This article has been accepted for publication in IEEE Transactions on Information Forensics and Security. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIFS.2024.3359510

Authentication of copy detection patterns: a pattern reliability based approach

Joakim Tutt, Olga Taran, Roman Chaban, Brian Pulfer, Yury Belousov, Taras Holotyak and Slava Voloshynovskiy
Department of Computer Science, University of Geneva, Switzerland
{joakim.tutt, olga.taran, roman.chaban, brian.pulfer, yury.belousov, taras.holotyak, svolos}@unige.ch

***Abstract*—Copy Detection Pattern (CDP) technology is a promising anti-counterfeiting solution for the protection of physical goods. In recent years, it has been shown that this technology is threatened by powerful deep learning attacks that are able to bypass original authentication schemes. In this paper, we tackle this problem by proposing a new CDP authentication scheme based on statistical knowledge discovered about the printing and imaging process. The novelty of our approach lies in providing means to measure the reliability of each local pattern appearing in the CDP. This allows to define new authentication measures to better differentiate original CDP from fakes. Our results show that this new system is capable of performing reliable CDP authentication with smartphones without the need for heavyweight machine learning tools requiring massive data entries.**

***Index Terms*—copy detection patterns, smartphone authentication, binary pattern-based channel, deep learning fakes.**

I. Introduction

Nowadays, counterfeiting and piracy are among the main challenges for modern economy. Existing methods of anti-counterfeiting are very diverse, ranging from watermarking techniques, special ink, holograms, electronic IDs, etc. The drawbacks of these technologies are that they can be expensive, often proprietary, and usually, authentication is performed in a non-digital way.

A newly promising emerged field in digital anti-counterfeiting technologies is the usage of Printing Unclonable Features (PUF) which are based on intrinsic forensic unclonable features of physical objects, such as randomness of ink blots or paper micro-structures [1]–[3]. Another technology is Copy Detection Patterns (CDP) [4] which are random binary patterns of high entropy printed at the highest possible resolution that are thus difficult to clone. The CDP, in comparison to other technologies, represent an attractive trade-off as they are easily implemented in a production pipeline as a sub-structure of QR-codes and allow for digital authentication [5]. They can be easily integrated into a tracking and tracing distribution framework. The main challenge of this technology today is that, although designed to be robust to common copy attacks when simple decision rules are used based on the similarity to a reference digital template, the CDP face significant difficulties with the advanced machine-learning (ML) copy attacks [6]. The authors in [7] demonstrated a possibility to use powerful deep classifiers in two-class classification set up and showed that it allows one to reliably distinguish original CDP from fakes, if the fakes used at testing time match the statistics of those used during training. However, in the case of mismatches, the method fails to distinguish originals and fakes. In practice, the situation is further complicated by several factors:

- the high deviations in printing and imaging leading to large intra-class variabilities;
- ML estimation attacks that are able to produce digital template estimations based on their printed counterpart with an accuracy score as high as 94% [6];
- the natural lack of exact prior knowledge for the authenticator about the fakes in field. Fakes can be produced in multiple ways and it is unknown which fake will be used at the attacking time;
- the absence of a reliable model of printing-imaging channel that complicates the design of optimal authentication rules.

These problems might be partially addressed by changing from a digital template reference to a physical registered template. Although this technique has proven to be satisfactory for the detection of forged copies, it is unrealistic in practice as it requires a prior enrollment of every printed CDP sent to

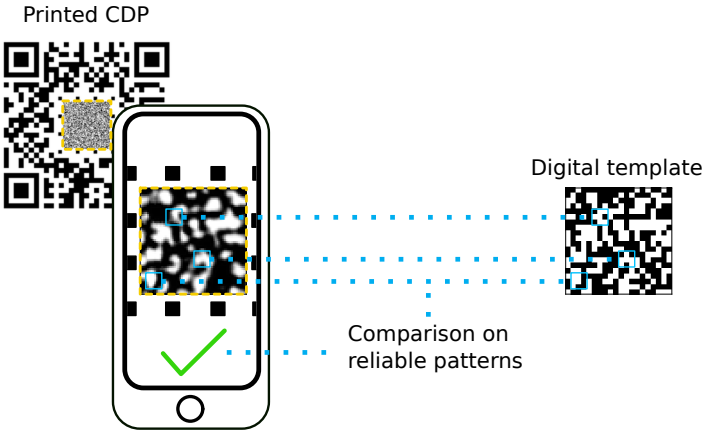


Fig. 1: Authentication of a CDP performed by smartphone. The CDP is embedded as a sub-structure of a QR code as described in [5]. The proposed approach offers the possibility to select only reliable patterns (in blue) to perform the comparison with a reference digital template stored on a remote server. A decision is then made based on the measured patterns.

