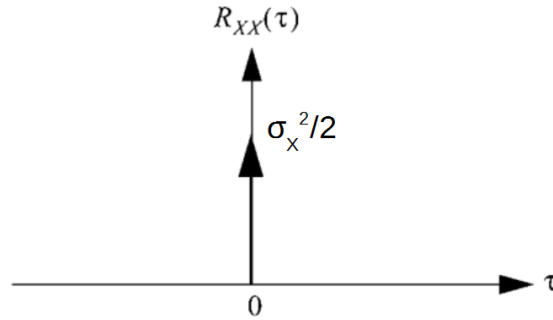


FIGURE 8.2: Autocovariance function of white noise random process.

The Fourier transform (if it exists) of the autocovariance function of a stationary in the wide sense random process  $X(t, s)$  is called its power spectral density. Since  $R_{XX}(\tau) = \frac{\sigma_x^2}{2} \delta(\tau)$ , then  $S_{XX}(\tau) = TF\{R_{XX}(\tau)\} = \frac{\sigma_x^2}{2}$  does not depend on the frequency. This is why it is called white noise (by analogy with the spectral property of the white light).

### 8.2.2 Some classical statistical distributions

Different classical statistical distributions are used in this chapter. Here we give their Probability Density Function (PDF) and Cumulative Distribution Function (CDF).

A **normal (or Gaussian) distribution** is a distribution which is completely defined by its first two moments:

- $\mu$ , the mean (or expectation) of the distribution,
- $\sigma$ , the standard deviation of the distribution.

The normal distribution has the following PDF:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \quad (8.9)$$

The CDF of the normal distribution is given using the following formula:

$$F(x) = \frac{1}{2} \left[ 1 + \operatorname{erf} \left( \frac{x-\mu}{\sigma\sqrt{2}} \right) \right], \quad (8.10)$$

where  $\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt$ .

A **log-normal distribution** is a continuous probability distribution of a random variable whose logarithm is normally distributed. Therefore, if the random variable  $X$  is log-normally distributed, then  $Y = \ln(X)$  has a normal distribution. Its parameters are:

- $\mu$ , the mean of the variable's natural logarithm, called the location parameter on a logarithmic scale,
- $\sigma$ , the standard deviation of the variable's natural logarithm, called the scale parameter on a logarithmic scale.

The log-normal distribution has the following PDF:

$$f(x) = \frac{1}{x\sigma\sqrt{2\pi}e^{-\frac{(\ln x - \mu)^2}{2\sigma^2}}} \quad (8.11)$$

The CDF of the log-normal distribution is given using the following formula:

$$F(x) = \frac{1}{2} + \frac{1}{2} \operatorname{erf} \left[ \frac{\ln x - \mu}{\sqrt{2}\sigma} \right] \quad (8.12)$$

A **Laplace distribution** is a continuous probability distribution. It is called the double exponential distribution because it can be thought of as two exponential distributions (with an additional location parameter) spliced together back-to-back. The parameters of this distribution are as follows:

- $\mu$ , the mean value, also called its location parameter,
- $b$ , referred to as the diversity, also called its scale parameter.

The variance of a Laplace distribution is given using the following formula:

$$\operatorname{Var}(X) = 2b^2. \quad (8.13)$$

The Laplace distribution has the following PDF:

$$f(x) = \frac{1}{2b} e^{-\frac{|x-\mu|}{b}}. \quad (8.14)$$

The CDF of the Laplace distribution is given using the following formula:

$$F(x) = \begin{cases} \frac{1}{2} e^{-\frac{x-\mu}{b}}, & \text{if } x < \mu, \\ 1 - \frac{1}{2} e^{-\frac{-(x-\mu)}{b}}, & \text{if } x \geq \mu \end{cases} \quad (8.15)$$

### 8.2.3 $\chi^2$ goodness of fit test

The  $\chi^2$  goodness of fit test is used to identify whether sample data are consistent with a given distribution or not [161]. This test can be used when the sample data are categorical and when the number of observations in each variable level is at least 5.

The null hypothesis of this test is: *the data are consistent with the specified distribution.*

The alternative hypothesis is: *the data are not consistent with the specified distribution.*

Generally, only one reject of null hypothesis is enough to ensure, at a given significant level, that the data are not consistent with the distribution. However, several tests are needed to be confident that the null hypothesis can be accepted at the same significance level. The significance level is chosen by the user. It is often equal to 0.01, 0.05, or 0.10. Let  $X$  be a discrete random variable, whose domain can be divided into  $k$  partition classes  $A_1, A_2, \dots, A_k$ . Let  $n_i, i = 1, \dots, k$  be the number of samples in each class  $i$ , with  $N = n_1 + n_2 + \dots + n_k$  being the total number of samples in  $A$ . Let  $p_i, i = 1, \dots, k$



be the probabilities of the class  $i$  based on the specified distribution. The  $\chi^2$  goodness of fit test considers the statistics  $D^2$  defined as follow:

$$D^2 = \sum_{i=1}^k \frac{(n_i - Np_i)^2}{Np_i}. \quad (8.16)$$

The degree of freedom of  $\chi^2$  goodness of fit test equals to  $k - 1$ . The null hypothesis can be accepted if:

$$D^2 \sim \chi^2. \quad (8.17)$$

The table of critical values for the  $\chi^2$  goodness of fit test presents threshold values  $\chi_{k-1,\alpha}^2$  for this test for different significance levels  $\alpha$  (also called p-value). The null hypothesis is rejected if:

$$D^2 > \chi_{k-1,\alpha}^2. \quad (8.18)$$

If the same  $A$  is used to estimate  $l$  parameters of the specified distribution, then the  $\chi_{k-1-l,\alpha}^2$  critical value has to be chosen.

#### 8.2.4 Kolmogorov-Smirnov test

The Kolmogorov-Smirnov (KS) test can be used to verify whether two probability distributions differ or not [161]. The null hypothesis of this test is: *the two distributions are close*. The alternative hypothesis is: *the two distributions are different*. Acceptance or rejection are also done at significance level chosen by user.

This test uses the maximal distance between the empirical distribution functions of both probability distributions. The empirical distribution function  $F_n$  for  $n$  observations  $x_i$  of  $X$  is defined as:

$$F_n(x) = \frac{1}{n} \sum_{i=1}^n I_{]-\infty, x]}(x_i), \quad (8.19)$$

where  $I_{]-\infty, x]}(x_i)$ , the indicator function of  $]-\infty, x]$ , equals to 1 if  $x_i \leq x$  and to 0 otherwise.

Let  $X_1$  and  $X_2$  be two discrete random variables. Let  $F_{1,n}$  and  $F_{2,n'}$  be the empirical distribution functions of random variables  $X_1$  and  $X_2$  respectively. The KS test considers the statistics  $D_{n,n'}$  defined as follow:

$$D_{n,n'} = \sup_x |F_{1,n}(x) - F_{2,n'}(x)|. \quad (8.20)$$

The null hypothesis is rejected at significant level  $\alpha$  if:

$$D_{n,n'} > c(\alpha) \sqrt{\frac{n+n'}{nn'}}, \quad (8.21)$$

where  $c(\alpha)$  is defined using the table of critical values for the Kolmogorov-Smirnov test [129]. For example,  $c(\alpha) = 1.36$ , if  $\alpha = 0.05$  or  $c(\alpha) = 1.22$ , if  $\alpha = 0.1$ .

### 8.2.5 Mann-Whitney test

The Mann-Whitney U-test (also called Wilcoxon rank-sum test) is a non-parametric test that is used to test whether two independent random variables have identical distributions or not [161]. The null hypothesis of this test is: *the two distributions are identical*. The alternative hypothesis is: *the two distributions are different*. As in the previous hypothesis tests, acceptance or rejection are done at a user chosen significance level.

This test is based on the idea that the information about the relationship between two random variables  $X_1$  and  $X_2$  can be obtained from  $n_1$  observations of the random variable  $X_1$  and  $n_2$  observations of the random variable  $X_2$  that are arranged together in increasing order.

The test aims at testing whether the two random variables  $X_1$  and  $X_2$  are mixed or not. Let us suppose that the random variable  $X_1$  has less observations, e.g.  $n_1 < n_2$ . The combined set of data is first arranged in ascending order with tied scores receiving a rank equal to the average position of those scores in the ordered sequence. Then, the ranks for the observations which came from  $X_1$  are summed up to provide the rank sums  $R_1$ . The Mann-Whitney U-test considers the statistics  $U$  defined as follow:

$$U = n_1 n_2 + \frac{n_1(n_1 + 1)}{2} - R_1. \quad (8.22)$$

The test consists of comparing the value  $U$  with the value given in the table of critical values for the Mann-Whitney U-test  $U_{n_1, n_2, \alpha}$ , where the critical values are provided for given values  $n_1$ ,  $n_2$  and  $\alpha$ . The null hypothesis is rejected if:

$$U > U_{n_1, n_2, \alpha}. \quad (8.23)$$

## 8.3 Proposed methodologies

In this section, we present how to use the proposed tests to verify the random process stationarity and ergodicity of first order and its stationarity of second order. Let us consider the discrete random process  $X(t, s)$  defined in finite sequences  $t = (t_1, \dots, t_m)$  and  $s = (s_1, \dots, s_n)$ .

### 8.3.1 Stationarity of first order test

As it was mentioned before, the process is stationary of first order if its statistical average  $E[X(t, s)]$  is constant over the temporal separation  $t$ , i.e.  $E[X(t, s)] = \mu_X$ .

To verify the first order stationarity of random process  $X(t, s)$ , we calculate its statistical mean values for each time  $t_i$ , i.e. we have a set of  $m$  mean values. In order to show the constancy of statistical mean, we divide this set into two independent random subsets

and apply the Kolmogorov-Smirnov test. The null hypothesis of this test is formulated as: *the two independent random subsets of mean value set have the same distributions*. If the null hypothesis is rejected at significant level  $\alpha$  at least once, the random process  $X(t, s)$  is not stationary of first order. Otherwise, the random process  $X(t, s)$  can be considered as being stationary of first order.

### 8.3.2 Ergodicity of first order test

The process is ergodic of first order if the spatial mean equals the statistical mean. As the random process is stationary of first order, we have the set of  $m$  statistical mean values. Then we calculate the spatial mean value for each state  $s_j$  (thus we obtain a set of  $n$  spatial mean values).

To verify the first order ergodicity of random process  $X(t, s)$ , we perform a Mann-Whitney U-test. The null hypothesis is formulated as: *the spatial mean and statistical mean are identical*.

If the null hypothesis is rejected at significant level  $\alpha$ , the random process  $X(t, s)$  is not ergodic of first order. Otherwise, we declare that the random process  $X(t, s)$  can be considered as being ergodic of first order.

### 8.3.3 Stationarity of second order test

We want to verify whether the random process  $X(t, s)$  is stationary of second order or not. As it was mentioned before, the process is stationary of second order if the autocovariance function does not depend on absolute time, i.e.

$$E(X(t, s), X(t + \tau, s)) = R_{XX}(\tau), \forall t \in T.$$

To verify the second order stationarity, we calculate its autocovariance function at each time  $t$  (i.e. we have a set of  $m'$  autocovariance values). In order to show its constancy, we divide this set of autocovariance values into two independent random subsets and apply the Kolmogorov-Smirnov test. The null hypothesis of this test is formulated as: *the two independent random subsets of autocovariance set have the same distributions*. If the null hypothesis is rejected at significant level  $\alpha$  at least once, the random process  $X(t, s)$  is not stationary of second order, and therefore, it is neither wide-sense stationary nor wide-sense ergodic. Otherwise, we declare that the random process  $X(t, s)$  can be considered as being stationary of second order.

## 8.4 Experiments: noise after P&S impact

In this section, we present a series of experiments dedicated to the study of the nature of the noise added by a P&S process. First, we isolate the scanner noise and perform several statistical tests to study its nature and to investigate its characteristics. Then, we perform the same experiments to study the noise added by the whole P&S process.

### 8.4.1 Study of scanner noise

**Experimental setup.** This experiment aims at isolating the scanner noise. To achieve this isolation, we propose to print an image once and then scan it  $n$  times. Let  $I$  be the original image,  $P$  be the printed image of original image  $I$  and  $S_j$  ( $j = 1, \dots, n$ ) be the  $n$  scanned samples of printed image  $P$ . The scheme can be picked as:

$$I \rightarrow \mathbf{Print} \rightarrow P \rightarrow \mathbf{Scan} \rightarrow S_j. \quad (8.24)$$

Let us consider  $P$  as a function of  $I$  and  $\varepsilon_P$ ,  $S_j$  as a function of  $I$ ,  $\varepsilon_P$  and  $\varepsilon_S$ , i.e:

$$P = f(I, \varepsilon_P),$$

$$S_j = g(P, \varepsilon_{S_j}) = P \oplus \varepsilon_{S_j} = f(I, \varepsilon_P) \oplus \varepsilon_{S_j},$$

where  $f()$  is the function associated to the printing process,  $g()$  is the function associated to the scanning process and  $\varepsilon_P$  and  $\varepsilon_{S_j}$  are the noises introduced by the printer and the scanner respectively. The  $\oplus$  operator shows that the printed and scanned image  $S_j$  can be represented by a function of the printing process and the scanner noise (that also depends on the function of the printing process).

If the noise has a regularity, we can consider to calculate the difference among samples in order to study the noise nature. We propose thus to calculate the differences among each pair of  $n$  samples, i.e. subtract pairs of scanned images to provide samples of the scanner noise:

$$S_j - S_{j'} = \varepsilon_{S_j} - \varepsilon_{S_{j'}}, \quad (8.25)$$

The  $\varepsilon_{jj'} = \varepsilon_{S_j} - \varepsilon_{S_{j'}}$  can be used to characterize the stationarity and the ergodicity of the noise introduced by the scanning process. These characteristics, that are inherent to  $\varepsilon_{jj'}$ , are also inherent to  $\varepsilon_{S_j}$  and  $\varepsilon_{S_{j'}}$ . Therefore, we can characterize the scanner noise by characterizing the stationarity and ergodicity of  $\varepsilon_{jj'}$ .

Let us introduce the same experimental setup when using the terminology of random processes. After a P&S process each image can be considered as random variable  $X(t, s)$ . Therefore, the set of  $n$  P&S images can be considered as random process  $X(t, s)$ , where  $s = \{1, \dots, n\}$  is the dimension of our data set and  $t = \{1, \dots, m\} \times \{1, \dots, m\}$  is the set of pixels in an image  $I$ .

As shown before, the scanner noise function can be obtained by subtracting every two random variables  $X(t, s_j)$  and  $X(t, s_{j'})$ , in order to create a new random process  $Y(t, s)$  with  $s = \{1, \dots, n'\}$ :

$$Y(t, s) = X(t, s_j) - X(t, s_{j'}), j \neq j', s_j = s_{j'} = \{1, \dots, n'\}, \quad (8.26)$$

where  $n' = \frac{n(n-1)}{2}$ , as the number of subtracting pairs we have is equal to the number of 2 element sub-sets from  $N$  elements, e.g. is equal to  $\frac{N!}{(N-2)!} = N(N-1)$ . Then, removing the symmetric pairs (e.g.  $(j_1, j_2)$  and  $(j_2, j_1)$ ), we obtain  $N' = \frac{N(N-1)}{2}$ .

**Database description.** We print one 2LQR code once and then scan it  $n = 90$  times. Each P&S image has a  $300 \times 300$  pixel size, i.e.  $m = 300$ . The number of subtracting pairs we have is equal to  $n' = 4,005$ .

Let us visualize the distribution of  $Y(t, s)$  for any values of  $t$  and  $s$  (i.e. as if it was an ergodic process) in Fig. 8.3. In total here we visualize  $4,005 \times 90,000 = 360,450,000$  data observations. The mean value of this distribution is equal to  $\mu = -0.0187$ , its standard deviation is equal to  $\sigma = 10.2264$ .

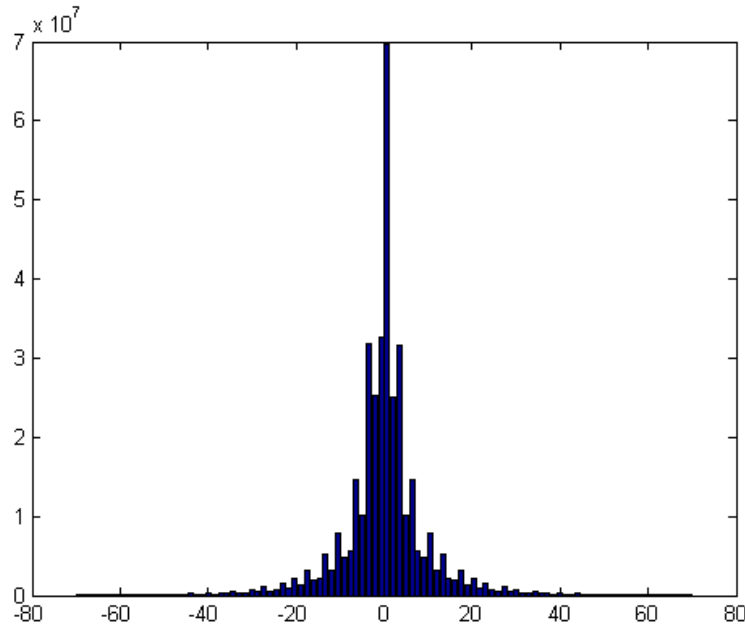


FIGURE 8.3: Scan noise distribution.

**Characterization of the  $Y(t, s)$  distribution.** In this experiment, we want to verify whether the  $Y(t, s)$  follow a given classical distribution (a normal or a Laplace) or not. In order to answer this question, we compare the distribution of the random process  $Y(t, s)$  with each specified distribution using the  $\chi^2$  goodness-of-fit test described in Section 8.2.3.

A reduced sample data set has been created by randomly selecting  $n' = 100$  vectors

of  $Y(t, s)$  (for lower computational time). We therefore work with a data subset of 9,000,000 samples. To counteract working with a subsample, we perform each test several times.

An algorithm of  $\chi^2$  goodness-of-fit test involving the estimation of mean and variance is presented in Algorithm 2. The result of this test is that both hypotheses are rejected at a significance level of 0.05. Thus our data-set has neither a normal nor a Laplace distribution.

---

**Algorithm 2** Chi-square goodness-of-fit test

---

**Require:** Data  $X = \{x_1, \dots, x_N\}$

- 1: Calculate the data histogram  $h$  with  $k$  bins
  - 2:  $n_i$  number of samples in bin  $i$
  - 3:  $\mu \leftarrow \text{mean}(X)$
  - 4:  $\sigma \leftarrow \text{std}(X)$
  - 5: **for**  $i = 1 : k$  **do**
  - 6:    $p_i \leftarrow \int f(x, \mu, \sigma) dx$
  - 7: **end for**
  - 8:  $D^2 \leftarrow \sum_{i=1}^k \left( \frac{(n_i - Np_i)^2}{Np_i} \right)$
  - 9: **if**  $D^2 < \chi_{k-3}^2$  **then**
  - 10:    $X$  has distribution, that corresponds to probability density function  $f$
  - 11: **end if**
- 

We computed two distances to see how far the empirical CDF based on the initial data is from the CDF of a normal or of a Laplace distribution. The comparison involves:

- the maximal distance  $d_{max}$  between the CDFs (the measure used in Kolmogorov-Smirnov test):

$$d_{max} = \sup_x |F_n(x) - F(x)|, \quad (8.27)$$

- the squared distance  $d^2$  between the CDFs (the measure used in Cramer-von Mises test):

$$d^2 = \sum_i (F_n(x_i) - F(x_i))^2 (x_i - x_{i-1}). \quad (8.28)$$

Table 8.1 presents the differences among the CDF based on our data-set and the CDF(s) of a normal and a Laplace distributions. We notice that these distances are rather stable under the changes of sample size.