S. Voloshynovskiy is a corresponding author.

This research was partially funded by the Swiss National Science Foundation SNF No. 200021_182063.

This article has been accepted for publication in IEEE Transactions on Information Forensics and Security. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIFS.2024.3359510

the public domain and so it is not scalable and costly.

In [8], the authors proposed a generative deep-learning model producing synthetic physical references, mimicking the behaviour of real physical references. The results they obtain are very promising in terms of ML fake detection but this model requires heavy computations to be implemented in practice, lacks interpretability and thus gives no control on the stability of the fake detection. In [9], the authors perform an in-depth study of such generative models, trained in paired or unpaired setups and compare the outcomes with real acquired CDP. However, the interpretability of the results produced and execution on mobile devices remain open issues for this approach.

In view of these shortcomings, we present in this paper an authentication scheme relying on a theoretical binary pattern-based channel model. This paper is a further extension of our conference paper [10]. In addition to our previous work where we addressed the following problems:

- proposed a new method of authentication based on digital templates able to perform authentication under complete ignorance about the actual fakes;
- introduced a new measure of similarity for printed CDP which outperforms the standard metrics used before;
- introduced a measure of reliability for printed patterns, giving the defender the ability to select only the parts of the CDP which are reliable for authentication (see Fig. 1 for an illustration);
- performed reliable authentication based on high-resolution scanner in the face of very strong ML attacks;

in this work we extend our study to:

- address a challenging problem of reliable authentication of CDP based on smartphone devices against ML attacks. The authentication of CDP against ML attacks on mobile devices is a considerably more complex task in comparison to those based on high-resolution scanners;
- reformulate the mathematical model in terms of channel reliability, explicitly linking it to the stationary binary symmetric channels from information theory [11], [12] and previous related works addressing bit reliability in identification and retrieval applications [13], [14];
- perform new experiments that give better understanding of the link between the concept of pattern reliability and authentication performance enhancement;
- consider several additional measures of authentication system performance.

The paper is organized as follows: Sections II and III introduce the classical framework of CDP-based authentication schemes and present the pattern-based channel model and its properties that form the theoretical foundations on which the new framework is built. Section IV describes three algorithms that form the new authentication scheme. Section V and VI describe the experiments that were performed on a real dataset of CDP enrolled with smartphones and discuss the results obtained. The final section VII concludes the paper and considers possible extensions and perspectives. All mathematical notations used in the paper can be found in Table I.

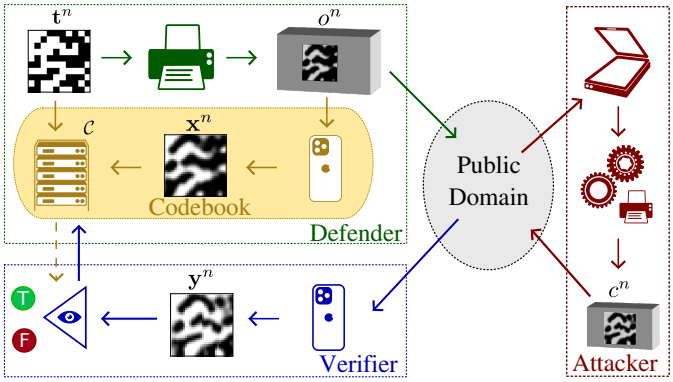


Fig. 2: Illustration of the CDP framework seen as a 3-player game. The Defender (in green) generates and prints template $\mathbf{t}^n$ on the surface of an object o^n and sends it to the public domain. The Attacker (in red) can use o^n to create a counterfeited version c^n of it. Finally, the Verifier (in blue) scans the object and authentifies the probe $\mathbf{y}^n$ to decide whether it is an original or a counterfeit. The novelty of our model consists in adding a codebook (in yellow) which is trained by the Defender and used by the Verifier to enhance the authentication accuracy.

II. Classical CDP framework

A. The printing-authentication scheme

The anti-counterfeit technology based on CDP can be described as a 3-player game with a Defender, an Attacker and a Verifier as shown in Fig. 2.

The Defender protects his brand by using a family of digital CDP templates $\{\mathbf{t}^n\}_{n=1}^N$ stored in the form of a binary matrix $\mathbf{t}^n$, which is then printed on the surface of an object o^n and sent to the public domain. The Attacker has access to the printed version of the CDP and may use it to create a counterfeit c^n, through the process of scanning, post-processing and reprinting (see [6], [15]–[17] for investigations of attacking techniques). At the authentication stage, the Verifier receives an unknown package (either o^n or c^n) from which a digital image $\mathbf{y}^n$ is acquired, using a mobile phone. We denote $\mathbf{x}^n$ the code acquired from o^n and $\mathbf{f}^n$ the code acquired from c^n. An authentication is then performed based on the probe $\mathbf{y}^n$, which might be either $\mathbf{x}^n$ or $\mathbf{f}^n$, with respect to the reference digital template $\mathbf{t}^n$.

The information about the reference digital template $\mathbf{t}^n$, which should be used for the authentication by the Verifier, can be obtained in various ways. For instance, the surrounding QR-code shown in Fig. 1 may contain a pointer to the secured dataset and then, via a secured channel, the corresponding template is communicated to the Verifier who checks its authenticity in a secured environment. Otherwise, the probe can be sent to the secured and trusted server via the acquisition application and the authentication performed on the server. (see the blue and yellow arrows between the Verifier and the Defender in Fig. 2)

This article has been accepted for publication in IEEE Transactions on Information Forensics and Security. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIFS.2024.3359510

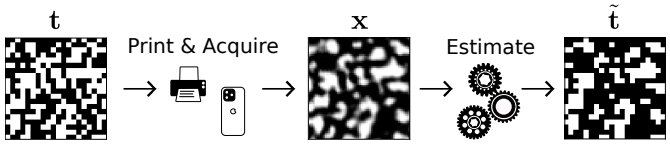


Fig. 3: A visual representation of the printing-imaging-estimating process. Notice how the dot-gain effect creates dependencies on neighbouring symbols.

B. Authentication techniques

The algorithms used for authentication evolved a lot in the last few years. At first, CDP were designed with an idea to be resistant to simple scanning and reprinting attacks [4]. Due to the dot gain effect of printers, a portion of the information stored in the template $\mathbf{t}$ is lost in the probe $\mathbf{y}$ through the random process of printing and imaging. Various ways to measure the information loss have been proposed which can be formalized with different types of metrics:

1) ℓ_1- or ℓ_2-distance between the probe $\mathbf{y}$ and the template $\mathbf{t}$;
2) Pearson correlation between $\mathbf{t}$ and $\mathbf{y}$;
3) Hamming distance between the template $\mathbf{t}$ and an estimation $\tilde{\mathbf{t}}$ of the template, based on the probe $\mathbf{y}$.[1] [18].

These scores are then compared with a decision threshold γ to decide whether the probe $\mathbf{y}$ is genuine or fake. Nowadays, new techniques emerge with the use of machine learning, allowing one to train deep classifiers [7], [19] and deep binarization techniques [6], [15]–[17]. Although showing very promising results, these new algorithms act as black boxes and thus lack interpretability, which is paramount when working on reliability questions and security-critical applications such as the protection of pharmaceutical products.

III. Proposed model

A. Markov chain and communication channels

The proposed mathematical model describes the process of printing and imaging as a Markov Chain $\mathbf{T} \to \mathbf{X} \to \tilde{\mathbf{T}}$, where:

- $\mathbf{T}$ is a random binary matrix of size $L \times L$ sampled from i.i.d. Bernoulli distribution: $T_{ij} \sim Bern(p)$, $p \in [0,1]$ is the probability of white symbol;
- $\mathbf{X}$ is a random matrix of size $kL \times kL$, $X_{ij} \in [0,1]$ for some magnification factor[2] $k = 1, 2, 3, ...$;
- $\tilde{\mathbf{T}}$ is a random binary matrix of size $L \times L$.

The core idea of the model relies on information theory, where each symbol is interpreted as a communication channel $T_{ij} \mapsto \tilde{T}_{ij}$. Thus a very natural question arises about how reliable each channel is, which we measure in terms of posterior probability of bit-flip:

[1]In real situations, the size and orientation of the probe $\mathbf{y}$ and the digital template $\mathbf{t}$ might be different due to the complex printing and imaging processes. To proceed with the estimation of $\tilde{\mathbf{t}}$ from $\mathbf{y}$, the image $\mathbf{y}$ is first geometrically aligned with $\mathbf{t}$ using common template matching techniques. In our case, we have used the synchronization markers around the printed codes to align $\mathbf{y}$ and $\mathbf{t}$. After the geometrical alignment, the binarization based on Otsu's algorithm is applied to the aligned image. As a result, the image $\mathbf{y}$ is transformed to a binary counterpart $\tilde{\mathbf{t}}$ aligned in size and orientation to $\mathbf{t}$.