Number of samples	Laplace distribution		Normal distribution	
	$d_{max}$	$d^2$	$d_{max}$	$d^2$
30,000	0.0933	0.0712	0.1482	0.2507
100,000	0.0936	0.0722	0.1483	0.2496
200,000	0.0922	0.0704	0.1474	0.2482
300,000	0.0934	0.0719	0.1490	0.2488
400,000	0.0925	0.0714	0.1484	0.2497

TABLE 8.1: Distances among CDFs of initial data (scanner noise) and estimated Laplace and normal distributions.

We illustrate these distances among CDFs in Fig. 8.4. We note that the CDF of the initial data (blue line) is closer to the CDF of the Laplace distribution (green line) than the CDF of the normal distribution (red line). Therefore the scanner noise can be said to be closer to a Laplace distribution than to a normal distribution.

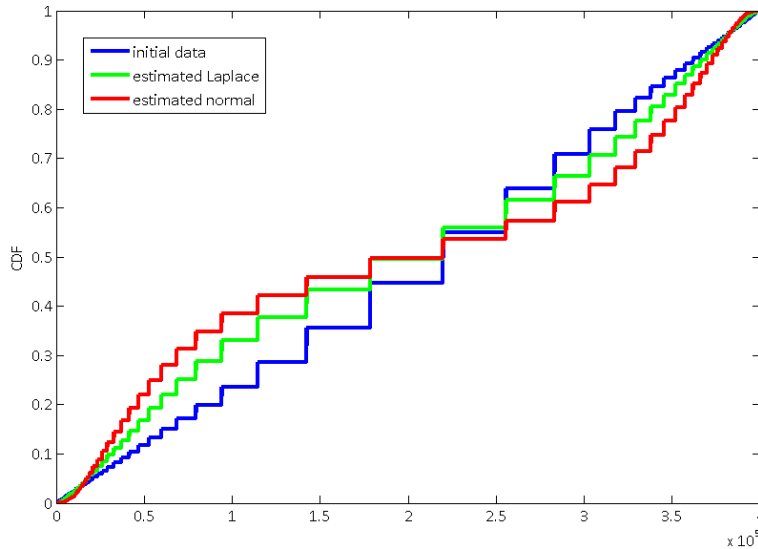


FIGURE 8.4: CDF of scanner noise (blue line), estimated Laplace (green line) and estimated normal (red line) distributions.

**Additive noise.** After a P&S process, the image obtained with one of the 2LQR codes consists of gray level pixels when the original 2LQR code is binary. Therefore, all gray level pixels can be separated into two classes: the black class, that consists of black pixels from 2LQR code, and the white class, that consists of white pixels from 2LQR code.

The noise cannot be assumed to be additive if the error distribution of the black pixels is different from the error distribution of the white pixels. As we know the true map of black and white pixel placement, we separate the  $\varepsilon_{jj'}$  in two classes: black class and white class. This leads to two random processes  $B(t, s)$  and  $W(t, s)$ . The histograms of error black random processes and white random processes are illustrated in Fig. 8.5.

We apply the Mann-Whitney test in order to decide whether or not the error distribution can be said to be additive, i.e. quantify the hypothesis that the black and white distributions are the same. The hypothesis that the distributions are the same is rejected at a significance level of 0.05. Therefore, the noise added by the scanning process cannot be considered as being additive.

**Stationarity of first order.** To verify the first order stationarity of random process  $Y(t, s)$ , we use the method described in Section 8.3.1. The null hypothesis of Kolmogorov-Smirnov test is formulated as: *the two independent random subsets of mean*

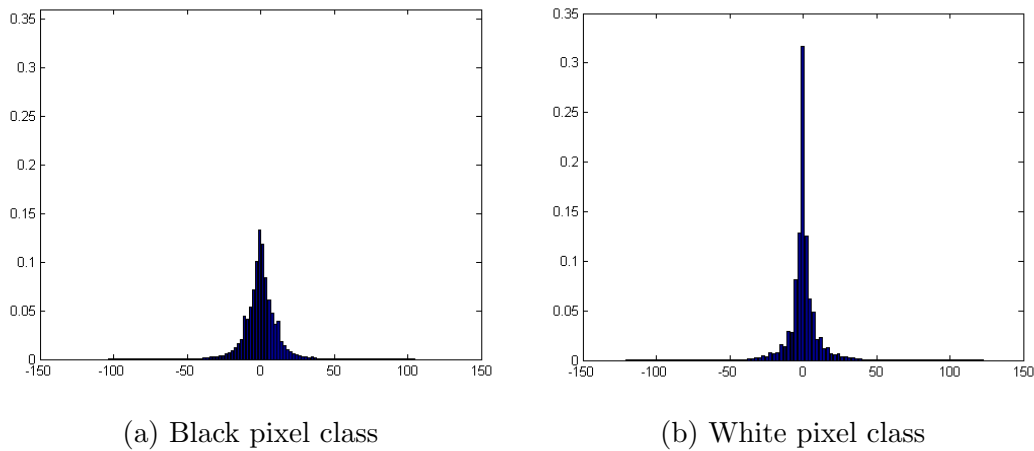


FIGURE 8.5: Histograms of scan noise introduced to a) Black pixels and b) White pixels.

*value set have the same distributions.* The Kolmogorov-Smirnov test does not reject the null hypothesis at a significance level of 0.05, therefore our process can be supposed being stationary of first order, i.e. the statistical average respects  $E[Y(t, s)] = \mu_Y$ .

**Ergodicity of first order.** To verify the first order ergodicity of the random process  $Y(t, s)$ , we use the method described in Section 8.3.2. The null hypothesis is formulated as: *the spatial mean and statistical mean are identical.* The Mann-Whitney test does not reject the null hypothesis at a significance level of 0.05. Therefore, our process can be supposed being ergodic of first order.

**Stationarity of second order.** To verify the second order stationarity of the random process  $Y(t, s)$ , we use the method described in Section 8.3.3. The null hypothesis of the Kolmogorov-Smirnov test is formulated as: *the two independent random subsets of autocovariance set have the same distributions.* The Kolmogorov-Smirnov test rejects the null hypothesis at a significance level of 0.05, therefore our process is not stationary of second order. That means our process is not wide-sense stationary.

**Test for white noise.** A random process can be a white-noise process if it is wide-sense stationary (as it was mentioned in Section 8.2.1). Our process is not wide-sense stationary, therefore our process is not a white-noise process.

#### 8.4.2 Study of P&S noise

**Experimental setup.** This experiment aims at studying the noise added by P&S process. To achieve this noise, we propose to print an image  $n$  times and then scan each printed image once. Let  $I$  be the original image,  $P_i$  ( $i = 1, \dots, n$ ) be the  $n$  printed images of original image  $I$  and  $S_i$  be the  $n$  scanned samples of printed images  $P_i$ . The



scheme can be picked as:

$$I \rightarrow \mathbf{Print} \rightarrow P_i \rightarrow \mathbf{Scan} \rightarrow S_i. \quad (8.29)$$

Let us consider  $P_i$  as a function of  $I$  and  $\varepsilon_{P_i}$ ,  $S_i$  as a function of  $I$ ,  $\varepsilon_{P_i}$  and  $\varepsilon_{S_i}$ , i.e:

$$P_i = f(I, \varepsilon_{P_i}),$$

$$S_i = g(P_i, \varepsilon_{S_i}) = P_i \oplus \varepsilon_{S_i} = f(I, \varepsilon_{P_i}) \oplus \varepsilon_{S_i},$$

where  $f()$  is the function associated to the printing process,  $g()$  is the function associated to the scanning process and  $\varepsilon_{P_i}$  and  $\varepsilon_{S_i}$  are the noises introduced by the printer and the printer-and-scanner respectively. The  $\oplus$  operator shows that the printed and scanned image  $S_i$  can be represented by a function of the printing process and the scanner noise (that also depends on the function of the printing process).

If the noise has a regularity, we can consider to calculate the difference among samples in order to study the noise nature. We propose thus to calculate the differences among each pair of  $n$  samples, i.e. subtract pairs of scanned images to provide samples of the P&S noise:

$$\begin{aligned} S_i - S_{i'} &= \varepsilon_{S_i} - \varepsilon_{S_{i'}} & (8.30) \\ &= \varepsilon_P + \varepsilon_{PS_i} - \varepsilon_P - \varepsilon_{PS_{i'}} \\ &= \varepsilon_{PS_i} - \varepsilon_{PS_{i'}}, \end{aligned}$$

The  $\varepsilon_{ii'} = \varepsilon_{PS_i} - \varepsilon_{PS_{i'}}$  can be used to characterize the stationarity and the ergodicity of the noise introduced by the P&S process. These characteristics, that are inherent to  $\varepsilon_{ii'}$ , are also inherent to  $\varepsilon_{PS_i}$  and  $\varepsilon_{PS_{i'}}$ . Therefore, we can characterize the P&S noise by characterizing the stationarity and ergodicity of  $\varepsilon_{ii'}$ .

Let us introduce the same experimental setup when using the terminology of random processes. After a P&S process each image can be considered as random variable  $X(t)$ . Therefore, the set of  $n$  P&S images can be considered as random process  $X(t, s)$ , where  $s = \{1, \dots, n\}$  is the dimension of our data set and  $t = \{1, \dots, m\} \times \{1, \dots, m\}$  is the set of pixels in an image  $I$ .

As shown before, the P&S noise function can be obtained by subtracting every two random variables  $X(t, s_j)$  and  $X(t, s_{j'})$ , in order to create a new random process  $Y(t, s)$  with  $s = \{1, \dots, n'\}$ :

$$Y(t, s) = X(t, s_j) - X(t, s_{j'}), j \neq j', s_j = s_{j'} = \{1, \dots, n'\}, \quad (8.31)$$

where  $n' = \frac{n(n-1)}{2}$ , as was shown in Section 8.4.1.

**Database description.** We print one 2LQR code  $n = 30$  times and then scan all

printed images once. Each P&S image has a  $300 \times 300$  pixel size, i.e.  $m = 300$ . The number of subtracting pairs we have is equal to  $n' = 435$ .

Let us visualize the distribution of  $Y(t, s)$  for any values of  $t$  and  $s$  (i.e. as if it was an ergodic process) in Fig. 8.6. In total here we visualize  $435 \times 90,000 = 39,150,000$  data observations. The mean value of this distribution is equal to  $\mu = -0.3563$ , the standard deviation is equal to  $\sigma = 14.6581$ .

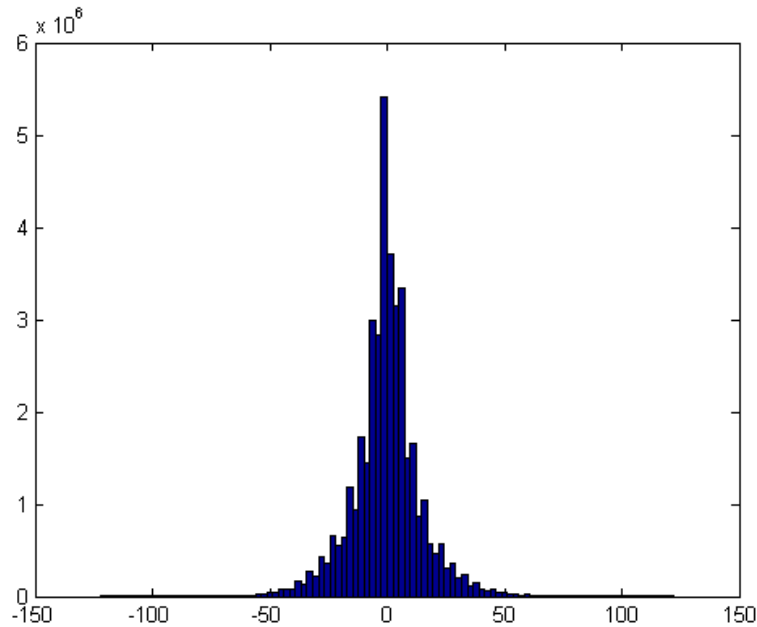


FIGURE 8.6: P&S noise distribution.

**Characterization of the  $Y(t, s)$  distribution.** In this experiment, we want to verify whether the  $Y(t, s)$  follow a given classical distribution (a normal or a Laplace) or not. In order to answer this question, we compare the distribution of the random process  $Y(t, s)$  with each specified distribution using the  $\chi^2$  goodness-of-fit test described in Section 8.2.3.

A reduced sample data set has been created by randomly selecting  $n' = 100$  vectors of  $Y(t, s)$  (for lower computational time). We therefore work with a data subset of 9,000,000 samples. To counteract working with a subsample, we perform each test several times.

An algorithm of  $\chi^2$  goodness-of-fit test involving the estimation of mean and variance is presented in Algorithm 2. The result of this test is that both hypotheses are rejected at a significance level of 0.05. Thus our data-set has neither a normal nor a Laplace distribution.

We computed two distances to see how far the empirical CDF based on the initial data is from the CDF of a normal or of a Laplace distribution. This comparison involves the calculation of the maximal distance  $d_{max}$  between the CDFs (see the formula (8.27)) and the squared distance  $d^2$  between the CDFs (see the formula (8.28)).

Number of samples $N$	Laplace distribution		Normal distribution	
	$d_{max}$	$d^2$	$d_{max}$	$d^2$
30,000	0.0405	0.0141	0.0884	0.1208
100,000	0.0471	0.0185	0.0929	0.1297
200,000	0.0434	0.0168	0.0929	0.1247
300,000	0.0431	0.0158	0.0909	0.1225
400,000	0.0424	0.0158	0.0922	0.1233

TABLE 8.2: Distances among CDFs of initial data (P&S noise) and estimated Laplace and normal distributions.

Table 8.2 presents the differences among the CDF based on our data-set and the CDF(s) of a normal and a Laplace distributions. We notice that these distances are rather stable under the changes of sample size.

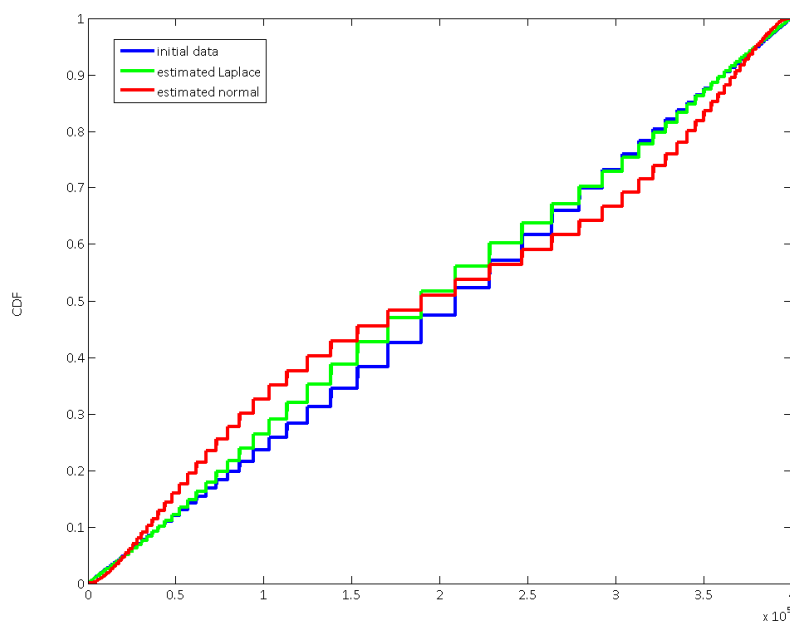


FIGURE 8.7: CDF of P&S noise (blue line), estimated Laplace (green line) and estimated normal (red line) distributions.

We illustrate these distances among CDFs in Fig. 8.7. We note that the CDF of the initial data (blue line) is closer to the CDF of the Laplace distribution (green line) than the CDF of the normal distribution (red line). Therefore the P&S noise can be said to be closer to a Laplace distribution than to a normal distribution.

**Additive noise.** After a P&S process, the image obtained with one of the 2LQR codes consists of gray level pixels when the original 2LQR code is binary. Therefore, all gray level pixels can be separated into two classes: the black class, that consists of black pixels from 2LQR code, and the white class, that consists of white pixels from 2LQR code.

The noise cannot be assumed to be additive if the error distribution of the black pixels is different from the error distribution of the white pixels. As we know the true map of black and white pixel placement, we separate the  $\varepsilon_{ii'}$  in two classes: black class and white class. This leads to two random processes  $B(t, s)$  and  $W(t, s)$ . The histograms of error black random processes and white random processes are illustrated in Fig. 8.8.

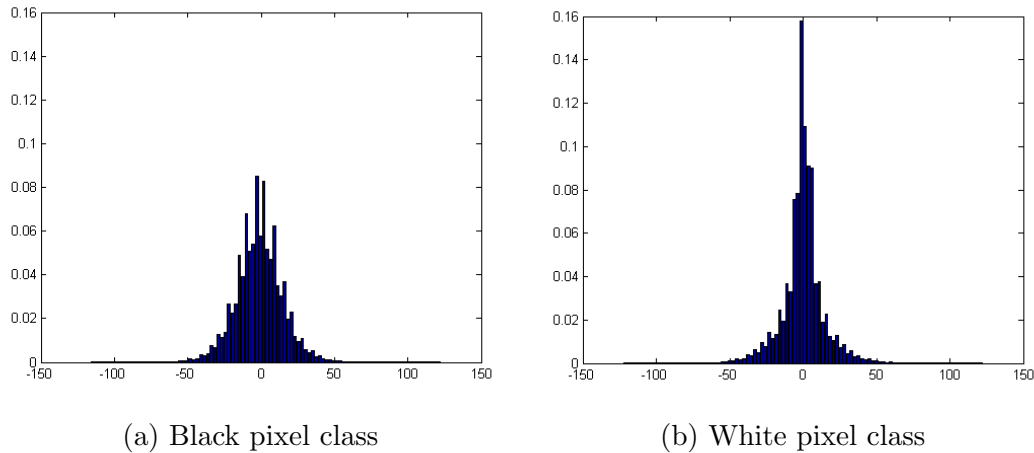


FIGURE 8.8: Histograms of P&S noise introduced to a) Black pixels and b) White pixels.

We apply the Mann-Whitney test in order to decide whether or not the error distribution can be said to be additive, i.e. quantify the hypothesis that the black and white distributions are the same. The hypothesis that the distributions are the same is rejected at a significance level of 0.05. Therefore, the noise added by the printing and scanning process cannot be considered as being additive.

**Stationarity of first order.** To verify the first order stationarity of random process  $Y(t, s)$ , we use the method described in Section 8.3.1. The null hypothesis of Kolmogorov-Smirnov test is formulated as: *the two independent random subsets of mean value set have the same distributions*. The Kolmogorov-Smirnov test does not reject the null hypothesis at a significance level of 0.05, therefore our process can be supposed being stationary of first order, i.e. the statistical average respects  $E[Y(t, s)] = \mu_Y$ .

**Ergodicity of first order.** To verify the first order ergodicity of the random process  $Y(t, s)$ , we use the method described in Section 8.3.2. The null hypothesis is formulated as: *the spatial mean and statistical mean are identical*. The Mann-Whitney test rejects the null hypothesis at a significance level of 0.05. Therefore, our process is not ergodic of first order.