[2]The magnification factor is related to the resolution of enrollment equipment. Nowadays, with modern scanners and mobile phones, $k \geq 1$.

TABLE I: Mathematical notations used in the paper.

	Mathematical notation	Meaning
CDP	$\mathbf{t}$	binary digital template
	$\mathbf{x}$	digital original printed from $\mathbf{t}$
	$\mathbf{f}$	digital fake version of $\mathbf{t}$
	$\mathbf{y}$	probe representing either $\mathbf{x}$ or $\mathbf{f}$
	$\tilde{\mathbf{t}}$	digital template estimated from $\mathbf{y}$
BPC	$\mathbf{T}$	binary random matrix for $\mathbf{t}$
	$\mathbf{X}$	random matrix for $\mathbf{x}$
	$\tilde{\mathbf{T}}$	binary random matrix for $\tilde{\mathbf{t}}$
	$p \in [0,1]$	probability of white symbol in $\mathbf{T}$
	$\omega \in \Omega$	set of all patterns
	$P_b(\omega)$	probability of bit-flipping at ω
	$\mathcal{C}$	codebook of probabilities
	$\mathbb{P}$	probability measure for $(\mathbf{T}, \mathbf{X}, \tilde{\mathbf{T}})$
Numbers	$n = 1, ..., N$	index within the dataset
	(i,j) or (r,s)	coordinates of pixels in $\mathbf{t}$
	$L \times L$	size of $\mathbf{t}$
	$h = 1, 3, 5, ...$	integer defining the size of ω
	$k = 1, 2, 3, ...$	magnification factor from $\mathbf{t}$ to $\mathbf{x}$
	$\mu \in [0,1]$	threshold used for reliable patterns
	$\gamma \in \mathbb{R}$	decision threshold for measures

$$P_b(\mathbf{T}) = \mathbb{P}(T_{ij} \neq \tilde{T}_{ij} | \mathbf{T}) \quad \forall i, j. \tag{1}$$

The underlying real process this model is trying to describe is the following: when a digital template $\mathbf{t}$ is being printed and acquired as an image $\mathbf{x}$, some distortions occur in $\mathbf{x}$ due to the dot-gain effect, printing-related natural randomness and acquisition conditions. Thus, when one tries to estimate $\tilde{\mathbf{t}}$ from $\mathbf{x}$, it will end up with some errors, dependant on the printer, type of paper, acquisition device, conditions and chosen estimator. Fig. 3 illustrates this process.

A first model describing this phenomenon was studied in [12]. In this article, the authors model the probability of bit-flip as a Binary Symmetric Channel (BSC):

$$\bar{P}_b = \mathbb{P}(\tilde{T}_{ij} \neq T_{ij}) \quad \forall i, j. \tag{2}$$

This model assumes that all symbols T_{ij} of $\mathbf{T}$ have the same average probability of bit-flip $\bar{P}_b$. Although being very common in information theory, this model is too restrictive to be able to capture the random dependencies of the printing-imaging process. Indeed, the BSC model imposes the following assumptions:

1) *Symmetry:* the probability of bit-flipping from a black symbol to a white symbol is the same as from a white symbol to a black one;
2) *Stationarity and independence:* all channels are independent and identically distributed (iid), regardless of their particular location (i,j).

These assumptions are very strong and do not match the real behaviour of the printing-imaging process. Indeed, when looking at Fig. 3, one easily notices that the dot-gain effect

This article has been accepted for publication in IEEE Transactions on Information Forensics and Security. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIFS.2024.3359510

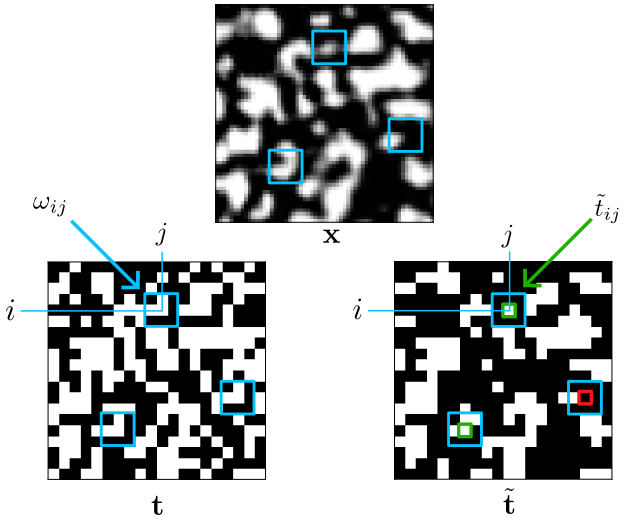


Fig. 4: An illustration of the BPC model. The pattern ω_{ij} (in blue) appears three times in the template $\mathbf{t}$. This same pattern undergoes various random deformations in $\mathbf{x}$ leading to correct and incorrect symbol estimates in $\tilde{\mathbf{t}}$ (see the green and red symbols in $\tilde{\mathbf{t}}$. Measuring the average behaviour of the estimated symbol $\tilde{t}_{ij}$ is a relevant measure of the reliability of the pattern.

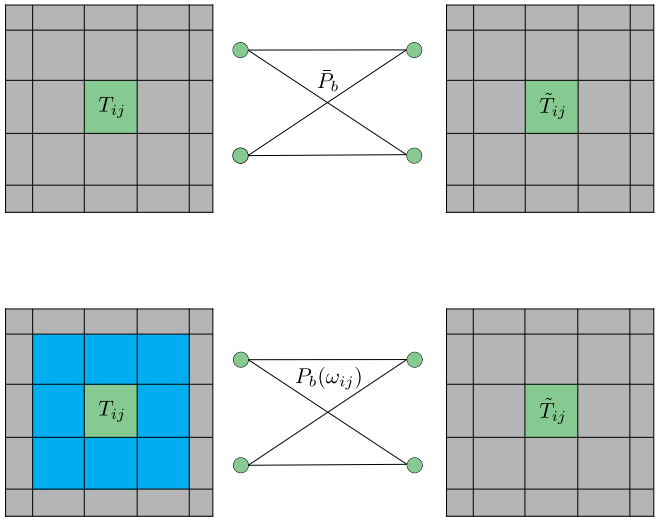


Fig. 5: A visual comparison of the BSC model (on top) and the BPC model (on bottom). In the BSC model, each symbol T_{ij} has the same probability of bit-flip $\bar{P}_b$ independent of the neighbouring patterns (in gray). In the BPC model, the probability of bit-flip $P_b(\omega_{ij})$ depends on the local pattern ω_{ij} (in blue) surrounding the symbol T_{ij}.

breaks the symmetry, with black symbols having a lower probability of bit-flip than white symbols. The random deviations of the printing process creates a lot of local dependencies on neighbouring symbols. White symbols surrounded by black symbols tend to flip more easily than those surrounded by white pixels. As such, independency is not a realistic assumption as well. Stationarity however, if carefuly stated, should be an expected behaviour, as there is no physical reason for the printing process to differ from one location to another on the paper. Another related model with multilevel symbols has been studied in [20].

B. Proposed Binary Pattern-Based Channel model

Based on the limitations of the standard BSC model, we introduce the Binary Pattern-based Channel (BPC) model, a new stochastic model aimed at better describing the local dependencies of the printing-imaging process. To this end, we introduce the notion of pattern:

Definition 1. *A pattern $\omega_{ij} \subset \mathbf{T}$, is a small neighbourhood surrounding symbol T_{ij} in $\mathbf{T}$. The set of all possible patterns is denoted by Ω.*

In practice, we will restrict our study to square patterns centered around (i, j):

$$\omega_{ij} = \{T_{i\pm a, j\pm b} | 0 \leq a, b < h/2\},$$

where $h = 1, 3, 5, \ldots$ is fixed and describes the size of the square.

We now introduce the assumptions of the proposed model:

1) *Locality:* the posterior probability of $\tilde{T}_{ij}$ at a particular symbol location (i, j) only depends on the local pattern ω_{ij} surrounding it:
$$\mathbb{P}(\tilde{T}_{ij}|\mathbf{T}) = \mathbb{P}(\tilde{T}_{ij}|\omega_{ij}). \tag{3}$$
2) *Stationarity:* the posterior probability does not depend on the location inside the image. Similar patterns in $\mathbf{T}$ lead to similar probability values:[3]
$$\mathbb{P}(\tilde{T}_{ij}|\omega_{ij}) = \mathbb{P}(\tilde{T}_{rs}|\omega_{rs}), \text{ if } \omega_{ij} = \omega_{rs}. \tag{4}$$
3) *Posterior independance:* the joint posterior probability factorizes as:
$$\mathbb{P}(\tilde{\mathbf{T}}|\mathbf{T}) = \prod_{i,j} \mathbb{P}(\tilde{T}_{ij}|\mathbf{T}). \tag{5}$$

These assumptions are illustrated in Fig. 4, where a similar pattern ω in $\mathbf{t}$ leads to various outcomes in the estimated template $\tilde{\mathbf{t}}$. Fig. 5 illustrates the fundamental difference between the BSC model and the BPC model.