**Stationarity of second order.** To verify the second order stationarity of the random process  $Y(t, s)$ , we use the method described in Section 8.3.3. The null hypothesis of the Kolmogorov-Smirnov test is formulated as: *the two independent random subsets*

of autocovariance set have the same distributions. The Kolmogorov-Smirnov test rejects the null hypothesis at a significance level of 0.05, therefore our process is not stationary of second order. That means our process is not wide-sense stationary.

**Test for white noise.** A random process can be a white-noise process if it is wide-sense stationary (as it was mentioned in Section 8.2.1). Our process is not wide-sense stationary, therefore our process is not a white-noise process.

## 8.5 Experiment of color distributions after P&S process

In Section 3.4, we mentioned that the authors in [10] experimentally show that the black/white block of pixels are related to random variables following asymmetric log-normal distributions. Using our database, we have decided to do the same experiment. Nevertheless, as in our experiments we used the 2LQR code images, the black and white pixels could be isolated. The results presented in this section are the work-in-progress, and we suppose to continue the study of color changes after P&S process.

**Experimental setup.** We print and scan a graphical code 90 times. Therefore we have 90 samples of 2LQR code (see Fig.7.9). The histogram of all these images is illustrated in Fig. 8.9.a. Then, using the true map, we separate our pixels in two classes: black class and white class. We illustrated the histograms of obtained classes in Fig. 8.9.b.

As expected, the histogram of black pixels is uni-modal (see Fig. 8.10.a). However, the bi-modal histogram of white pixels needs a more throughout study. Indeed, our P&S samples contain three different kinds of pixels: black pixels from textured patterns, white pixels from textured patterns and white pixels from completely white modules (see Fig. 7.9). Therefore, we separate the white pixels again in two classes: white pixels of textured patterns and white pixels of white modules. The results of this separation are illustrated in Fig. 8.10.b and Fig. 8.10.c.

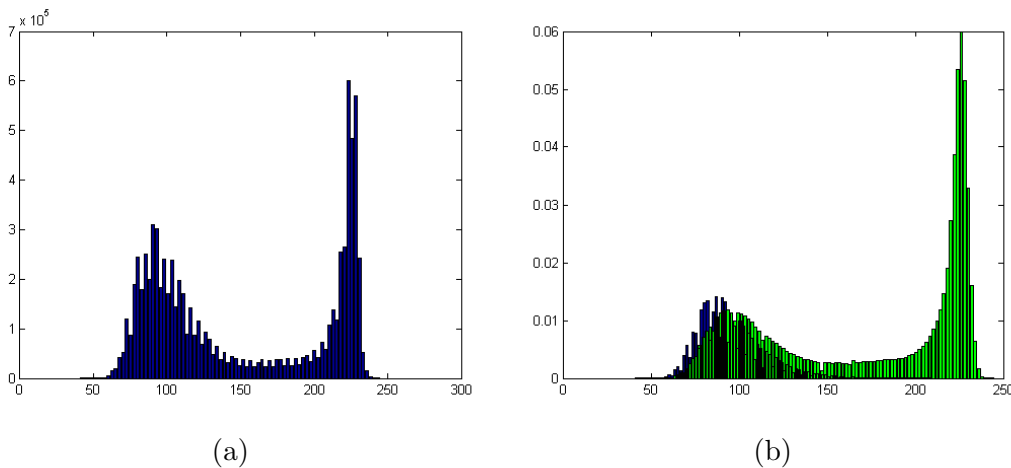


FIGURE 8.9: Histogram of 90 P&S samples a) all pixels together and b) pixels separated in black (blue color) and white (green color) classes.

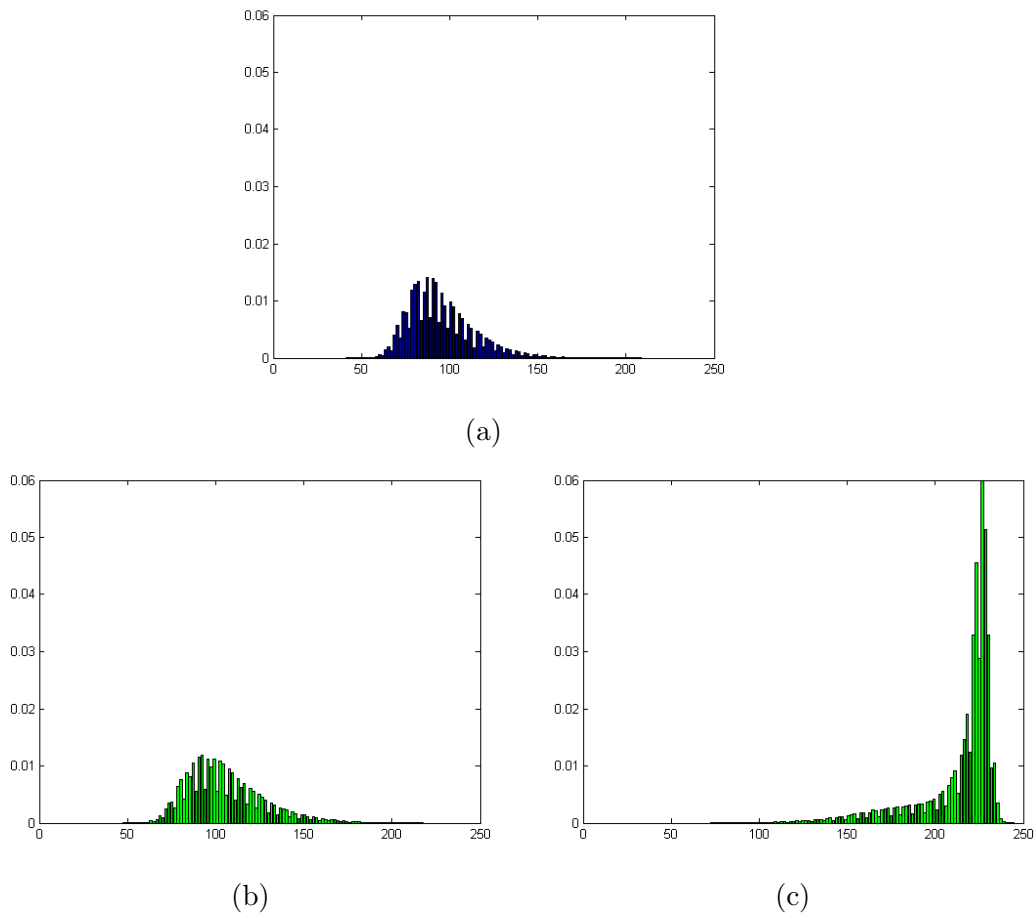


FIGURE 8.10: Histogram of 90 P&S samples a) Black pixels, b) White pixels from textured patterns and c) White pixels from white modules.

These histograms (Fig. 8.10.b and Fig. 8.10.c) clearly show that the white pixels of textured patterns are placed at the same part of histogram as the black pixels. This could be simply explained by the structure of our textured patterns, where the black and white pixels change frequently.

We have applied the  $\chi^2$  goodness-of-fit test to verify whether these distributions have normal or log-normal nature. The  $\chi^2$  test rejects the null hypothesis, that these distributions are normal, at a significance level of 0.05. In the same time, it accepts the null hypothesis, that they are log-normal, at a significance level of 0.05.

These results are only the beginning of this study. These experiments need to be evaluated better, using different statistical tests. These first results highlight several open questions:

- The histograms seem to present artifacts (looking like quantification effects). We should find the nature of these artifacts and find the possibility to remove them.
- Raw histograms seem to present a log-normal distribution. Several additional statistical test need to be done in order to confirm or disprove these observations.

## 8.6 Conclusions

In this chapter we proposed to experimentally study the impact of P&S process as a random process. The main results of this chapter are: the characterization of the scanning noise is done independently from the printing noise, the invalidation of the classical hypothesis that the P&S noise is normal distributed, and either white or ergodic. Additionally, we show that the scanning noise and the P&S noise are both closer to Laplace distribution, than to normal distribution. The first experiment of color distributions after P&S process are presented in this chapter. We conclude that this problem needs a more throughout study, but the first experimental results are interesting.

The contributions of this chapter will be published in the international conference IT&S Electronic Imaging (EI) 2016.

## Chapter 9

# Conclusions and perspectives

### 9.1 Conclusions

The availability of high quality copy machines provides a large amount of printed document counterfeits. Numerous authentication techniques based on security printing, graphical codes, hashing or local hashing have been suggested earlier. In this thesis we focus on graphical codes sensitive to printing, scanning and copying processes. The proposed graphical elements consist of specific textured patterns, that are sensitive to reproduction process.

The combinations of these textured patterns are separated into three classes (distinguishable, semi-distinguishable and indistinguishable combinations) depending on human perception. This separation is based on the respondents votes for different combinations. Then, the characterization of obtained results have been performed. We show, that the textured patterns with related spectra create the indistinguishable pattern combinations. We use the semi-distinguishable combinations for proposed graphical code called textured pattern with visual message. The choice of such pattern combinations is done to facilitate the pattern detection after print-and-scan process. Several pattern detection methods based on maximization of correlation values have been suggested. The method, that maximizes the Pearson correlation value between original and printed-and-scanned patterns, performs the best detection results.

The high density barcodes have several problems during reading process due to frequent changes of black and white modules. We suggest to compare the printed-and-scanned modules with its originals in order to refine the binarization results and improve the readability of high density codes. The weighted mean squared error measure, that highlights the significance of central module pixels and decreases the significance of border module pixels, has been suggested. The experiments, that were performed using high density QR code (version 40 with module size  $3 \times 3$  pixels), show the effectiveness of this measure. This method increases the recognition rate at least up to 93%, so we can use the QR code in low error correction level, that allows us to store more information.



The high density graphical codes have a lot of applications. Therefore, a new rich QR code, called Two Level QR (2LQR) code, was presented in this thesis. The suggested 2LQR code has two storage levels: the public level is readable by any standard QR code reader, the private level is accessible only for authorized users. The private level is constructed using textured patterns. Therefore, this storage level allows us not only to increase the storage capacity of initial QR code, but also to detect unauthorized duplication of the 2LQR code. The maximal tested storage capacity of this 2LQR code is  $13934 \text{ bits/inch}^2$  using 8 textured patterns of size  $6 \times 6$  pixels and the QR code version 40, where the public level contains  $7548 \text{ bits/inch}^2$  and the private level stores  $6386 \text{ bits/inch}^2$ . Numerous experiments of 2LQR code copy sensitivity were provided using four different copy machines. The obtained results show the robustness of our 2LQR code against duplication. The proposed enrichment process can be extended to any standard barcode.

In the end, the experimental studies of print-and-scan process show us that this process cannot be modeled as white and ergodic in the wide sense Gaussian process. Additionally, the scanner noise was separated from printer noise. It is shown that the scanner noise seems to be mean stationary and impacts less in total print-and-scan noise.

## 9.2 Perspectives

During this work, several interesting future developments were determined. In this section we aim to present these perspectives, that could either improve the obtained results or start out new research work.

**Textured pattern improvements.** In the thesis, we have defined several restrictions for textured patterns, such as the use of binary images with constant number of black pixels. It is interesting to use gray level patterns, that offer the opportunity to employ the printer halftoning process and random printing effect. Probably, the use of gray level patterns will allow us to increase the storage capacity of graphical codes. Additionally, we can use textured patterns with different densities to improve the protection against opponent reconstruction of proposed graphical elements.

The effect of human perception is also essential to study. First of all, the study of pattern spectra should be performed, in order to find a way for automatic generation of patterns used in graphical elements. Secondly, the pattern combination criteria could be changed. In this thesis, these criteria are based on correlation values and experimental samples. Criteria based on DFT or DWT have to be studied. This choice of criteria could help us to apply pattern detection methods, that are not based on correlation.

An interesting direction is the use of indistinguishable pattern combinations into graphical codes. This research direction offers us several interesting studies in HVS and printing process. Of course, the detection of such patterns will be much more difficult. But after several experiments we have some expectations, that the use of contour detection can

offer us interesting results.

**Improvements of graphical codes with textured patterns.** Several perspectives concerning storage capacity and application scenario can be developed.

A third (or extended second) level can be created by replacement of white modules with low density textured modules (10–20% black pixels in pattern). This replacement could create reading problems due to changes in contrast among "almost white" patterns and "almost black" (textured) patterns. Therefore, this idea is worth to be developed, in order to determine the pattern contrast trade-off.

In order to complicate the reconstruction of suggested graphical codes, the textured patterns with the same density can be used to represent the same encoded character. Thus, in graphical codes for encoding of each alphabet character we use the textured patterns with defined number of black pixels, but these patterns can have different structures.

The suggested pattern detection methods have a strict supposition, that the graphical code is captured with low rotation angle. Therefore, the exact pattern position can be detected correctly. However, we should highlight that the methods based on correlation maximization are very sensitive to one pixel shifts. Therefore, the search of more robust textured pattern detection methods is needed. For example, these methods can be based on contour detection, DFT, DWT or Hough transformations.

The copy sensitivity of suggested graphical codes needs to be proved under several "smart" attacks: the reconstruction of textured patterns using an inverse print-and-scan process [37] or using huge amount of authentic samples.

Additionally, it could be interesting to blend the copy sensitivity of suggested two level barcodes (document support authentication) and tamper-proofing technique (document content authentication) [150]. For this, the specific document features, that are not sensitive to print-and-scan changes, but sensitive to unauthorized document changes, have to be suggested. This direction deserves study of perceptual hashes and forensics, as well as huge amount of different reconstruction attacks.

**R&D project.** We suggest to develop an application capable to read the two level barcode after capture by camera or smartphone. This process can significantly resize the proposed graphical code and, therefore, the textured patterns. Additionally, due to inhomogeneous lighting conditions and human factors (shaking, capture distance, precision), the correct detection of textured patterns using suggested in this thesis methods can be failed. Thus, we need to study the geometrical changes of textured patterns after camera capture, to find the methods of acceptable correction without loss of textured pattern sensitivity to copying process, and to develop the acceptable pattern recognition method.

**Impact of P&S process.** Last, but not the least perspective is a throughout study of P&S process, that is stated in Chapter 8. We need to use statistical methods to determine the characteristics of P&S process and to study the changes of black and white pixels in our textured patterns. Additionally, we have to conduct the same experiments using different printers and scanners.



# Chapitre 10

## Résumé en Français

### 10.1 Introduction

De nos jours, la sécurisation des données est très importante pour protéger les droits d'auteur et vérifier l'authenticité des documents [28]. Les défis actuels sont par exemple la détection de fraudes sur des factures, des chèques bancaires, des diplômes, des formulaires d'impôt ou d'autres documents de valeur. En raison du développement et de la disponibilité des appareils d'impression et de numérisation, le nombre de documents de valeur et d'emballages de produits faux ou contrefaits augmente de plus en plus. Par conséquent, différents éléments de sécurité ont été proposés pour prévenir ces actions illicites.

Les éléments de sécurité diffèrent par leur complexité de production, leur niveau de protection et leur processus de vérification. Par exemple, la fabrication de filigranes est difficile et prend beaucoup de temps, mais ils peuvent être vérifiés à l'œil nu. Dans le même temps, des codes graphiques sensibles à la copie [108] sont générés de façon simple, mais leur vérification ne peut être effectuée que par un spécialiste équipé d'un scanner à haute résolution. Enfin, la vérification de papiers sécurisés ne peut être effectuée que par des professionnels, en laboratoires.

Les codes graphiques sensibles à la copie sont très populaires dans le domaine de l'authentification de produits, de par leur facilité de génération, leur faible coût de production, leur processus de vérification simple et leur haute performance d'authentification. Plusieurs entreprises ont ainsi été créées dans le but de trouver et commercialiser de nouveaux codes graphiques sensibles à la copie pour développer ce marché.

Authentication Industries (AI) est une start-up montpelliéraine travaillant dans la protection et l'authentification des documents. AI développe des solutions contre les fraudes à l'aide d'éléments de sécurité graphique, qui sont utilisés, par exemple, pour la protection d'emballage de produits et la certification de documents imprimés à partir d'un support numérique.

Afin de renforcer la base de recherche de l'entreprise, AI et l'équipe ICAR (Image &

Interaction) du Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier (LIRMM) ont mis en place une collaboration. Les contributions de cette thèse sont les résultats de cette période de collaboration de 3 ans.

Dans cette thèse, nous présentons de nouveaux éléments de sécurité qui visent à protéger les documents de valeur et les emballages contre la copie non autorisée. De plus, ces codes graphiques permettent le stockage d'une grande quantité d'informations secrètes.

La synthèse du système d'authentification proposé, utilisant ces éléments de sécurité, est illustrée à la Fig. 10.1. Ce système, s'organise autour de deux acteurs principaux : d'une part la source légitime, qui encode le message secret, génère le code graphique de sécurité et l'imprime en utilisant une imprimante autorisée et d'autre part, le centre de compétence qui exécute le processus de vérification. Ce processus de vérification consiste à numériser le code graphique de sécurité imprimé, à effectuer les procédures de pré-traitement et enfin à appliquer un test d'authentification. Enfin, si le code graphique est reconnu comme authentique, l'extraction du message peut être faite.

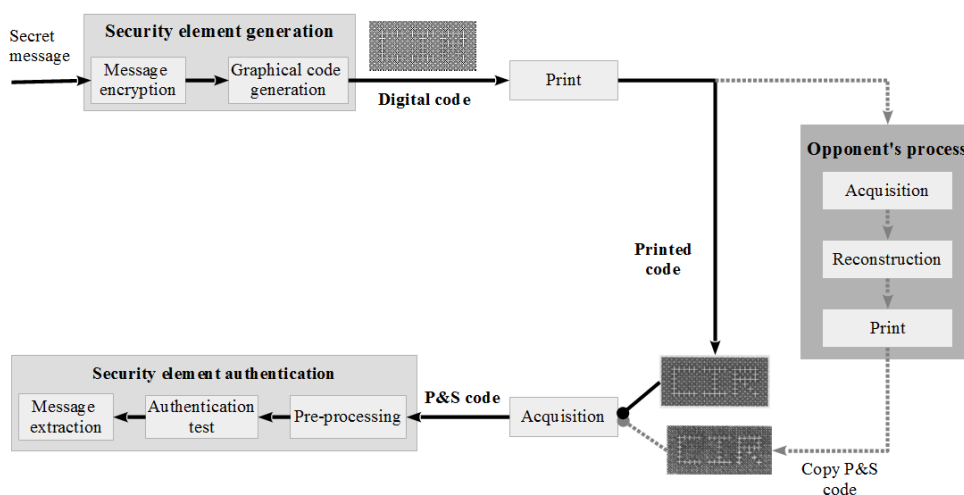


FIGURE 10.1: Système d'authentification considéré en utilisant des éléments de sécurité imprimés.

La plupart des contrefaçons sont réalisées entre l'impression (l'émission du document) et la numérisation (le contrôle). Un troisième acteur intervient alors en tant qu'adversaire tentant de falsifier le document protégé par un code graphique. Son but est de créer un code graphique considéré comme authentique par le centre de compétence. Il a uniquement accès au code graphique imprimé et doit donc numériser, reconstruire et ré-imprimer le code graphique pour créer un document ressemblant à l'original.