C. Concept of pattern reliability

With assumptions (3) and (4), one can easily prove the expectation formula for the posterior distribution:

$$\mathbb{P}(\tilde{T}_{ij}|\omega_{ij}) = \mathbb{E}_{r,s:\omega_{rs}=\omega_{ij}}[\mathbb{P}(\tilde{T}_{rs}|\omega_{rs})]. \tag{6}$$

This formula is a key to the proposed authentication scheme as it can be estimated directly using statistical inference on a

[3]The printing and imaging process introduces a lot of variability. The goal of the model is not to learn the fingerprint of a particular realization but rather measure the average variability for each neighbourhood and to take advantage of this knowledge. (4) should be read as an equality in distribution, allowing every realisation of $\tilde{T}_{i,j}$ to be different while still following a common law, independent of the location (i, j).

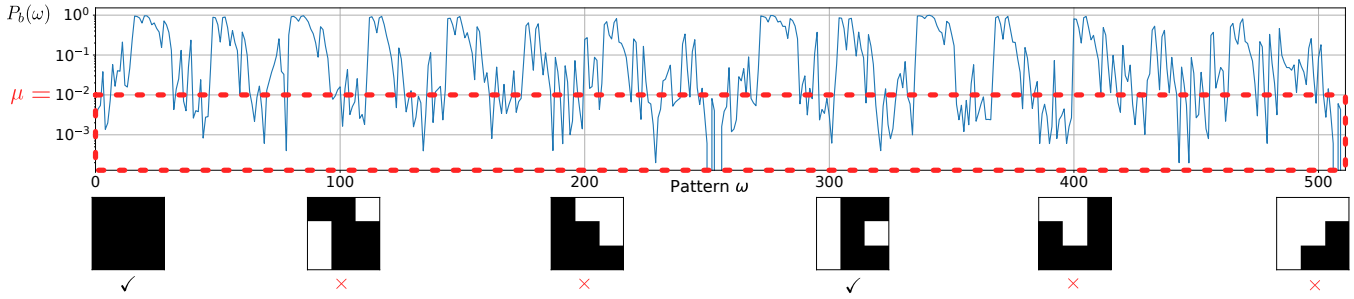


Fig. 6: Visualisation of a codebook for 3×3 square patterns ω. The x-axis represents the 512 different possible patterns ordered by their flattened binary representations. The y-axis represents the probability of bit-flipping for the central pixel of each pattern. As can be observed, some patterns almost certainly flip with $P_b(\omega)$ close to 1, whereas others produce reliable results with $P_b(\omega)$ close to 0. The red box indicates a selection of reliable patterns which have a probability of flipping less than a chosen threshold μ.

training dataset. For each type of pattern $\omega \in \Omega$ (there can be at most 2^{h^2} such patterns), we learn the probability distribution which is highly related to the printing-imaging process on which it was trained. The natural measure associated with this distribution is the "probability of bit-flipping of the central symbol in pattern ω" which we define as:

$$P_b(\omega) := \mathbb{E}_{i,j:\omega_{ij}=\omega}[\mathbb{P}(\tilde{T}_{ij} \neq T_{ij}|\omega_{ij})] \tag{7}$$

$$= \frac{1}{N_\omega} \sum_{i,j:\omega_{ij}=\omega} 1_{\{\tilde{t}_{ij} \neq t_{ij}\}}, \tag{8}$$

where $1_{\{condition\}}$ denotes an indicator function, which is equal to 1 when *condition* is true and 0 otherwise.

The last equality is a statistical estimator of the conditional expectation where N_ω represents the number of times the pattern ω appears in $\mathbf{T}$. We can thus create a codebook in which we store all these different probability values for each type of neighbourhood and use them as references in the authentication scheme.

D. Proposed metrics

The BPC model described above gives us a theoretical tool to better understanding the process of printing and acquisition of CDP. In this subsection, we show that this model comes with a very natural metric that can be easily implemented and used for authentication. Indeed, given an estimated template $\tilde{\mathbf{t}}$, one can ask how likely it is that the BPC model produced such an outcome. The answer is given by the following lemma:

Lemma 1. *In the BPC model, the posterior log-likelihood (PLL) can be computed as:*

$$\log \mathbb{P}(\tilde{\mathbf{T}} = \tilde{\mathbf{t}}|\mathbf{T}) = \sum_{i,j} \log \pi_{ij}, \tag{9}$$

where,

$$\pi_{ij} = \begin{cases} 1 - P_b(\omega_{ij}) & \text{if } \tilde{t}_{ij} = t_{ij} \\ P_b(\omega_{ij}) & \text{if } \tilde{t}_{ij} \neq t_{ij}. \end{cases}$$

Proof. The proof relies on two steps. The first one is to use posterior independence of the symbols in $\tilde{\mathbf{T}}$ given $\mathbf{T}$ (5) and locality (3):

$$\log \mathbb{P}(\tilde{\mathbf{T}} = \tilde{\mathbf{t}}|\mathbf{T}) = \sum_{i,j} \log \mathbb{P}(\tilde{T}_{ij} = \tilde{t}_{ij}|\mathbf{T})$$

$$= \sum_{i,j} \log \mathbb{P}(\tilde{T}_{ij} = \tilde{t}_{ij}|\omega_{ij}).$$

The last step is a simple case study for $t_{ij}, \tilde{t}_{ij} \in \{0, 1\}$:

$$\mathbb{P}(\tilde{T}_{ij} = \tilde{t}_{ij}|\omega_{ij}) = \begin{cases} 1 - P_b(\omega_{ij}) & \text{if } \tilde{t}_{ij} = t_{ij} \\ P_b(\omega_{ij}) & \text{if } \tilde{t}_{ij} \neq t_{ij}. \end{cases}$$

■

IV. Proposed Authentication Algorithms

The core idea of building an authentication system based on the BPC model relies on the construction of *the codebook* $\mathcal{C}$. We use a database $\{(\mathbf{t}^n, \mathbf{x}^n)\}_{n=1}^{N_{train}}$ consisting of pairs of digital templates and printed codes to learn the codebook and compute an estimated probability of bit-flipping $P_b(\omega)$ for the central pixel of each pattern $\omega \in \Omega$ (see Fig. 6).

In more details, we create a dictionary $\mathbb{D}_b$ whose keys are the different types of neighbourhoods. For each $\omega_{ij} \in \Omega$, $\mathbb{D}_b[\omega_{ij}]$ lists the boolean values $(\tilde{t}_{ij} \neq t_{ij})$. Finally, we compute the statistics $P_b(\omega)$ for each type of pattern ω by averaging the corresponding list. A pseudo-code is given in Algorithm 1.

The strength of this scheme lies in the fact that the Defender only needs to run this algorithm on a small subset $N_{train} \ll N$ of the total number of printed objects (see section V-B for a detailed discussion of this matter), and thus does not need to enroll all printed CDP before sending them to the public domain. This step corresponds to the codebook in Fig. 2.

We now consider several authentication schemes based on the BPC model and on the PLL measure.

A. The posterior log-likelihood measure

The first authentication scheme is a direct implementation of (9). It starts by learning the codebook $\mathcal{C}$, running Algorithm 1 on the training set. At the authentication stage, given a probe $\mathbf{y}$, we perform the following steps:

Algorithm 1 Algorithm for codebook estimation

Input: training set $\{(\mathbf{t}^n, \mathbf{x}^n)\}_{n=1}^{N_{train}}$
Output: estimated codebook $\mathcal{C} = (\omega, P_b(\omega))_{\omega \in \Omega}$
Initialisation:
create a dictionary $\mathbb{D}_b$ with the set Ω as keys and empty lists as values.
for $n = 1$ to N_{train} **do**
  estimate $\tilde{\mathbf{t}}^n$ from $\mathbf{x}^n$
  for symbol t_{ij}^n in $\mathbf{t}^n$ **do**
    extract pattern ω_{ij}^n in $\mathbf{t}^n$
    extract symbol $\tilde{t}_{ij}^n$ in $\tilde{\mathbf{t}}^n$
    append boolean value $(\tilde{t}_{ij}^n \neq t_{ij}^n)$ in dictionary $\mathbb{D}_b$ at key ω_{ij}^n
  end for
end for
for ω in Ω **do**
  compute mean value: $P_b(\omega) = \text{mean}(\mathbb{D}_b[\omega])$
  store the couple $(\omega, P_b(\omega))$
end for
return codebook $\mathcal{C} = (\omega, P_b(\omega))_{\omega \in \Omega}$

1) estimate $\tilde{\mathbf{t}}$ from the probe $\mathbf{y}$;
2) with the reference template $\mathbf{t}$, the estimated $\tilde{\mathbf{t}}$ and the codebook $\mathcal{C}$, compute the posterior log-likelihood of $\tilde{\mathbf{t}}$ applying (9);
3) compare the score with a chosen threshold γ fixed on a validation set to decide whether $\mathbf{y}$ is original or fake.[4]

It should be pointed out here that symbols t_{ij} located too close to the border of the template do not have a well-defined neighbourhood ω_{ij}. We propose two solutions to address this problem:

- the first solution is simply to ignore these symbols and run the model only on the symbols located on the inside of $\mathbf{t}$;
- another solution is to consider a white padding surrounding template $\mathbf{t}$ as this is the natural padding for $\mathbf{x}$ when printing CDP on white paper.