## 10.2 État de l'art

### 10.2.1 Outils de protection de documents

Dans notre vie quotidienne, nous traitons souvent différents documents : diplômes, contrats, factures, passeports et billets de banque. A chaque instant il faut décider si ce document est un authentique ou un faux. Pour nous guider, chaque document de valeur a un ensemble particulier d'éléments de protection.

La protection des documents est un domaine exploré depuis longtemps et bien développé. Il existe un grand nombre d'éléments de protection différents et de techniques qui peuvent être classées différemment. Nous proposons une classification des outils de protection de documents. Bien sûr, en raison des liens entre les différents outils de protection, notre classement est parfois perméable.

La protection de document est un ensemble d'éléments visuels, de caractéristiques physiques et de techniques spéciales, qui permettent de conclure avec certitude à l'authenticité. Il existe trois niveaux d'outils pour la protection des documents en fonction de la complexité du contrôle : 1) pour la vérification par des consommateurs ; 2) pour la vérification par des spécialistes ; 3) pour la vérification au laboratoire. Le premier niveau utilise des éléments de protection qui peuvent être vérifiés par des personnes ordinaires sans aucun dispositif spécifique. Un exemple de ces éléments est un filigrane. Le deuxième niveau contient des éléments de protection qui peuvent être vérifiés par des personnes formées utilisant des dispositifs simples comme des scanners et des loupes. Les éléments vérifiables par des tests automatiques sont également dans le deuxième niveau. Les codes graphiques et certaines techniques d'impression sont des exemples associés à ce niveau. Le troisième niveau contient des éléments de protection qui doivent être vérifiés dans des laboratoires spécifiques par des experts. Les dégâts irréversibles occasionnés au document peuvent être acceptés dans ce niveau. Ainsi, des tests de vérification peuvent exiger des microscopes de haute capacité ou une expertise chimique. On peut citer comme exemple un ADN synthétique qui impose une vérification spécifique en laboratoire.

Les techniques de protection de documents ont été créées pour lutter contre les contrefaçons et les falsifications. La contrefaçon est une reproduction illégale à l'identique du document. Cela signifie que le contrefacteur essaie de copier la totalité ou la plupart des symboles de protection sans modifier le contenu du document afin de produire un faux qui ne peut être distinguable de l'original. La falsification est la production d'un document faux. Cela signifie que le contenu du document est modifié pour produire un nouveau document à partir d'une source non autorisée et présentant des informations bien formées.

La sécurité optique (par exemple un hologramme) est un type fort de protection robuste pour documents de valeur. Les points forts de ces protections viennent de l'accès à la technologie (par exemple, la production réelle de l'hologramme) et des difficultés

de mise en œuvre. Les outils de protection de document sont divisés en trois branches principales [149] :

- Protection technologique : des éléments de sécurité sont ajoutés lors de la fabrication du papier ou par la technologie d'impression.
- Protection chimique : concerne la structure chimique du papier, des encres (UV, thermique, ...) et des propriétés physiques des éléments de sécurité.
- Protection par impression sécurisée : englobe certaines encres de sécurité (optiquement variables), les techniques d'impression de sécurité et les modes d'impression de sécurité.

Dans cette thèse, nous nous concentrons sur les effets du processus d'impression pour l'authentification des documents. Le processus d'impression et la structure du papier offrent des marqueurs permettant de protéger un document contre la copie. Les éléments de sécurité standards composant un document (classiquement un fond et des images sécurisées) n'utilisent pas ces caractéristiques. Les techniques d'impression pour l'authentification de documents sont basées sur deux déclarations :

- Chaque fois qu'une image numérique originale est imprimée ou numérisée, certaines informations sont perdues. Cet effet est appelé "principe de la perte de l'information" [108]. Les processus d'impression et de numérisation sont affectés par le bruit, le flou et d'autres changements présentés dans la section 10.2.2. Les canaux de communication, où les informations sont transmises souffrent de bruit et de variations, se caractérisant toujours par une perte d'informations. La perte peut être minimale et imperceptible à l'œil nu, mais être suffisante pour contrarier le test d'authentification.
- Chaque dispositif d'impression et de copie a sa propre signature. L'utilisation de cette signature et des modifications spécifiques ajoutées par le dispositif sont utilisées pour le test d'authentification. Des systèmes d'analyse d'images spécifiques peuvent détecter les modifications apportées à des documents imprimés, même si cette modification est invisible [140].

### 10.2.2 Modélisation du processus d'impression et de numérisation et traitement d'image

Aujourd'hui, les dispositifs d'impression et de numérisation sont accessibles à tous, induisant l'augmentation du nombre de documents de valeur ou de documents administratifs contrefaits ou falsifiés. Les documents papier souffrent tout d'abord de distorsions au cours du processus d'impression, puis de l'impact de la numérisation réalisée lors de l'authentification automatique.

Deux exemplaires imprimés du même document ne sont pas distinguables à l'œil nu alors qu'ils diffèrent l'un de l'autre d'un point de vue numérique. Afin de comparer l'image imprimée avec l'image numérique originale, l'image imprimée est numérisée. Ainsi les effets des procédés d'impression et de numérisation ne sont pas séparables l'un de l'autre et les distorsions résultantes proviennent des deux processus [164].

**Le processus d'impression** peut être assuré par l'utilisation de diverses technologies : les plus disponibles sont les imprimantes à jet d'encre, les imprimantes laser et les copieurs électrostatiques. Les imprimantes laser et les photocopieurs électrostatiques ont des structures très proches alors que celles des imprimantes à jet d'encre est très différente.

Tout document réalisé à partir d'une imprimante électrophotographique comporte des défauts causés par les fluctuations électromécaniques ou les imperfections du mécanisme d'impression [87]. Les processus physiques qui peuvent intervenir sont :

- La lumière du laser, localisation droite, la taille et la focale peuvent produire des imperfections et peuvent varier durant la vie de l'imprimante.
- Les mouvements d'éléments mécanique et optique peuvent introduire des imperfections.
- Le système optique peut introduire des non-linéarités [164].
- Les fluctuations de la vitesse angulaire du tambour de photoconducteur optique peuvent causer des fluctuations.
- Le transfert d'encre du tambour de photoconducteur optique vers le papier peut produire des erreurs aléatoires.
- Le fixateur peut impliquer des défauts thermiques.
- Des défauts de fabrication peuvent conduire à une distribution non uniforme du toner.
- Le nettoyage et la détérioration dans le temps peuvent provoquer l'impression d'objets supplémentaires fantômes.

Les options d'impression à jet d'encre engendrent une signature intrinsèque très complexe [28]. Une impression à jet d'encre peut être affectée par :

- La vitesse non-constante et la fluctuation du charriot peut introduire des erreurs d'impression.
- Les buses obstruées peuvent avoir un impact sur l'image imprimée.
- Le processus physique générant les gouttes peut affecter l'image imprimée.
- La forme des points diffère selon le mouvement du charriot et le nombre de passages nécessaire.

Les éléments les plus significatifs considérés lors de l'impression sont le halftoning numérique, la résolution de l'imprimante, la distribution de l'encre et la qualité du papier.

**Le dispositif de scanner** numérise optiquement, c'est à dire produit une image numérique à partir de documents imprimés ou d'objets. Comme pour le processus d'impression, les caractéristiques physiques du processus de numérisation peuvent affecter l'image obtenue :

- Les fluctuations de la vitesse du moteur peuvent entraîner des variations d'enregistrement de la couleur dans le document numérisé [28].



- Le procédé de fabrication des capteurs d'imagerie induit une variabilité pouvant introduire des distorsions et du bruit.
- La capture photo-électronique peut introduire des variations dans les pixels numérisés. Outre les imperfections physiques, la correction gamma et la résolution du scanner sont des caractéristiques importantes.

Les chercheurs modélisent le processus d'impression et de numérisation comme un canal d'authentification [105]. Le processus d'impression à très haute résolution peut être considéré comme un processus stochastique en raison de la nature des caractéristiques de l'imprimante [106]. Les auteurs simulent le processus d'impression comme une distribution gaussienne généralisée ou une distribution log-normale (qui a été proposée par [10]).

### 10.2.3 Code-barres enrichis

Un code-barres est une représentation optique des données lisible par une machine et utilisé pour créer une relation entre les données et l'objet qui les porte. Les exigences définies pour les créateurs de code-barres sont une forte densité d'informations codées, une grande fiabilité du processus de lecture, un coût minimal de production et un coût minimal du matériel de lecture [102].

Aujourd'hui, les codes graphiques, tels que les code-barres EAN-13 [1], Quick Response (QR) Code [4], DataMatrix [3], PDF 417 [2], sont souvent utilisés dans nos vies quotidiennes. Ces codes ont un grand nombre d'applications, notamment : le stockage d'informations (publicité, description d'œuvre d'art), la redirection vers des sites Web, le track and trace (pour les billets de transport ou les marques), l'identification (informations sur des passagers, des produits de supermarchés), etc.

La popularité de ces codes est principalement due aux caractéristiques suivantes : ils sont robustes pour le processus de copie, facile à lire par tout dispositif et tout utilisateur, ils ont une haute capacité d'encodage renforcée par l'utilisation de codes correcteurs d'erreurs, ils ont une petite taille et sont résistants aux distorsions géométriques. Toutefois, ces avantages indéniables ont aussi leurs contreparties :

1. La capacité de stockage des code-barres est limitée. L'augmentation de la capacité de stockage implique la croissance significative de la taille des code-barres.
2. Les informations codées dans un code-barres sont toujours accessibles à tous, même si elles sont chiffrées et sont donc seulement lisibles pour les utilisateurs autorisés (la différence entre "voir" et "comprendre").
3. Il est impossible de distinguer un code-barres imprimé de sa copie en raison de leur insensibilité au processus de reproduction.

L'évolution des code-barres des versions à une dimension (1D) vers des systèmes à quatre dimensions (4D) peut être décrite comme suit :

- Les code-barres 1D (par exemple Universal Product Code [1]) représentent des données en faisant varier les largeurs et les espacements des lignes parallèles. Ils ont d'énormes limites de capacité de stockage.
- Les codes linéaires empilés (Portable Data File (PDF417) [2]) encodent l'information en utilisant des barres, des espaces et des blocs.
- Les code-barres 2D stockent l'information dans les deux directions du plan utilisé pour augmenter la capacité de stockage. Les code-barres les plus connus sont les QR codes [4], Datamatrix [3] et Aztec [5].
- Les code-barres 2D avec signification visuelle [8, 38, 40] augmentent l'esthétique des code-barres 2D en ajoutant différentes couleurs, formes et/ou logos.
- Les codes graphiques enrichis encodent des informations en utilisant les deux dimensions classiques et en considérant une dimension supplémentaire (en utilisant des couleurs [47, 151], la structure des modules [27, 77, 91] ou des techniques cachées spécifiques [83, 20, 138, 156]).
- Les code-barres 4D [75] encodent des informations à l'aide de deux dimensions, de la couleur et du temps.

Un des points forts des code-barres est leur résistance aux distorsions des processus d'impression et de numérisation. Cette caractéristique les rend cependant inutilisables dans des applications d'authentification de support. Néanmoins, la capacité de stockage élevée et les bas coûts de production, incitent les chercheurs à mettre au point des codes graphiques sensibles aux processus d'impression et de numérisation.

Les codes graphiques, sensibles à la reproduction non autorisée [108, 110], sont basés sur la nature aléatoire des processus d'impression et de numérisation et sur l'impossibilité de les reconstruire.

Les codes graphiques enrichis qui autorisent les données cachées et l'authentification de supports préservent le stockage de l'information dans le niveau standard et insèrent un niveau supplémentaire utilisable pour le stockage de données binaires et l'authentification du support [109].

## 10.3 Contributions

### 10.3.1 Motifs texturés utilisés pour l'authentification de documents

Comme présenté dans la section 10.1, l'objectif principal de cette thèse est de proposer une solution imprimable pour protéger des documents. La solution envisagée consiste en une image sensible aux processus d'impression et de numérisation. Nous appelons cette image un élément de sécurité. Cette section présente cet élément de sécurité avec un message incorporé.

L'objectif est d'intégrer un message (visible ou invisible) dans un élément de sécurité, peu visible, mais détectable automatiquement après impression et numérisation. Cet

élément de sécurité est utilisé pour l'authentification du support du document, et doit donc être sensible aux processus d'impression et de numérisation. Cet élément de sécurité doit contenir des informations.

Nous proposons d'intégrer un message dans l'élément de sécurité en utilisant des motifs texturés spécifiques. Ces motifs texturés sont choisis pour assurer une bonne reconnaissance après impression et numérisation et être difficilement reproductibles. Les différents motifs codant le message incorporé dans une image sont distinguables uniquement dans le cas d'un document légitime. La légitimité de cet élément de sécurité peut donc être vérifiée.

Les motifs texturés subissent des changements de structure, de forme et de couleur au cours de leur impression et numérisation. Nous avons expérimentalement souligné les distorsions ajoutées par ces processus dans un cas réel. Les résultats de ces expérimentations montrent :

- que la structure interne des motifs texturés est perdue, les images devenant floues et les couleurs étant modifiées (Tableau 5.1).
- qu'il est possible de compenser certains changements de couleur en utilisant une table de correspondance (LUT), dédiée à chaque paire imprimante-scanner. La Fig. 5.2 montre qu'une image perd surtout les couleurs sombres après impression et numérisation. La LUT peut alors corriger certains défauts de couleur. Cependant, il est très difficile d'envisager de construire une telle LUT pour chaque paire imprimante-scanner et chaque type de papier.
- que la valeur d'un même pixel varie beaucoup d'une impression à l'autre. En outre, les graphiques des Fig. 5.3.b - Fig. 5.3.g montrent que les pixels noirs peuvent être représentés par une valeur de niveaux de gris pris dans l'intervalle (60 – 160). Des pixels noirs peuvent donc parfois être considérés comme des pixels blanc, et vice versa. Le seuil de binarisation est alors très difficile à identifier et les méthodes standards de seuillage global sont alors très difficiles à appliquer pour les images imprimées et scannées.
- que la méthode de binarisation connaissant *a priori* le nombre de pixels noirs et blancs dans le motif texturé met en lumière l'effet aléatoire des processus d'impression et de numérisation ainsi que la perte de la structure des motifs texturés (Fig. 5.4).

Afin de déterminer les combinaisons optimales de motifs texturés produisant des éléments de sécurité reconnaissables automatiquement et peu perceptibles pour le système visuel humain, nous avons effectué des mesures expérimentales sur un panel représentatif d'utilisateurs. Un questionnaire a été réalisé pour déterminer les meilleures combinaisons de motifs texturés pouvant être utilisées pour la construction d'éléments de sécurité. Une fois identifiées, ces combinaisons de motifs texturés ont été analysées afin de déterminer les critères mesurables pouvant être retenus pour identifier les combinaisons optimales de motifs.

Les résultats obtenus permettent de séparer clairement les éléments de sécurité en trois groupes : les éléments de sécurité avec des motifs indiscernables, semi-distinguables et

reconnaissables. Schématiquement cette séparation peut être réalisée comme indiqué sur la Fig. 5.7. Les conclusions de cette étude sont :

- les éléments de sécurité avec des motifs indiscernables répondent aux objectifs de discrétion (message imperceptible) mais rendent la vérification de la légitimité impossible. En effet, ces motifs texturés ne se distinguent pas les uns des autres après le processus d'impression et de numérisation.
- les éléments de sécurité avec des motifs reconnaissables peuvent être utilisés pour l'intégration d'un message visible. Ces éléments résistent aux distorsions d'impression et de numérisation. Néanmoins, les motifs reconnaissables facilitent la reconstruction de l'élément de sécurité et ne peuvent donc pas être utilisés pour la vérification de la légitimité du document.
- les éléments de sécurité avec des motifs semi-distinguables sont les candidats répondant le mieux à nos exigences : insérer un message difficilement perceptible et assurer la vérification de la légitimité du document.

Cette étude a permis de montrer que les éléments de sécurité dits indiscernables sont obtenus lorsque les spectres de motifs texturés les constituant sont liés. Des motifs présentant des spectres non-connexes donnent des images avec des motifs reconnaissables.

Cette approche expérimentale permet de conclure que le choix de motifs texturés doit être fait en considérant deux fonctionnalités : la perception humaine et le décodage automatique. Le type d'éléments de sécurité proposé contient des motifs semi-distinguables, ayant les caractéristiques suivantes :

- Une taille constante  $r \times r$  pixels. La contrainte de taille est posée en raison de la construction et du processus de reconnaissance utilisé.
- Une image binaire noire et blanche. En raison de la procédure de tramage pendant l'impression, la production de couleur noire pure est moins aléatoire, que la production de toute autre couleur en niveaux de gris.
- Un rapport constant de pixels noirs (densité de motif  $b = const$ ). Cette contrainte permet de réduire l'espace des motifs texturés possibles.
- Des spectres liés entre eux. Cette caractéristique permet d'assurer que les motifs constituant l'élément de sécurité sont semi-distinguables.
- Des caractéristiques de combinaison. La valeur de corrélation entre le motif original et sa version dégradée doit être élevée. La combinaison des motifs doit satisfaire aux critères (5.2) et (5.3).

Nous avons proposé trois types d'images texturées : une image texturée avec message visuel (Fig. 10.2.a), une image texturée avec des lignes et colonnes blanches (Fig. 10.2.b) et une image texturée en damier (Fig. 10.2.c).

Nous proposons plusieurs méthodes de classification pour détecter les différents motifs : 1) en corrélant les motifs représentatifs  $C_l$  avec les modules extraits de l'image texturée

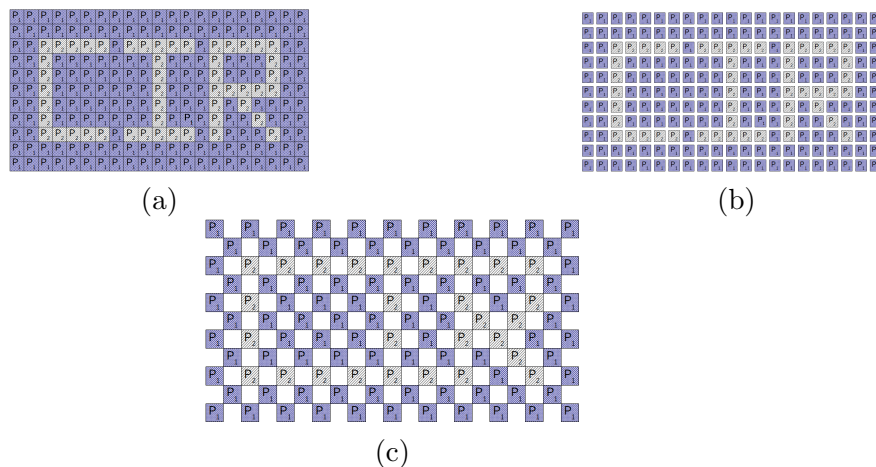


FIGURE 10.2: Génération d'images texturées avec un message visuel : a) Image texturée avec message visuel, b) Image texturée avec des lignes et colonnes blanches, c) Image texturée en damier.

numérisée  $I_s$ , 2) en utilisant une méthode de classification, 3) en utilisant la distance de Hamming entre les motifs représentatifs  $C_l$  et les modules extraits de l'image texturée numérisée  $I_s$ , et 4) par la corrélation de Kendall caractérisée.

Les résultats expérimentaux montrent que les méthodes basées sur la corrélation donnent les meilleurs résultats de reconnaissance des motifs, en utilisant les originaux comme motifs représentatifs  $C_l$ . Dans le même temps, l'utilisation de méthodes basées sur la corrélation de Kendall caractérisée donne les meilleurs résultats en utilisant les motifs représentatifs  $C_l$ .

### 10.3.2 Code-barres de haute densité

Les code-barres de haute densité constituent un domaine en pleine expansion : l'objectif consiste à augmenter la quantité d'informations stockée tout en maintenant, voire en diminuant, la taille du code. Ce compromis est le principal problème actuel. Plusieurs code-barres standards autorisent des versions permettant de stocker une grande quantité d'informations, comme le QR code. Toutefois, les versions denses de QR code posent des problèmes de lecture dus aux changements fréquents et à la petite taille des modules noirs et blancs. Les problèmes de binarisation et des corrections géométriques sont beaucoup plus importants pour un code dense, et ce, malgré la présence de codes correcteur d'erreurs.

La détection des motifs décrits dans la section précédente montre que les pixels centraux sont statistiquement moins changés que les pixels périphériques. Cette observation a initié l'idée d'une méthode permettant d'améliorer les résultats de binarisation des modules. Nous proposons donc une mesure favorisant la partie centrale de chaque module, basée sur l'erreur quadratique moyenne pondérée (WMSE). Cette mesure WMSE est utilisée pour une nouvelle méthode de classification des modules et pour l'amélioration

des méthodes de binarisation standard : binarisation par ISO et binarisation d'Otsu. Nous proposons d'augmenter l'importance des pixels centraux de chaque module et de diminuer l'importance des pixels périphériques en utilisant la mesure WMSE. Considérons le module  $M$  comme un bloc carré de  $(r + 1)^2$  pixels, où  $r$  est un nombre entier pair positif. Dans le système de référence, le pixel en haut à gauche a la position  $p(-\frac{r}{2}, -\frac{r}{2})$ , le pixel en bas à droite a la position  $p(\frac{r}{2}, \frac{r}{2})$  et le pixel central est  $p(0, 0)$ . La mesure WMSE entre un module original  $M$  (module noir  $M_B$  ou module blanc  $M_W$ ) et un module imprimé et numérisé  $M'$  (avec des pixels  $p'(i, j)$ ) est :

$$WMSE(M, M') = \frac{1}{T} \sum_{i=-\frac{r}{2}}^{\frac{r}{2}} \sum_{j=-\frac{r}{2}}^{\frac{r}{2}} w(i, j) D(p(i, j), p'(i, j)), \quad (10.1)$$

où

$$D(p(i, j), p'(i, j)) = (p(i, j) - p'(i, j))^2, \quad (10.2)$$

et  $w(i, j)$  est une fonction de pondération permettant de donner plus de poids aux pixels centraux,  $T$  est le nombre total de pondérations utilisées dans ces calculs. Nous proposons d'utiliser :

$$w(i, j) = \begin{cases} 1 - \frac{|i|+|j|}{W}, & \text{si } 1 - \frac{|i|+|j|}{W} \geq 0 \\ 0, & \text{sinon} \end{cases}, \quad (10.3)$$

où  $W$  est le poids, qui est donné à la valeur centrale du module, ce poids est un nombre réel et respecte la condition  $W \geq 1$ . Par conséquent, pour  $T$ , nous avons :

$$T = W(r + 1)^2 + \frac{r}{2}(r + 1)(r + 2). \quad (10.4)$$

La méthode proposée vise à augmenter le taux de reconnaissance d'un QR code. La classification des modules noirs et blancs du QR code s'effectue en 3 étapes. La première étape permet de proposer une classification initiale des modules en deux classes ( $C_W$  - classe de modules blancs,  $C_B$  - classe de module noir). La deuxième étape vise à calculer des modules caractéristiques pour les classes de modules noirs et blancs ( $M_{C_B}$  - module caractérisé noir,  $M_{C_W}$  - module caractérisé blanc). Enfin, la troisième étape compare chaque module avec les modules caractérisés ( $M_{C_B}$  et  $M_{C_W}$ ). La Fig. 6.7 présente la méthode proposée. Au cours de la première étape, nous classons nos modules en deux classes en minimisant la mesure WMSE entre les modules originaux et des modules imprimés et scannés. Au cours de la deuxième étape, nous calculons des images moyennes ou médianes qui seront utilisées comme des modules caractérisés. Au cours de la troisième étape, le classement est fait en minimisant la mesure WMSE entre les modules caractérisés et les modules imprimés et scannés.

Les méthodes standards de seuillage global exécutent la binarisation de QR codes en deux étapes :

- Binarisation d'image. Cette étape peut être considérée comme la classification des pixels en classes noir et blanc (identification d'un seuil global).
- Décision concernant la couleur des modules. Celle-ci détermine la couleur du module à l'aide du vote majoritaire pour chaque module. Les nombres de pixels blancs et noirs dans chaque module sont calculés. Le plus grand nombre correspond à la couleur (blanc ou noir) du module. Cette méthode ne permet pas d'assurer des taux de reconnaissance élevés pour les QR codes de haute densité.

La **méthode standard de binarisation ISO** [4] calcule le seuil global comme étant la valeur médiane entre la valeur maximale et la valeur minimale de réflectance de l'image.

La **méthode de binarisation d'Otsu** [98] choisit le seuil qui minimise la variance au sein du groupe.

Des mesures expérimentales sont réalisées en utilisant les deux méthodes de seuillage pour la binarisation, puis le vote majoritaire pour la décision finale de la couleur des modules. Afin d'améliorer le taux de reconnaissance des méthodes de binarisation globale, nous proposons de remplacer le vote majoritaire par le calcul de la mesure WMSE. La décision de la couleur des modules, dans ce cas, est réalisée en utilisant la mesure WMSE. La mesure WMSE est calculée entre les modules binarisés et les modules de référence noir et blanc ( $M_B$  et  $M_W$ ). Les gains apportés par la mesure proposée sont illustrés en Fig. 6.8.

Les expériences décrites ont été menées en utilisant une grande base de données contenant plus de 10,000 QR codes V40 imprimés et scannés à l'aide de trois imprimantes et trois scanners différents. Les résultats de ces expériences montrent que l'utilisation de la mesure WMSE améliore toujours le taux de reconnaissance des modules. Ces résultats sont présentés dans le Tableau 6.2 et le Tableau 6.3.

Une étude d'optimisation a été réalisée afin de déterminer la valeur optimale du paramètre pondéré  $W$ . Nous remarquons quatre cas particuliers :

- $W = 1$ . Dans ce cas, seul le pixel central est pris en compte. Ce cas ne peut pas être utilisé car le module ne peut être représenté par un unique pixel central. Les effets des processus d'impression et de numérisation rendraient le résultat trop aléatoire. En outre, si la correction géométrique n'est pas correctement réalisée l'étape de reconnaissance sera exécutée de façon incorrecte, et par conséquent, le QR code sera décodé de façon incorrecte. Comme mentionné dans la section 6.3, le paramètre de poids  $W$  doit satisfaire la condition  $W > 1$ .
- $W \rightarrow \infty$ . Dans ce cas, la méthode de reconnaissance tend à fournir les mêmes résultats que les méthodes de seuillage global. Nous perdons l'idée principale de la mesure WMSE car la différence entre les poids donnés aux pixels centraux et les poids donnés aux pixels de bord est trop faible.
- $W \in ]1, 2[$ . Dans ce cas, les poids des 4 pixels voisins diagonales du pixel central  $p(0, 0)$  sont égaux à 0. La binarisation est réalisée en utilisant les valeurs pondérées du pixel central et des 4 pixels connectés.



–  $W \in [2, \infty[$ . Dans ce cas, les valeurs pondérées de tous les pixels sont utilisées pour la binarisation. Expérimentalement, nous montrons que l'augmentation du paramètre de poids  $W$  diminue les résultats de la reconnaissance (voir les Fig. 6.13 - Fig. 6.15). Toutes les expériences réalisées montrent que les méthodes proposées améliorent les résultats de reconnaissance d'au moins 5%. Le taux de reconnaissance minimal avec nos méthodes est égal à 93%. Ce gain rend possible l'utilisation du niveau de correction d'erreurs faible (qui restaure 7% des mots de code) et permet donc de stocker plus d'informations dans un code-barres donné.

### 10.3.3 Motifs texturés utilisés pour des code-barres

Nous proposons de combler les lacunes évoquées des code-barres en enrichissant leur capacité d'encodage. Cet enrichissement est obtenu en remplaçant les modules noirs par des motifs texturés spécifiques. Outre le gain de capacité de stockage (en introduisant une notion d'alphabet  $q$ -aire), ces motifs peuvent être conçus pour être sensibles aux distorsions des processus d'impression et de numérisation. Ces motifs, qui n'introduisent pas de perturbations dans le processus de lecture standard, sont toujours perçus comme des modules noirs par tout lecteur de code-barres.

Ainsi, nous obtenons un code-barres à deux niveaux : un premier niveau (public) accessible à tout lecteur de code-barres standard en conservant donc ses caractéristiques fortes ; et un second niveau (privé) qui améliore les capacités et les caractéristiques du code-barres initial.

Les informations contenues dans le deuxième niveau sont donc encodées en utilisant un alphabet  $q$ -aire associé à un code correcteur d'erreurs (CCE)  $q$ -aire ( $q \geq 2$ ). Ce nouvel alphabet introduit une nouvelle dimension de codage constituant la principale différence du code à deux niveaux proposé par rapport aux autres code-barres utilisant un niveau supplémentaire. En effet, cette information secondaire est ici invisible pour les lecteurs de code-barres standards puisqu'ils perçoivent les motifs texturés comme des modules noirs. Le deuxième niveau peut donc être utilisé pour le partage de message privé. En outre, grâce à la sensibilité des motifs à l'impression et à la numérisation, nous proposons ces codes pour l'authentification de documents imprimés.

Dans cette section, toutes les descriptions sont faites pour renforcer les capacités d'un QR code classique, en proposant un nouveau QR code à deux niveaux (2LQR). Néanmoins, les procédures décrites peuvent être facilement étendues à tout code-barres existant.

Nous suggérons d'utiliser le code 2LQR dans deux scénarios : pour le partage d'un message privé d'une part et pour l'authentification d'un document d'autre part. La structure du code est la même dans les deux scénarios, les différences étant portées par les blocs de position. Dans le scénario pour le partage d'un message privé, les modules noirs des blocs de positions sont également remplacés par des motifs texturés alors que dans le



scénario pour l'authentification du document, ils ne sont pas modifiés (ils restent composés de modules noirs). La comparaison de QR codes standards avec des codes 2LQR suivant les deux scénarios est présentée dans la Fig. 10.3.

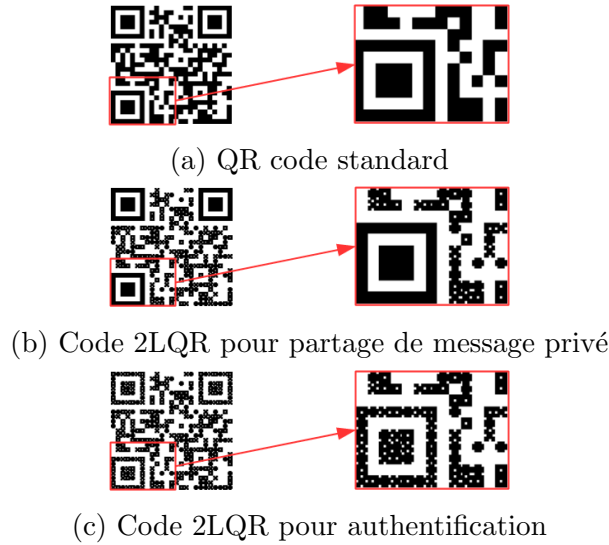


FIGURE 10.3: Comparaison de a) QR code standard avec b) Code 2LQR proposé pour partage de message privé et c) Code 2LQR proposé pour authentification.

Le système de génération de code 2LQR respecte les étapes suivantes. Les messages public  $M_{pub}$  et privé  $M_{priv}$  sont définis. Le message public est encodé dans le QR code en utilisant l'algorithme standard. Dans le même temps, le message privé est encodé en utilisant l'algorithme de CCE puis l'alphabet choisi. Enfin, le QR code avec les messages public  $M_{pub}$  et privé  $M_{priv}$  constitue le code 2LQR avec deux niveaux de stockage. Les étapes de génération du code 2LQR sont illustrées dans la Fig. 10.4.

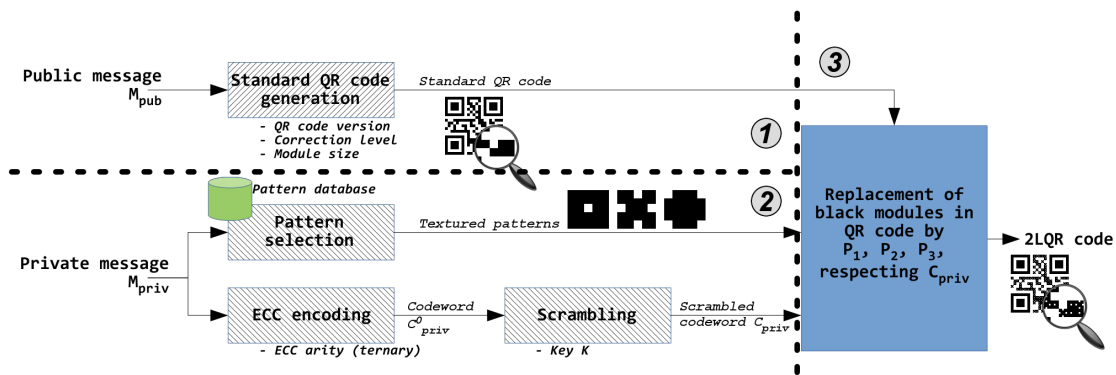


FIGURE 10.4: Les étapes de génération de code 2LQR.

**Le choix des motifs.** Les motifs texturés  $P_i, i = 1, \dots, q$  sont des images de taille  $r \times r$  pixels introduites dans la section 10.3.1. Nous avons choisi  $q$  motifs provenant d'une base de  $Q \gg q$  motifs texturés, qui respectent toutes les caractéristiques mentionnées dans la section 10.3.1 : les motifs sont binaires, ils ont la même densité (rapport de pixels noirs), ils ont des spectres liés et ils respectent les critères (5.2) et (5.3). Les deux

critères (5.2) et (5.3) peuvent être réécrits sous la forme :

$$\begin{aligned} \forall l, l' \in \{1, \dots, q\}, \text{cor}(P_l, S_l) - \max_{l \neq l'}(\text{cor}(P_l, S_{l'})) &\geq \varepsilon, \\ \text{cor}(P_l, S_l) - \max_{l \neq l'}(\text{cor}(P_{l'}, S_l)) &\geq \varepsilon, \end{aligned} \quad (10.5)$$

où  $\varepsilon > 0$ .

La condition (10.5) représente un nouveau critère qui tient compte de la distance minimale acceptable entre le meilleur score de corrélation et le deuxième meilleur score. Cette distance doit être supérieure à un seuil donné  $\varepsilon$ . Par conséquent, seuls les motifs texturés qui respectent la condition (10.5) peuvent être combinés et utilisés pour la génération du code 2LQR.

Expérimentalement nous avons montré que pour l'authentification de documents dans un cadre "de bureau" (imprimantes, copieurs et scanners), le seuil doit être  $\varepsilon < 0.40$ .

**Lecture de code 2LQR.** L'ensemble du processus de lecture de code 2LQR est illustré sur la Fig. 7.4. Tout d'abord, la correction des distorsions géométriques est réalisée lors d'une étape de pré-traitement. Les blocs de position sont localisés par le processus standard [4] pour déterminer les coordonnées des coins. Nous avons ré-échantillonné les codes 2LQR imprimés et scannés avec une interpolation linéaire. A la fin de cette étape, le code 2LQR a la bonne orientation et la taille attendue de  $N \times N$  pixels.

La deuxième étape concerne le classement des modules effectué par une méthode de seuillage. Nous utilisons un seuil global, qui est calculé comme une valeur moyenne de l'ensemble des valeurs de niveaux de gris du code 2LQR. Ensuite, si la valeur moyenne du bloc de  $r \times r$  pixels est plus petite que le seuil global, ce bloc appartient à la classe noire ( $BC$ ), sinon ce bloc appartient à la classe blanche ( $WC$ ). Le résultat de cette étape est donc deux classes de modules.

Dans l'étape suivante, nous complétons le **processus de lecture du QR code standard**, qui en décodant le message public  $M_{pub}$  corrige les erreurs de classification des modules et garantit donc que la classe  $BC$  ne contient que des motifs texturés.

Enfin, nous utilisons la classe  $BC$  pour la **reconnaissance des motifs texturés** après impression et numérisation du code 2LQR. La classe  $BC$  contient les motifs texturés  $BP_i, i = 1, \dots, N_{code} \times n$ , où  $N_{code} \times n$  est le nombre total de caractères de mots de code,  $N_{code}$  est le nombre de mots de code,  $n$  est le nombre de caractères dans le mot de code. Nous avons  $N_{code} \times n$  motifs texturés, qui appartiennent à  $q$  classes. La méthode de détection de motifs proposée, compare les motifs imprimés et numérisés ( $BP_i, i = 1, \dots, N_{code} \times n$ ) avec des motifs représentatifs en utilisant la corrélation de Pearson (5.1).

Plusieurs expérimentations ont été réalisées pour évaluer la méthode proposée de détection des motifs texturés. Ces tests nous permettent de souligner les différences entre les scénarii pour le partage d'un message privé et pour l'authentification du document. Pour cela, nous fixons la dimension de l'alphabet  $q = 3$ , la version du QR code V2, ainsi que la taille des motifs texturés à  $12 \times 12$  pixels. Nous avons fait varier le seuil

$\varepsilon$  du critère (10.5). Dans le tableau 7.15, les taux d'erreurs, d'une part de détection des motifs et d'autre part de décodage des caractères sont présentés pour  $\varepsilon = 0.60$  et  $\varepsilon = 0.25$ . Les résultats de ce tableau montrent que le choix du seuil  $\varepsilon$  doit changer selon le scénario d'application pour le code 2LQR envisagé. Si nous voulons intégrer un grand nombre d'informations, être résistant au processus de copie, sans assurer l'authenticité du code 2LQR, nous devons utiliser un seuil  $\varepsilon \in [0.40, 0.70]$ . Si nous ne voulons pas seulement intégrer un grand nombre d'informations, mais aussi vérifier l'authenticité du code 2LQR, nous devons choisir  $\varepsilon \in (0.20, 0.40)$ . Naturellement, dans le deuxième scénario, nous ne pouvons pas être robustes au processus de copie, car seul le code 2LQR imprimé (une opération d'impression et de numérisation) doit être reconnu comme le code authentique.

L'étude de la capacité de stockage du code 2LQR a été réalisée en fixant arbitrairement deux valeurs : la surface utilisée à  $1.2 \times 1.2 \text{cm}^2$  et la densité des motifs à environ 42%. Nous pouvons augmenter la capacité de stockage du code 2LQR en jouant sur :

- l'augmentation de la valeur  $q$ , qui est le nombre de caractères de l'alphabet et donc de motifs texturés ;
- l'augmentation du nombre de modules (soit une augmentation de la version du QR code et une réduction de la taille des motifs texturés  $r \times r$  pixels).

Dans les deux cas, nous avons constaté une diminution de la fiabilité de la détection des motifs. Cela se traduit par un compromis nécessaire entre la taille des motifs et le nombre de motifs utilisés. Les résultats expérimentaux montrent que pour assurer un processus de lecture correct des niveaux public et privé :

- La taille des motifs doit être supérieure ou égale à  $5 \times 5$  pixels à 600 dpi (tests réalisés jusqu'à  $12 \times 12$  pixels : des tailles supérieures augmentent de manière significative la surface d'impression du QR code).
- La densité des motifs peut varier de 20% à 80%. Cependant il s'avère que le contraste de l'image est trop faible, quand la densité est inférieure à 40%. D'autre part, quand la densité des motifs est supérieure à 70%, la distance entre les valeurs de corrélation devient faible.
- La dimension de l'alphabet  $q$  a été évaluée jusqu'à  $q = 8$ . Des dimensions d'alphabet plus grandes pourraient perturber la détection des motifs texturés en raison de l'impact des processus d'impression et de numérisation.

Plusieurs expérimentations ont été réalisées pour démontrer la robustesse des codes 2LQR contre la duplication non-autorisée. Tous les codes copiés et moins de 2% des codes 2LQR originaux ne passent pas le test d'authentification. Notre objectif est de détecter toutes les reproductions non-autorisées de documents (le taux de faux positifs doit être nul) même si le taux de faux négatifs est non nul.

### 10.3.4 Étude expérimentale du processus d'impression et de numérisation

La dégradation de l'information causée par les processus d'impression et de numérisation est un enjeu majeur dans la criminalistique numérique et la protection de documents imprimés ainsi que dans l'authentification d'images. Cette dégradation est généralement considérée comme étant un processus stochastique qui peut être modélisé par un bruit Gaussien additif ou multiplicatif ou une distribution log-normale [10, 164]. Ces modèles comprennent un bruit aléatoire dû à la dispersion de l'encre sur le papier lors de l'impression ainsi qu'aux conditions d'éclairage pendant la numérisation. Il est généralement admis que ces deux dégradations ne peuvent pas être séparées.

Dans cette section, nous voulons déterminer expérimentalement la nature du bruit de l'imprimante et du scanner, ainsi qu'identifier la distribution théorique des couleurs blanches et noires dans les codes graphiques après les processus d'impression et de numérisation.

Un des objectifs principaux de cette section est de proposer un moyen de caractériser la nature d'un processus stochastique dans le traitement de l'image et notamment de répondre aux questions suivantes :

1. Est-ce que le processus suit une distribution statistique donnée (par exemple, normale ou Laplace) ?
2. Pouvons-nous considérer le bruit comme étant additif ?
3. Pouvons-nous considérer le bruit comme étant stationnaire ?
4. Pouvons-nous considérer le bruit comme étant ergodique ?
5. Pouvons-nous considérer le bruit comme étant blanc ?

Pour répondre à ces questions, nous proposons une série de méthodes statistiques. Nous expérimentons ces méthodes sur une grande base de données contenant des images en noir et blanc de notre code 2LQR proposé. Ces images ont été recueillies après de nombreuses opérations d'impression et de numérisation. Nous avons choisi ces images car elles présentent une structure au contraste très élevé car constituée de pixels noirs et blancs. Tous les tests statistiques présentés dans cette section sont effectués en utilisant des données réelles.

**Bruit du scanner.** Dans la première expérience, nous cherchons à isoler le bruit du scanner. Pour atteindre cet objectif, nous proposons d'imprimer une image une fois, puis de la scanner  $N$  fois. Soit  $I$  l'image originale,  $P$  l'image imprimée de l'image originale  $I$  et  $S_j$  ( $j = 1, \dots, N$ ) les  $N$  échantillons scannés de l'image imprimée  $P$ . Le système peut être décrit comme :

$$I \rightarrow \mathbf{Impression} \rightarrow P \rightarrow \mathbf{Numérisation} \rightarrow S_j. \quad (10.6)$$

Nous considérons  $P$  comme une fonction de  $I$  et de  $\varepsilon_P$ ,  $S_j$  comme une fonction de  $I$ , de  $\varepsilon_P$  et de  $\varepsilon_S$ , donc :

$$P = f(I, \varepsilon_P),$$

$$S_j = g(P, \varepsilon_{S_j}) = P \oplus \varepsilon_{S_j} = f(I, \varepsilon_P) \oplus \varepsilon_{S_j},$$

où  $f()$  est une fonction associée au processus d'impression,  $g()$  est une fonction associée au processus de numérisation et  $\varepsilon_P$  et  $\varepsilon_{S_j}$  sont des bruits introduits par l'imprimante et le scanner, respectivement. L'opérateur  $\oplus$  indique que l'image imprimée et numérisée  $S_j$  peut être représentée par la fonction du processus d'impression et le bruit de numérisation (qui dépend aussi de la fonction du procédé d'impression).

Si le bruit a une régularité, nous pouvons envisager le calcul de la différence entre les échantillons afin d'évaluer la nature du bruit. Par conséquent, nous proposons de calculer les différences entre chacune des  $N$  paires d'échantillons. La soustraction de paires d'images numérisées fournit des échantillons du bruit introduit par le scanner :

$$S_j - S_{j'} = \varepsilon_{S_j - \varepsilon_{S_{j'}}}, \quad (10.7)$$

où  $\varepsilon = \varepsilon_{S_j - \varepsilon_{S_{j'}}$  est le bruit introduit par le processus de numérisation.

Nous présentons maintenant la même configuration expérimentale en utilisant la terminologie des processus aléatoires. Chaque image après impression et numérisation peut être considérée comme une variable aléatoire  $X(t)$ . Par conséquent, l'ensemble des  $N$  images imprimées et scannées peut être considéré comme un processus aléatoire  $X(t, s)$ , où  $s = \{1, \dots, N\}$  est la dimension de la base de données et  $t = \{1, \dots, n\} \times \{1, \dots, n\}$  est le nombre des pixels dans une image  $I$ .

Comme montré précédemment, la fonction de bruit du scanner peut être obtenue en soustrayant les variables aléatoires  $X(t, s_j)$  et  $X(t, s_{j'})$ , afin de créer un nouveau processus aléatoire  $Y(t, s)$  avec  $t = \{1, \dots, N'\}$  :

$$Y(t, s) = X(t, s_j) - X(t, s_{j'}), j \neq j', s_j = s_{j'} = \{1, \dots, N'\}, \quad (10.8)$$

où  $N' = \frac{N(N-1)}{2}$ .

**Description de la base de données.** Nous imprimons un code 2LQR une fois, puis le numérisons  $N = 90$  fois. Chaque image imprimée et scannée a une taille de  $300 \times 300$  pixels, donc  $n = 300$ . Le nombre de soustractions deux à deux est égal à  $N' = 4,005$ . L'étude de ce processus aléatoire nous permet de conclure que :

1. Le bruit du scanner peut être dit plus proche d'une distribution de Laplace, que d'une distribution normale.
2. Le bruit du scanner n'est pas additif.
3. Le processus aléatoire  $Y(t, s)$  du bruit du scanner est stationnaire d'ordre un.
4. Le processus aléatoire  $Y(t, s)$  du bruit du scanner n'est pas ergodique d'ordre un.

5. Le processus aléatoire  $Y(t, s)$  du bruit du scanner n'est pas stationnaire d'ordre deux, donc il n'est pas stationnaire au sens large.
6. Le processus aléatoire  $Y(t, s)$  du bruit du scanner n'est pas un bruit blanc.

**Bruit d'imprimante et de scanner.** Afin de caractériser le bruit global des processus d'impression et de numérisation, nous avons effectué les mêmes expérimentations pour  $N = 30$  images différentes imprimées et numérisées. Les conclusions de ces mesures sont :

1. Le bruit d'imprimante et de scanner peut être dit plus proche d'une distribution de Laplace, que d'une distribution normale.
2. Le bruit d'imprimante et de scanner n'est pas un bruit additif.
3. Le processus aléatoire  $Y(t, s)$  du bruit d'imprimante et de scanner est stationnaire d'ordre un.
4. Le processus aléatoire  $Y(t, s)$  du bruit d'imprimante et de scanner n'est pas ergodique d'ordre un.
5. Le processus aléatoire  $Y(t, s)$  du bruit d'imprimante et de scanner n'est pas stationnaire d'ordre deux, donc il n'est pas stationnaire au sens large.
6. Le processus aléatoire  $Y(t, s)$  du bruit d'imprimante et de scanner n'est pas un bruit blanc.

**Distribution des couleurs après impression et numérisation.** Les auteurs dans [10] montrent expérimentalement que les blocs de pixels noirs/blancs sont liés à des variables aléatoires qui suivent des distributions log-normale asymétriques. En utilisant notre base de données, nous avons décidé de faire la même expérience. Néanmoins, comme dans nos expériences précédentes, nous utilisons les images de code 2LQR, les pixels noirs et blancs sont connus et peuvent être isolés, constituant une carte de vérité associée à chaque code. Nous voulons mentionner, que les résultats présentés ici sont issus d'un travail toujours en cours. Et nous espérons pouvoir poursuivre cette étude des changements de couleur après impression et numérisation.

Nous imprimons et numérisons un code graphique 90 fois. Par conséquent, nous avons 90 échantillons de codes 2LQR imprimés et numérisés. L'histogramme de l'ensemble de ces images est illustré sur la Fig. 10.5.a. En utilisant la carte de vérité, nous séparons les pixels en deux classes : la classe noire et la classe blanche. Nous avons illustré les histogrammes de ces classes dans la Fig. 10.5.b.

Nous avons appliqué le test  $\chi^2$  pour vérifier si ces distributions suivent une loi normale ou log-normale. Le test  $\chi^2$  rejette l'hypothèse que ces distributions sont normales, à un niveau de signification de 0.05. Dans le même temps, il accepte l'hypothèse qu'elles soient log-normale, à un niveau de signification de 0.05.

Ces résultats ne sont que le début de cette étude. Ces expériences doivent être mieux évaluées, en utilisant différents tests statistiques. Ces premiers résultats mettent en évidence plusieurs questions ouvertes :

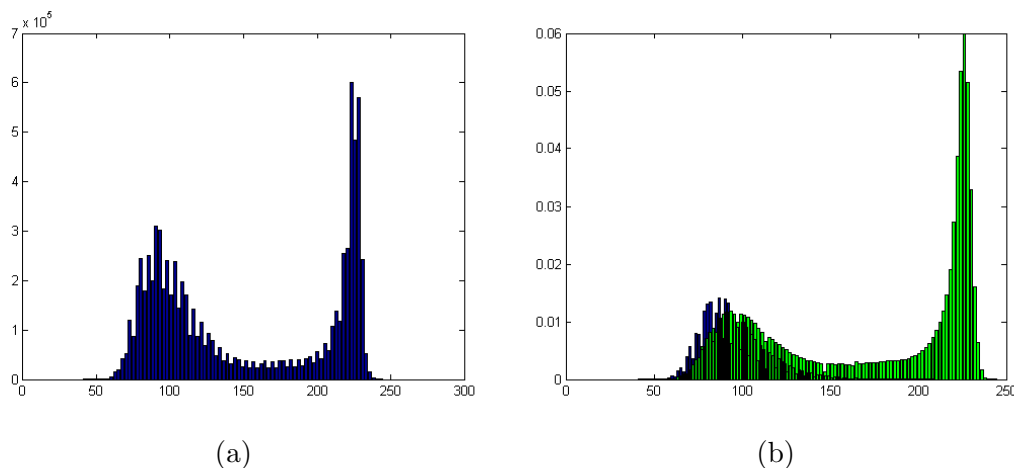


FIGURE 10.5: L’histogramme de 90 échantillons imprimés et numérisés a) Tous les pixels ensemble et b) Pixels séparés dans la classe noire (couleur bleue) et blanche (couleur verte).

- Les histogrammes semblent présenter des artefacts (ressemblant à des effets de quantification). Nous devrions trouver la nature de ces objets pour éventuellement pouvoir les supprimer.
- Les histogrammes bruts semblent présenter une distribution log-normale. Plusieurs tests statistiques supplémentaires doivent être réalisés pour confirmer ou infirmer ces observations.

## 10.4 Conclusion et perspectives

**Conclusion.** La disponibilité des machines de copie de haute qualité ouvre la voie aux contrefaçons de documents imprimés. De nombreuses techniques d’authentification basées sur l’impression sécurisée (codes graphiques, hachage ou hachage local) ont été suggérées. Dans cette thèse, nous nous concentrons sur les codes graphiques sensibles aux processus d’impression, de numérisation et de copie. Les éléments graphiques proposés consistent en des motifs texturés spécifiques, sensibles aux processus de reproduction. Les combinaisons de ces motifs texturés sont séparées en trois classes (combinaisons distinguables, semi-distinguables et non distinguables) en fonction de leur perception par un système de vision humaine. Cette séparation est réalisée à partir de questionnaires présentant différentes combinaisons. La caractérisation des résultats obtenus a été effectuée. Nous avons montré que les motifs texturés avec des spectres liés créent des combinaisons de motifs indiscernables. Nous utilisons les combinaisons semi-distinguables pour le code graphique proposé appelé ”image texturée avec message visuel”. Le choix de ces combinaisons de motifs est fait pour en faciliter la détection après impression et numérisation. Plusieurs méthodes de détection de motifs basées sur la maximisation des valeurs de corrélation ont été suggérées. La méthode, qui maximise la valeur de corrélation de Pearson entre les motifs originaux et les motifs imprimés et numérisés

assure les meilleurs résultats de détection.

Les code-barres de haute densité subissent plusieurs problèmes au cours du processus de lecture en raison des fréquents changements de modules noirs et blancs. Nous proposons de comparer les modules imprimés et numérisés avec leurs versions originales afin d'affiner les résultats de binarisation et d'améliorer la lisibilité des code-barres de haute densité. La mesure de l'erreur quadratique moyenne pondérée (WMSE), qui met en évidence l'importance des pixels centraux du module et diminue l'importance des pixels au bord du module, a été proposée. Les expériences, qui ont été effectuées en utilisant un code-barres de haute densité (QR code version 40, taille des modules  $3 \times 3$  pixels imprimé à 600 dpi), montrent l'efficacité de cette mesure. Cette méthode porte le taux de reconnaissance au delà de 93%, rendant possible l'utilisation du niveau bas de correction d'erreurs, permettant ainsi de stocker plus d'informations.

Les codes graphiques de haute densité ont beaucoup d'applications et un nouveau QR code enrichi, appelé le QR code à deux niveaux (2LQR), a été présenté dans cette thèse. Le code 2LQR suggéré dispose de deux niveaux de stockage : le niveau public lisible par tout lecteur de QR code standard et le niveau privé accessible uniquement pour les utilisateurs autorisés. Le niveau privé est construit en utilisant des motifs texturés. Ce niveau de stockage supplémentaire nous permet non seulement d'augmenter la capacité de stockage initiale du QR code, mais aussi d'en détecter la reproduction non autorisée. La capacité de stockage maximale testée de ce code 2LQR est de  $13934 \text{ bits/pouce}^2$  en utilisant 8 motifs texturés de taille  $6 \times 6$  pixels sur une base de QR code en version 40 : le niveau public contenant  $7548 \text{ bits/pouces}^2$  et le niveau privé  $6386 \text{ bits/pouce}^2$ . De nombreuses expériences de codes 2LQR copiés ont été réalisées à l'aide de quatre photocopieurs différents. Les résultats obtenus montrent la robustesse de notre code 2LQR contre la duplication. Le processus d'enrichissement proposé peut être étendu à tout code-barres standard.

Enfin, les études expérimentales des processus d'impression et de numérisation montrent qu'ils ne peuvent être modélisés comme des processus blanc et ergodique au sens large et qu'ils ne suivent pas une distribution gaussienne. En outre, le bruit du scanner a été séparé du bruit de l'imprimante. Le bruit du scanner semble être stationnaire d'ordre un et impacte moins le bruit total d'impression et de numérisation.

**Perspectives.** Au cours de ce travail, plusieurs perspectives intéressantes ont été déterminées. Ces perspectives peuvent être séparées en trois directions : l'amélioration des motifs texturés, l'amélioration des codes graphiques avec des motifs texturés et l'étude des distorsions ajoutées par l'imprimante et par le scanner.

Amélioration des motifs texturés : Il est intéressant d'utiliser des motifs à niveaux de gris, qui offrent la possibilité d'utiliser le procédé de tramage de l'imprimante et accroît l'effet d'impression aléatoire. De plus, l'utilisation de motifs à niveaux de gris permettra d'augmenter la capacité de stockage des codes graphiques. Il est également possible d'utiliser des motifs texturés avec des densités différentes pour améliorer la protection contre la reconstruction des éléments graphiques et rendre plus complexe l'attaque par



un adversaire. Il est important d'étudier les spectres des motifs et de proposer des critères de combinaison basés sur des mesures différentes de la corrélation. Une direction intéressante est certainement l'utilisation de combinaisons de motifs indiscernables dans les codes graphiques. Cette direction de recherche offre plusieurs études intéressantes prenant en compte le système visuel humain et le processus d'impression.

Amélioration des codes graphiques avec des motifs texturés : Un troisième niveau peut être créé par le remplacement des modules blancs par des motifs texturés de faible densité (10 – 20% des pixels noirs). Même si ce remplacement introduit des perturbations de lecture en raison des changements dans le contraste entre les motifs texturés "presque blancs" et "presque noirs", le gain espéré suggère le développement de cette idée afin de déterminer le meilleur compromis de contraste entre les motifs. Des méthodes de détection des motifs moins sensibles aux rotations et décalages de quelques pixels, doivent être développées. Des attaques "intelligentes" doivent être effectuées pour mieux étudier la robustesse des codes graphiques proposés pour évaluer leur capacité à résister à la reconstruction et à la duplication non-autorisée. En outre, il pourrait être intéressant de combiner la sensibilité à la copie (authentification du support du document) avec des techniques de "tamper-proofing" (authentification du contenu du document) [150]. Pour cela, les caractéristiques des documents, qui ne sont pas sensibles aux changements d'impression et de numérisation, mais sensibles aux changements de documents non autorisés, doivent être suggérées. Cette direction mérite une étude des fonctions de hachage perceptives, de la criminalistique, ainsi que des nombreuses attaques de reconstruction possibles. Un projet R&D doit être mené pour développer les fonctions de lecture des code-barres à deux niveaux proposés dans cette thèse par des dispositifs mobiles (téléphones portables par exemple).

Étude des distorsions ajoutées par l'imprimante et par le scanner : Une étude profonde statistique du processus d'impression et de numérisation doit être effectuée. Nous devons utiliser des méthodes statistiques pour déterminer les caractéristiques de ce processus et pour étudier les changements des pixels noirs et blancs dans les motifs texturés. Enfin, nous devons mener les mêmes expérimentations en utilisant un plus grand nombre d'imprimantes et de scanners.

# List of publications

## Submitted patents:

1. **Iu. Tkachenko**, W. Puech, O.Strauss, J.M. Gaudin, C. Destruel "Code visuel graphique à deux niveaux d'information et sensible à la copie, procédés de génération et de lecture d'un tel code visuel", 2014.

## Journal articles:

1. **Iu. Tkachenko**, W. Puech, C. Destruel, O.Strauss, J.M. Gaudin, C.Guichard "Two level QR code for private message sharing and document authentication", **Transactions on Information Forensics and Security**, accepted for publication.
2. **Iu. Tkachenko**, W. Puech, O.Strauss, J.M. Gaudin, C. Destruel, C.Guichard "Centrality bias measure for high density QR code module recognition", **Signal Processing: Image Communication**, accepted for publication.

## International conference papers:

1. **Iu. Tkachenko**, W. Puech, O.Strauss, C. Destruel "Experimental study of Print-and-Scan impact as Random process", **Electronic Imaging 2016**, February 2016, San Francisco, USA.
2. **Iu. Tkachenko**, W. Puech, O.Strauss, C. Destruel, J.M. Gaudin, C.Guichard "Rich QR code for multimedia management applications", **International Conference on Image Analysis and Processing (ICIAP) 2015**, September 2015, Genova, Italy.
3. **Iu. Tkachenko**, W. Puech, O.Strauss, J.M. Gaudin, C. Destruel, C.Guichard "Improving the module recognition rate of high density QR codes (Version 40) by using centrality bias", **IEEE International Conference on Image Processing Theory, Tools and Applications (IPTA) 2014**, October 2014, Paris, France.
4. **Iu. Tkachenko**, W. Puech, O.Strauss, J.M. Gaudin, C. Destruel, C.Guichard "Fighting against forged documents by using textured image", **European Signal Processing Conference (EUSIPCO) 2014**, September 2014, Lisbon, Portugal.
5. B. Assanovich, W. Puech, **Iu. Tkachenko** "Use of linear error-correcting subcodes in flow watermarking for channels with substitution and deletion errors", **Communications and Multimedia Security (CMS) 2013**, September 2013, Magdeburg, Germany.

**International Conference papers under review:**

1. **Iu. Tkachenko**, W. Puech, O.Strauss, C. Destruel, J.M. Gaudin "Printed document authentication using two level QR code", International Conference on Acoustics, Speech and Signal Processing (ICASSP) 2016, submitted.

**National Conference papers:**

1. **Iu. Tkachenko**, W. Puech, O.Strauss, J.M. Gaudin, C. Destruel, C.Guichard "Statistical Analysis for graphical elements recognition after Print-and-Scan process" (fr), **Compression et Représentation des Signaux Audiovisuels (CORESA) 2013**, November 2013, Le Creusot, France.

# Bibliography

- [1] ISO/IEC 15420:2009. Information technology - Automatic identification and data capture techniques - EAN/UPC bar code symbology specification. 2009.
- [2] ISO/IEC 15438:2006. Information technology – Automatic identification and data capture techniques – PDF417 bar code symbology specification. 2006.
- [3] ISO/IEC 16022:2006. Information technology - Automatic identification and data capture techniques - Data Matrix bar code symbology specification. 2006.
- [4] ISO/IEC 18004:2000. Information technology - Automatic identification and data capture techniques - Bar code symbology - QR Code. 2000.
- [5] ISO/IEC 24778:2008. Information technology - Automatic identification and data capture techniques - Aztec Code bar code symbology specification. 2008.
- [6] S. H. Amiri and M. Jamzad. An algorithm for modeling print and scan operations used for watermarking. In *Digital Watermarking*, pages 254–265. Springer, 2009.
- [7] T. Anan, K. Kuraki, and J. Takahashi. Paper encryption technology. *Fujitsu Sci. Tech. J.*, 46(1):87–94, 2010.
- [8] Z. Baharav and R. Kakarala. Visually significant QR codes: Image blending and statistical analysis. In *Multimedia and Expo (ICME), 2013 IEEE International Conference on*, pages 1–6. IEEE, 2013.
- [9] C. Baras and F. Cayre. 2D bar-codes for authentication: A security approach. In *Signal Processing Conference (EUSIPCO), Proceedings of the 20th European*, pages 1760–1766, 2012.
- [10] C. Baras and F. Cayre. Towards a realistic channel model for security analysis of authentication using graphical codes. In *Information Forensics and Security (WIFS), 2013 IEEE International Workshop on*, pages 115–119. IEEE, 2013.
- [11] P. Bas, J.-M. Chassery, and F. Davoine. Geometrical and frequential watermarking scheme using similarities. In *Electronic Imaging'99*, pages 264–272. International Society for Optics and Photonics, 1999.
- [12] P. Bas, J.-M. Chassery, and B. Macq. Geometrically invariant watermarking using feature points. *Image Processing, IEEE Transactions on*, 11(9):1014–1028, 2002.
- [13] P. Bas, J.-M. Chassery, and B. Macq. Image watermarking: an evolution to content based approaches. *Pattern recognition*, 35(3):545–561, 2002.

- [14] P. Bas, N. Le Bihan, and J.-M. Chassery. Color image watermarking using quaternion fourier transform. In *Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03). 2003 IEEE International Conference on*, volume 3, pages III-521. IEEE, 2003.
- [15] F. Beekhof, S. Voloshynovskiy, M. Diephuis, and F. Farhadzadeh. Physical object authentication with correlated camera noise. In *BTW Workshops*, pages 65-74. Citeseer, 2013.
- [16] M. Bellare and P. Rogaway. Introduction to modern cryptography. *UCSD CSE*, 207:207, 2005.
- [17] S. Bhattacharjee and M. Kutter. Compression tolerant image authentication. In *Image Processing, 1998. ICIP 98. Proceedings. 1998 International Conference on*, volume 1, pages 435-439. IEEE, 1998.
- [18] C. Blundo, A. De Santis, and M. Naor. Visual cryptography for grey level images. *Information Processing Letters*, 75(6):255-259, 2000.
- [19] P. V. K. Borges and J. Mayer. Position based watermarking. In *Image and Signal Processing and Analysis, 2003. ISPA 2003. Proceedings of the 3rd International Symposium on*, volume 2, pages 997-1002. IEEE, 2003.
- [20] T. V. Bui, N. K. Vu, T. T.P. Nguyen, I. Echizen, and T. D. Nguyen. Robust message hiding for QR code. In *Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP), 2014 Tenth International Conference on*, pages 520-523. IEEE, 2014.
- [21] O. Bulan, V. Monga, G. Sharma, and B. Oztan. Data embedding in hardcopy images via halftone-dot orientation modulation. In *Electronic Imaging 2008*, pages 68190C-68190C. International Society for Optics and Photonics, 2008.
- [22] J. Cai. A short survey on visual cryptography schemes. *Department of Computer Science, University of Toronto*, 2004.
- [23] A. Cheddad, J. Condell, K. Curran, and P. Mc Kevitt. A secure and improved self-embedding algorithm to combat digital document forgery. *Signal Processing*, 89(12):2324-2332, 2009.
- [24] A. Cheddad, J. Condell, K. Curran, and P. Mc Kevitt. Digital image steganography: Survey and analysis of current methods. *Signal processing*, 90(3):727-752, 2010.
- [25] G. Chen, Y. Mao, and C. K. Chui. A symmetric image encryption scheme based on 3D chaotic cat maps. *Chaos, Solitons & Fractals*, 21(3):749-761, 2004.
- [26] W.-Y. Chen and J.-W. Wang. Nested image steganography scheme using QR-barcode technique. *Optical Engineering*, 48(5):057004-057004, 2009.
- [27] Y.-H. Chen. QR code having hidden codes and methods of forming and identifying the hidden codes, August 2 2010. US Patent App. 12/848,934.

- [28] P.-J. Chiang, N. Khanna, A. K. Mikkilineni, M. V. O. Segovia, S. Suh, J. P. Allebach, G. T.-C. Chiu, and E. J. Delp. Printer and scanner forensics. *Signal Processing Magazine, IEEE*, 26(2):72–83, 2009.
- [29] C.-H. Chou, W.-H. Lin, and F. Chang. A binarization method with learning-built rules for document images produced by cameras. *Pattern Recognition*, 43(4):1518–1530, 2010.
- [30] C-H. Chu, D-N. Yang, Y-L. Pan, and M-S. Chen. Stabilization and extraction of 2D barcodes for camera phones. *Multimedia systems*, 17(2):113–133, 2011.
- [31] I. Cox, J. Kilian, F. T. Leighton, and T. Shamon. Secure spread spectrum watermarking for multimedia. *Image Processing, IEEE Transactions on*, 6(12):1673–1687, 1997.
- [32] I. Cox, M. Miller, J. Bloom, J. Fridrich, and T. Kalker. *Digital watermarking and steganography*. Morgan Kaufmann, 2007.
- [33] N. Degara-Quintela and F. Perez-Gonzalez. Visible encryption: Using paper as a secure channel. In *Electronic Imaging 2003*, pages 413–422. International Society for Optics and Photonics, 2003.
- [34] J.-F. Delaigle, C. De Vleeschouwer, and B. Macq. Watermarking algorithm based on a human visual model. *Signal processing*, 66(3):319–335, 1998.
- [35] M. L. Diong, P. Bas, C. Pelle, and W. Sawaya. Document authentication using 2D codes: Maximizing the decoding performance using statistical inference. In *Communications and Multimedia Security*, pages 39–54. Springer, 2012.
- [36] A. E. Dirik. Image encryption scheme for print and scan channel. In *International Conference on Applied and Computational Mathematics, ICACM*, 2012.
- [37] A. E. Dirik and B. Haas. Copy detection pattern-based document protection for variable media. *Image Processing, IET*, 6(8):1102–1113, 2012.
- [38] J. Duda, N. J. Gadgil, K. Tahboub, and E. J. Delp. Generalizations of the Kuznetsov-Tsybakov problem for generating image-like 2D barcodes. In *Image Processing (ICIP), 2014 IEEE International Conference on*, pages 4221–4225. IEEE, 2014.
- [39] P. Elias. *List decoding for noisy channels*. Technical Report 335, Research Laboratory of Electronics, MIT, 1957.
- [40] C. Fang, C. Zhang, and E-C. Chang. An optimization model for aesthetic two-dimensional barcodes. In *MultiMedia Modeling*, pages 278–290. Springer, 2014.
- [41] J. Fridrich. Visual hash for oblivious watermarking. In *Electronic Imaging*, pages 286–294. International Society for Optics and Photonics, 2000.
- [42] J. Fridrich. *Steganography in digital media: principles, algorithms, and applications*. Cambridge University Press, 2009.

- [43] J. Fridrich and M. Goljan. Robust hash functions for digital watermarking. In *Information Technology: Coding and Computing, 2000. Proceedings. International Conference on*, pages 178–183. IEEE, 2000.
- [44] J. Fridrich, J. Kodovský, V. Holub, and M. Goljan. Breaking HUGO—the process discovery. In *Information Hiding*, pages 85–101. Springer, 2011.
- [45] J. Fridrich, D. Soukal, and J. Lukáš. Detection of copy-move forgery in digital images. In *in Proceedings of Digital Forensic Research Workshop*. Citeseer, 2003.
- [46] T. Gao and Z. Chen. A new image encryption algorithm based on hyper-chaos. *Physics Letters A*, 372(4):394–400, 2008.
- [47] A. Grillo, A. Lentini, M. Querini, and G. F. Italiano. High capacity colored two dimensional codes. In *Computer Science and Information Technology (IMCSIT), Proceedings of the 2010 International Multiconference on*, pages 709–716. IEEE, 2010.
- [48] Y. Gu and W. Zhang. QR code recognition based on image processing. In *Information Science and Technology (ICIST), 2011 International Conference on*, pages 733–736. IEEE, 2011.
- [49] B. Haas, A. E. Dirik, and Y. Nawaz. Image encryption for print-and-scan channels using pixel position permutation, April 22 2014. US Patent 8,705,736.
- [50] B. Haas, C. Zeller, and Z. Xu. Copy detection system using correlations of copy detection patterns, August 31 2010. US Patent 7,787,152.
- [51] A. Hadmi, A. A. Ouahman, B. A. E. Said, and W. Puech. *Perceptual Image Hashing*. INTECH Open Access Publisher, 2012.
- [52] A. Hadmi, W. Puech, B. A. E. Said, and A. A. Ouahman. A robust and secure perceptual hashing system based on a quantization step analysis. *Signal Processing: Image Communication*, 28(8):929–948, 2013.
- [53] T. Haist and H. J. Tiziani. Optical detection of random features for high security applications. *Optics communications*, 147(1):173–179, 1998.
- [54] T. Hao, R. Zhou, and G. Xing. Cobra: color barcode streaming for smartphone systems. In *Proceedings of the 10th international conference on Mobile systems, applications, and services*, pages 85–98. ACM, 2012.
- [55] D. L. Hecht. Embedded data glyph technology for hardcopy digital documents. In *IS&T/SPIE 1994 International Symposium on Electronic Imaging: Science and Technology*, pages 341–352. International Society for Optics and Photonics, 1994.
- [56] D.L. Hecht. Printed embedded data graphical user interfaces. *Computer*, 34(3):47–55, Mar 2001.
- [57] A. TS. Ho and F. Shu. A print-and-scan resilient digital watermark for card authentication. In *Information, Communications and Signal Processing, 2003 and Fourth Pacific Rim Conference on Multimedia. Proceedings of the 2003 Joint Conference of the Fourth International Conference on*, volume 2, pages 1149–1152. IEEE, 2003.

- [58] J.-U. Hou, H.-U. Jang, and H.-K. Lee. Hue modification estimation using sensor pattern noise. In *Image Processing (ICIP), 2014 IEEE International Conference on*, pages 5287–5291, Oct 2014.
- [59] Y.C. Hou. Visual cryptography for color images. *Pattern Recognition*, 36(7):1619–1629, 2003.
- [60] Y.C. Hou, C.Y. Chang, and F. Lin. Visual cryptography for color images based on color decomposition. In *Proceedings of the Fifth Conference on Information Management, Taipei*, pages 584–591, 1999.
- [61] O. Ibe. *Fundamentals of applied probability and random processes*. Academic Press, 2014.
- [62] S. Ibrahim, M. Afrakhteh, and M. Salleh. Adaptive watermarking for printed document authentication. In *Computer Sciences and Convergence Information Technology (ICCIT), 2010 5th International Conference on*, pages 611–614. IEEE, 2010.
- [63] N. Islam, W. Puech, K. Hayat, and R. Brouzet. Application of homomorphism to secure image sharing. *Optics Communications*, 284(19):4412–4429, 2011.
- [64] D. Jin, W.-Q. Yan, and M. S. Kankanhalli. Progressive color visual cryptography. *Journal of Electronic Imaging*, 14(3):033019–033019, 2005.
- [65] H. Kamada and K. Fujimoto. High-speed, high-accuracy binarization method for recognizing text in images of low spatial resolutions. In *Document Analysis and Recognition, 1999. ICDAR'99. Proceedings of the Fifth International Conference on*, pages 139–142. IEEE, 1999.
- [66] J. Katz and Y. Lindell. *Introduction to modern cryptography*. CRC Press, 2014.
- [67] F. Khelifi and J. Jiang. Perceptual image hashing based on virtual watermark detection. *Image Processing, IEEE Transactions on*, 19(4):981–994, 2010.
- [68] D-G. Kim and H-K. Lee. Color laser printer identification using photographed halftone images. In *Signal Processing Conference (EUSIPCO), 2013 Proceedings of the 22nd European*, pages 795–799. IEEE, 2014.
- [69] J.W. Kim, K.T. Kim, J.S. Lee, and J.U. Choi. Development of visible anti-copy patterns. In *Trust and Privacy in Digital Business*, pages 209–218. Springer, 2004.
- [70] T. D. Kite, B. L. Evans, and A. C. Bovik. Modeling and quality assessment of halftoning by error diffusion. *Image Processing, IEEE Transactions on*, 9(5):909–922, 2000.
- [71] S. Kong. QR code image correction based on corner detection and convex hull algorithm. *Journal of Multimedia*, 8(6):662–668, 2013.
- [72] O. Koval, S. Voloshynovskiy, P. Bas, and F. Cayre. On security threats for robust perceptual hashing. In *IS&T/SPIE Electronic Imaging*, pages 72540H–72540H. International Society for Optics and Photonics, 2009.



- [73] L. Krikor, S. Baba, T. Arif, and Z. Shaaban. Image encryption using DCT and stream cipher. *European Journal of Scientific Research*, 32(1):47–57, 2009.
- [74] C. Kurak and J. McHugh. A cautionary note on image downgrading. In *Computer Security Applications Conference, 1992. Proceedings., Eighth Annual*, pages 153–159. IEEE, 1992.
- [75] T. Langlotz and O. Bimber. Unsynchronized 4D barcodes. In *Advances in Visual Computing*, pages 363–374. Springer, 2007.
- [76] Y. Lei, Y. Wang, and J. Huang. Robust image hash in Radon transform domain for authentication. *Signal Processing: Image Communication*, 26(6):280–288, 2011.
- [77] J.H. Lemelson and J.H. Hiett. Method and apparatus for encoding and decoding bar codes with primary and secondary information and method of using such bar codes, March 7 2000. US Patent 6,032,861.
- [78] S. Lian, J. Sun, and Z. Wang. A novel image encryption scheme based-on JPEG encoding. In *Information Visualisation, 2004. IV 2004. Proceedings. Eighth International Conference on*, pages 217–220. IEEE, 2004.
- [79] C.-C. Lin and W.-H. Tsai. Visual cryptography for gray-level images by dithering techniques. *Pattern Recognition Letters*, 24(1):349–358, 2003.
- [80] C.-Y. Lin and S.-F. Chang. Distortion modeling and invariant extraction for digital image print-and-scan process. In *Int. Symp. Multimedia Information Processing*, 1999.
- [81] H.-J. Lin, C.-W. Wang, and Y.-T. Kao. Fast copy-move forgery detection. *WSEAS Transactions on Signal Processing*, 5(5):188–197, 2009.
- [82] J.-A. Lin and C.-S. Fuh. 2D barcode image decoding. *Mathematical Problems in Engineering*, 2013, 2013.
- [83] P.-Y. Lin, Y.-H. Chen, E. J.-L. Lu, and P.-J. Chen. Secret hiding mechanism using QR barcode. In *Signal-Image Technology & Internet-Based Systems (SITIS), 2013 International Conference on*, pages 22–25. IEEE, 2013.
- [84] Y. Liu and M. Liu. Automatic recognition algorithm of quick response code based on embedded system. In *Intelligent Systems Design and Applications, 2006. ISDA '06. Sixth International Conference on*, volume 2, pages 783–788. IEEE, 2006.
- [85] C.-S. Lu and H.-Y. M. Liao. Structural digital signature for image authentication: an incidental distortion resistant scheme. *Multimedia, IEEE Transactions on*, 5(2):161–173, 2003.
- [86] A. Masmoudi, W. Puech, and M. S. Bouhlel. A generalized continued fraction-based asynchronous stream cipher for image protection. In *Signal Processing Conference, 2009 17th European*, pages 1829–1833. IEEE, 2009.
- [87] A. K. Mikkilineni, G. N. Ali, P.-J. Chiang, G. TC. Chiu, J. P. Allebach, and E. J. Delp. Signature-embedding in printed documents for security and forensic

- applications. In *Electronic Imaging 2004*, pages 455–466. International Society for Optics and Photonics, 2004.
- [88] V. Monga and B. L. Evans. Perceptual image hashing via feature points: performance evaluation and tradeoffs. *Image Processing, IEEE Transactions on*, 15(11):3452–3465, 2006.
- [89] V. Monga and M. K. Mihçak. Robust and secure image hashing via non-negative matrix factorizations. *Information Forensics and Security, IEEE Transactions on*, 2(3):376–390, 2007.
- [90] R. Muniz, L. Junco, and A. Otero. A robust software barcode reader using the Hough transform. In *Information Intelligence and Systems, 1999. Proceedings. 1999 International Conference on*, pages 313–319. IEEE, 1999.
- [91] D. J. Naddor. HD barcode, January 29 2013. US Patent 8,360,333.
- [92] M. Naor and B. Pinkas. Visual authentication and identification. In *Advances in Cryptology—CRYPTO’97*, pages 322–336. Springer, 1997.
- [93] M. Naor and A. Shamir. Visual cryptography. In *Advances in Cryptology—EUROCRYPT’94*, pages 1–12. Springer, 1995.
- [94] Q-T. Nguyen, Y. Delignon, L. Chagas, and F. Septier. Printer identification from micro-metric scale printing. In *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, pages 6236–6239. IEEE, 2014.
- [95] Q-T. Nguyen, Y. Delignon, L. Chagas, and F. Septier. Printer technology authentication from micrometric scan of a single printed dot. In *IS&T/SPIE Electronic Imaging*, pages 90280U–90280U. International Society for Optics and Photonics, 2014.
- [96] W. Oh and W. B. Lindquist. Image thresholding by indicator kriging. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 21(7):590–602, 1999.
- [97] E. Ohbuchi, H. Hanaizumi, and L. A. Hock. Barcode readers using the camera device in mobile phones. In *Cyberworlds, 2004 International Conference on*, pages 260–265. IEEE, 2004.
- [98] N. Otsu. A threshold selection method from gray-level histograms. *Automatica*, 11(285-296):23–27, 1975.
- [99] E. Ouaviani, A. Pavan, M. Bottazzi, E. Brunelli, F. Caselli, and M. Guerrero. A common image processing framework for 2D barcode reading. In *7th International Conference on Image Processing and its Applications*, pages 652–655. IET, 1999.
- [100] D. Parikh and G. Jancke. Localization and segmentation of a 2D high capacity color barcode. In *Applications of Computer Vision, 2008. WACV 2008. IEEE Workshop on*, pages 1–6. IEEE, 2008.
- [101] V. Patidar, G. Purohit, K.K. Sud, and N.K. Pareek. Image encryption through a novel permutation-substitution scheme based on chaotic standard map. In *Chaos-Fractals Theories and Applications (IWCF TA), 2010 International Workshop on*, pages 164–169. IEEE, 2010.

- [102] T. Pavlidis, J. Swartz, and Y. P. Wang. Fundamentals of bar code information theory. *Computer*, 23(4):74–86, 1990.
- [103] T. Pevný, T. Filler, and P. Bas. Using high-dimensional image models to perform highly undetectable steganography. In *Information Hiding*, pages 161–177. Springer, 2010.
- [104] A-T. Phan Ho, B-A. Mai Hoang, W. Sawaya, and P. Bas. Document authentication using graphical codes: impacts of the channel model. In *Proceedings of the first ACM workshop on Information hiding and multimedia security*, pages 87–94. ACM, 2013.
- [105] A-T. Phan Ho, B-A. Mai Hoang, W. Sawaya, and P. Bas. Authentication using graphical codes: Optimisation of the print and scan channels. In *Signal Processing Conference (EUSIPCO), Proceedings of the 22nd European*, pages 800–804, 2014.
- [106] A-T. Phan Ho, B. A. Mai Hoang, W. Sawaya, and P. Bas. Document authentication using graphical codes: Reliable performance analysis and channel optimization. *EURASIP Journal on Information Security*, 2014(1):9, 2014.
- [107] G. K. Phillips. New digital anti-copy/scan and verification technologies. In *Electronic Imaging 2004*, pages 133–141. International Society for Optics and Photonics, 2004.
- [108] J. Picard. Digital authentication with copy-detection patterns. In *Electronic Imaging 2004*, pages 176–183. International Society for Optics and Photonics, 2004.
- [109] J. Picard, Z. Sagan, A. Foucou, and J.P. Massicot. Procédé et dispositif d’authentification de codes géométriques, April 1 2010. WO Patent App. PC-T/FR2009/001,096.
- [110] J. Picard, C. Vielhauer, and N. Thorwirth. Towards fraud-proof id documents using multiple data hiding technologies and biometrics. In *Electronic Imaging 2004*, pages 416–427. International Society for Optics and Photonics, 2004.
- [111] J. Picard and J. Zhao. Techniques for detecting, analyzing, and using visible authentication patterns, May 3 2011. US Patent 7,937,588.
- [112] R. Pizzio. *Hardcopy watermarking for document authentication*. INTECH Open Access Publisher, 2012.
- [113] A. Poljicak, L. Mandic, and D. Agic. Robustness of a DFT based image watermarking method against AM halftoning. *Tehnički vjesnik*, 18(2):161–166, 2011.
- [114] P. Premaratne and C.C. Ko. A novel watermark embedding and detection scheme for images in DFT domain. In *Image Processing and Its Applications, 1999. Seventh International Conference on (Conf. Publ. No. 465)*, volume 2, pages 780–783. IET, 1999.
- [115] P. Premaratne and F. Safaei. 2D barcodes as watermarks in image authentication. In *Computer and Information Science, 2007. ICIS 2007. 6th IEEE/ACIS International Conference on*, pages 432–437. IEEE, 2007.

- [116] N. Provos and P. Honeyman. Hide and seek: An introduction to steganography. *Security & Privacy, IEEE*, 1(3):32–44, 2003.
- [117] W. Puech, A. G. Bors, and J. M. Rodrigues. Protection of colour images by selective encryption. In *Advanced Color Image Processing and Analysis*, pages 397–421. Springer, 2013.
- [118] J. Qin, X. Xiang, Y. Deng, Y. Li, and L. Pan. Steganalysis of highly undetectable steganography using convolution filtering. *Information Technology Journal*, 13(16), 2014.
- [119] M. Querini, A. Grillo, A. Lentini, and G. F. Italiano. 2D color barcodes for mobile phones. *IJCSA*, 8(1):136–155, 2011.
- [120] M. Querini and G. F. Italiano. Facial biometrics for 2D barcodes. In *Computer Science and Information Systems (FedCSIS), 2012 Federated Conference on*, pages 755–762. IEEE, 2012.
- [121] J. M. Rodrigues, W. Puech, and A. G. Bors. Selective encryption of human skin in JPEG images. In *Image Processing, 2006 IEEE International Conference on*, pages 1981–1984. IEEE, 2006.
- [122] N. J. Salkind. *Encyclopedia of measurement and statistics*. Sage Publications, 2006.
- [123] J. Sauvola and M. Pietikäinen. Adaptive document image binarization. *Pattern recognition*, 33(2):225–236, 2000.
- [124] S. M. Seyedzade, S. Mirzakuchaki, and R. E. Atani. A novel image encryption algorithm based on hash function. In *Machine Vision and Image Processing (MVIP), 2010 6th Iranian*, pages 1–6. IEEE, 2010.
- [125] M. Sezgin and B. Sankur. Survey over image thresholding techniques and quantitative performance evaluation. *Journal of Electronic imaging*, 13(1):146–168, 2004.
- [126] S. Shang, N. Memon, and X. Kong. Detecting documents forged by printing and copying. *EURASIP Journal on Advances in Signal Processing*, 2014(1):140, 2014.
- [127] S. Shariati, F.-X. Standaert, L. Jacques, B. Macq, M. A. Salhi, and P. Antoine. Random profiles of laser marks. In *Proceedings of the 31st WIC Symposium on Information Theory in the Benelux*, pages 464–478, 2010.
- [128] M. A. Sharma and M. C. Rao. Visual cryptography authentication for data matrix code. *International Journal of Computer Science and Telecommunications*, 2(8):58–62, 2011.
- [129] D. J. Sheskin. *Handbook of parametric and nonparametric statistical procedures*. crc Press, 2003.
- [130] G. J. Simmons. The prisoners’ problem and the subliminal channel. In *Advances in Cryptology*, pages 51–67. Springer, 1984.

- [131] A. Sinha and K. Singh. A technique for image encryption using digital signature. *Optics communications*, 218(4):229–234, 2003.
- [132] B. Sklar. *Digital communications*, volume 2. Prentice Hall NJ, 2001.
- [133] K. Solanki, U. Madhow, B. S. Manjunath, and S. Chandrasekaran. Modeling the print-scan process for resilient data hiding. In *Electronic Imaging 2005*, pages 418–429. International Society for Optics and Photonics, 2005.
- [134] K. Solanki, U. Madhow, B. S. Manjunath, S. Chandrasekaran, and I. El-Khalil. Print and scan resilient data hiding in images. *Information Forensics and Security, IEEE Transactions on*, 1(4):464–478, 2006.
- [135] P. Subpratsavee and P. Kuacharoen. An implementation of a high capacity 2D barcode. In *Advances in Information Technology*, pages 159–169. Springer, 2012.
- [136] G. E. Suh and S. Devadas. Physical unclonable functions for device authentication and secret key generation. In *Proceedings of the 44th annual Design Automation Conference*, pages 9–14. ACM, 2007.
- [137] A. Sun, Y. Sun, and C. Liu. The QR-code reorganization in illegible snapshots taken by mobile phones. In *Computational Science and its Applications, 2007. ICCSA 2007. International Conference on*, pages 532–538. IEEE, 2007.
- [138] M. Sun, J. Si, and S. Zhang. Research on embedding and extracting methods for digital watermarks applied to QR code images. *New Zealand Journal of Agricultural Research*, 50(5):861–867, 2007.
- [139] I. Szentandrás, A. Herout, and M. Dubská. Fast detection and recognition of QR codes in high-resolution images. In *Proceedings of the 28th Spring Conference on Computer Graphics*, pages 129–136. ACM, 2013.
- [140] J. Tchan. The development of an image analysis system that can detect fraudulent alterations made to printed images. In *Electronic Imaging 2004*, pages 151–159. International Society for Optics and Photonics, 2004.
- [141] J. Tchan, RC. Thompson, and A. Manning. A computational model of print-quality perception. *Expert Systems with Applications*, 17(4):243–256, 1999.
- [142] Iu. Tkachenko, W. Puech, O. Strauss, C. Destruel, J-M. Gaudin, and C. Guichard. Rich QR code for multimedia management applications. In *Image Analysis and Processing (ICIAP) 2015*, pages 383–393. Springer, 2015.
- [143] Iu. Tkachenko, W. Puech, O. Strauss, J.-M. Gaudin, C. Destruel, and C. Guichard. Fighting against forged documents by using textured image. In *Signal Processing Conference (EUSIPCO), Proceedings of the 22th European*, 2014.
- [144] Iu. Tkachenko, W. Puech, O. Strauss, J.-M. Gaudin, C. Destruel, and C. Guichard. Improving the module recognition rate of high density QR codes (version 40) by using centrality bias. In *Image Processing Theory, Tools and Applications (IPTA), 2014 4th International Conference on*, pages 1–6. IEEE, 2014.

- [145] O. D. Trier and A. K. Jain. Goal-directed evaluation of binarization methods. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 17(12):1191–1201, 1995.
- [146] A. Uhl and A. Pommer. *Image and video encryption: from digital rights management to secured personal communication*, volume 15. Springer Science & Business Media, 2005.
- [147] R. Ulichney. *Digital halftoning*. MIT press, 1987.
- [148] R. L. Van Renesse. Hidden and scrambled images: A review. In *Electronic Imaging 2002*, pages 333–348. International Society for Optics and Photonics, 2002.
- [149] R. L. Van Renesse. *Optical document security. Third edition*. Artech House optoelectronics library, 2005.
- [150] R. Villán, S. Voloshynovskiy, O. Koval, F. Deguillaume, and T. Pun. Tamper-proofing of electronic and printed text documents via robust hashing and data-hiding. In *Electronic Imaging 2007*, pages 65051T–65051T. International Society for Optics and Photonics, 2007.
- [151] R. Villán, S. Voloshynovskiy, O. Koval, and T. Pun. Multilevel 2D bar codes: Towards high capacity storage modules for multimedia security and management. *IEEE Transactions on Information Forensics and Security*, 1(4):405–420, December 2006.
- [152] S. Voloshynovskiy, M. Diephuis, F. Beekhof, O. J. Koval, and B. Keel. Towards reproducible results in authentication based on physical non-cloneable functions: The forensic authentication microstructure optical set (FAMOS). In *WIFS*, pages 43–48, 2012.
- [153] S. Voloshynovskiy, O. Koval, F. Deguillaume, and T. Pun. Visual communications with side information via distributed printing channels: extended multimedia and security perspectives. In *Electronic Imaging 2004*, pages 428–445. International Society for Optics and Photonics, 2004.
- [154] S. Voloshynovskiy, O. Koval, R. Villan, E. Topak, J. E. V. Forcén, F. Deguillaume, Y. Rytsar, and T. Pun. Information-theoretic analysis of electronic and printed document authentication. In *Electronic Imaging 2006*, pages 60721D–60721D. International Society for Optics and Photonics, 2006.
- [155] A. Vongkunghae, J. Yi, and R. B. Wells. A printer model using signal processing techniques. *Image Processing, IEEE Transactions on*, 12(7):776–783, 2003.
- [156] S. Vongpradhip and S. Rungraungsilp. QR code using invisible watermarking in frequency domain. In *ICT and Knowledge Engineering (ICT & Knowledge Engineering), 2011 9th International Conference on*, pages 47–52. IEEE, 2012.
- [157] H. Wang and Y. Zou. Camera readable 2D bar codes design and decoding for mobile phones. In *Image Processing, 2006 IEEE International Conference on*, pages 469–472. IEEE, 2006.

- [158] H-C. Wang, J-H. Sung, and Y-H. Chen. Elimination of artifacts in encrypted binary images by modified digital halftoning techniques. In *Optical Security and Counterfeit Deterrence Techniques V*, volume 5310, pages 404–415, 2004.
- [159] M. Warasart and P. Kuacharoen. Paper-based document authentication using digital signature and QR code. In *4TH International Conference on Computer Engineering and Technology (ICCET 2012)*, 2012.
- [160] A. Westfeld. F5—a steganographic algorithm. In *Information hiding*, pages 289–302. Springer, 2001.
- [161] R. R. Wilcox. *Introduction to robust estimation and hypothesis testing*. Academic Press, 2012.
- [162] D. Wu, X. Zhou, and X. Niu. A novel image hash algorithm resistant to print–scan. *Signal processing*, 89(12):2415–2424, 2009.
- [163] W-Q. Yan, D. Jin, and M. S. Kankanhalli. Visual cryptography for print and scan applications. In *Circuits and Systems, 2004. ISCAS'04. Proceedings of the 2004 International Symposium on*, volume 5, pages V–572. IEEE, 2004.
- [164] L. Yu, X. Niu, and S. Sun. Print-and-scan model and the watermarking countermeasure. *Image and Vision Computing*, 23(9):807–814, 2005.
- [165] Z. Yun-Peng, L. Wei, C. Shui-Ping, Z. Zheng-Jun, N. Xuan, and D. Wei-Di. Digital image encryption algorithm based on chaos and improved DES. In *Systems, Man and Cybernetics, 2009. SMC 2009. IEEE International Conference on*, pages 474–479. IEEE, 2009.
- [166] M. Zeghid, M. Machhout, L. Khriji, A. Baganne, and R. Tourki. A modified AES based algorithm for image encryption. *International Journal of Computer Science and Engineering*, 1(1):70–75, 2007.
- [167] L. Zhang, A. Veis, R. Ulichney, and J. Allebach. Binary text image file preprocessing to account for printer dot gain. In *Image Processing (ICIP), 2014 IEEE International Conference on*, pages 2639–2643. IEEE, 2014.
- [168] Y. Zhang, T. Gao, D. Li, and H. Lin. An improved binarization algorithm of QR code image. In *Consumer Electronics, Communications and Networks (CECNet), 2012 2nd International Conference on*, pages 2376–2379. IEEE, 2012.
- [169] J. Zhou, Y. Liu, and P. Li. Research on binarization of QR code image. In *Multimedia Technology (ICMT), 2010 International Conference on*, pages 1–4, Oct 2010.
- [170] B. Zhu, J. Wu, and M. S. Kankanhalli. Print signatures for document authentication. In *Proceedings of the 10th ACM conference on Computer and communications security*, pages 145–154. ACM, 2003.

## Abstract

Due to the development and availability of printing and scanning devices, the number of forged/counterfeited valuable documents and product packages is increasing. Therefore, different security elements (holograms, inks, papers) have been suggested to prevent these illegal actions. In this thesis, we focus on printed security elements that give access to a high security level with an easy implementation and integration. We present how to generate several novel security elements that aim to protect valuable documents and packaging against unauthorized copying process. Moreover, these security elements allow us to store a huge amount of hidden information.

The main characteristic of these security elements is their sensitivity to the print-and-scan process. This sensitivity stems from the use of specific textured patterns. These patterns, which are binary images, have a structure that changes during the printing, scanning and copying processes. We define new specific criteria that ensures the chosen textured patterns to have the appropriate property. The amount of additional information encoded in the patterns increases with the number of patterns used.

Additionally, we propose a new weighted mean squared error measure to improve the robustness of module detection for any high density barcodes. Thanks to this measure, the recognition rate of modules used in standard high density barcodes after print-and-scan process can be significantly increased. Finally, we experimentally study several effects: the physical print-and-scan process, separation of scanner noise from printer noise and changes of colors after print-and-scan process. We conclude, from these experimental results, that the print-and-scan process cannot be considered as being a Gaussian process. It has been also highlighted that this process is neither white nor ergodic in the wide sense.

## Résumé

En raison du développement et de la disponibilité des appareils d'impression et de numérisation, le nombre de documents contrefaits augmente rapidement. En effet, les documents de valeur ainsi que les emballages de produits sont de plus en plus ciblés par des duplications non autorisées. Par conséquent, différents éléments de sécurité (hologrammes, encres, papiers) ont été proposés pour prévenir ces actions illégales. Dans cette thèse, nous nous concentrons sur les éléments de sécurité imprimés qui offrent un haut niveau de sécurité et qui possèdent une mise en œuvre et une intégration simple. Nous présentons comment générer de nouveaux éléments de sécurité qui visent à protéger les documents de valeur et les emballages contre des processus de duplication non autorisés. Ces éléments nous permettent en outre de stocker une grande quantité d'informations cachées.

La caractéristique principale de ces éléments de sécurité est leur sensibilité au processus d'impression et de numérisation. Cette sensibilité est obtenue à l'aide de motifs texturés spécifiques. Ces motifs sont des images binaires qui possèdent une structure sensible aux processus d'impression, de numérisation et de copie. Nous définissons les critères spécifiques qui doivent être respectés lors du choix de ces motifs texturés. La quantité d'information encodée dans l'image augmente avec le nombre de motifs texturés utilisés.

En complément, nous proposons dans ce mémoire d'améliorer la robustesse de la détection des modules, pour tous les codes graphiques, par l'utilisation d'une nouvelle mesure d'erreur quadratique moyenne pondérée. L'utilisation de cette nouvelle mesure nous a permis d'augmenter de façon significative le taux de reconnaissance des modules lorsqu'ils sont utilisés dans des codes à barres standard à haute densité. Enfin, nous étudions expérimentalement plusieurs phénomènes : le processus physique d'impression et de numérisation, la séparation du bruit du scanner de celui de l'imprimante et les changements de couleurs après processus d'impression et de numérisation. Nous concluons à partir de cette étude expérimentale, que le processus d'impression et de numérisation ne peut pas être modélisé comme un loi Gaussienne. Nous mettons en avant que ce processus n'est ni blanc ni ergodique au sens large.