B. The reliable patterns selection technique

The idea behind the reliable patterns selection technique is to use the codebook of bit-error $P_b(\omega)$ to select only the patterns $\omega_{ij} \subset \mathbf{t}$ that have a low probability of bit-error (see the red box in Fig. 6). In this way, we discard all regions in $\mathbf{y}$ that are known to produce high error for original samples $\mathbf{x}$. The authentication steps are:

1) for each neighbourhood ω_{ij} in $\mathbf{t}$, search the probability of bit-flipping $P_b(\omega_{ij})$ in the codebook;
2) define an attention mask $m_{ij} := (P_b(\omega_{ij}) \leq \mu)$ for some fixed threshold $\mu \in [0, 1]$;
3) choose any standard metric that is computed pixel-wise such as mean squared error, Hamming distance or Pearson correlation. Note that some upsampling of $\mathbf{t}$ might be necessary for computation;

[4]If no fake samples are available in the validation set, one can simply fix the threshold at the lowest score of the posterior log-likelihood.

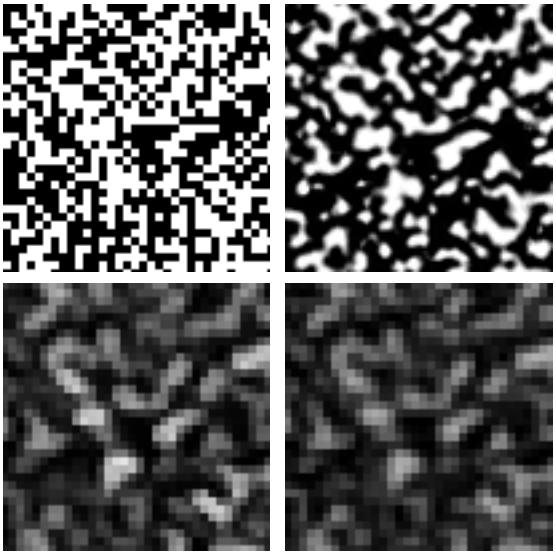


Fig. 7: Visualisation of the Indigo 1×1 base smartphone dataset. top-left corner is a digital template $\mathbf{t}$, top-right corner the printed CDP acquired by scanner, bottom-left the same CDP captured by iPhone 12 Pro, bottom-right the same CDP captured by Samsung Galaxy Note 20 Ultra.

4) weight the chosen metric $d(\mathbf{t}, \mathbf{y})$ by using the binary mask, upsampling it if needed:

$$d^m(\mathbf{t}, \mathbf{y}) = \frac{1}{M} \sum_{i,j} m_{ij} \cdot d(t_{ij}, y_{ij}), M = \sum_{i,j} m_{ij}. \quad (10)$$

V. Experimental setup

A. Mobile Phone Dataset

For our experiments, we use the Indigo 1×1 base smartphone dataset that was created in our project[5] (see Fig. 7 for an illustration). This dataset consists of 1440 unique digital templates $\mathbf{t}$ printed on an industrial printer *HP Indigo 5500 DS* at 812.8 dpi, and enrolled with two different smartphones: An *iPhone 12 Pro* and a *Samsung Galaxy Note 20 Ultra*. It also includes ML-based fakes obtained from originals by the process of scanning with an *Epson Perfection V850 Pro* at 2400dpi, deepnet-based binarization, and reprinted using the same printer. These fakes are also acquired with the same mobile phones in the same conditions. In order to account for the random variability of the acquisition device, the capture by mobile phone is performed 6 times (hereafter referred to as runs). The CDP acquired in this way are then processed using histogram matching to a reference CDP with a uniform histogram in order to mitigate the variability of the acquisition process and a quality check is performed to discard poorly captured CDP with an obvious blurring effect.

All the experiments described below are performed independently on each capture run and the selection of samples used for training, validation and testing is done randomly based on a seed. Results are then averaged and combined. The templates

[5]The dataset is publicly available and can be found here: https://sipcloud.unige.ch/index.php/s/tYKffnKNRgSwBAN

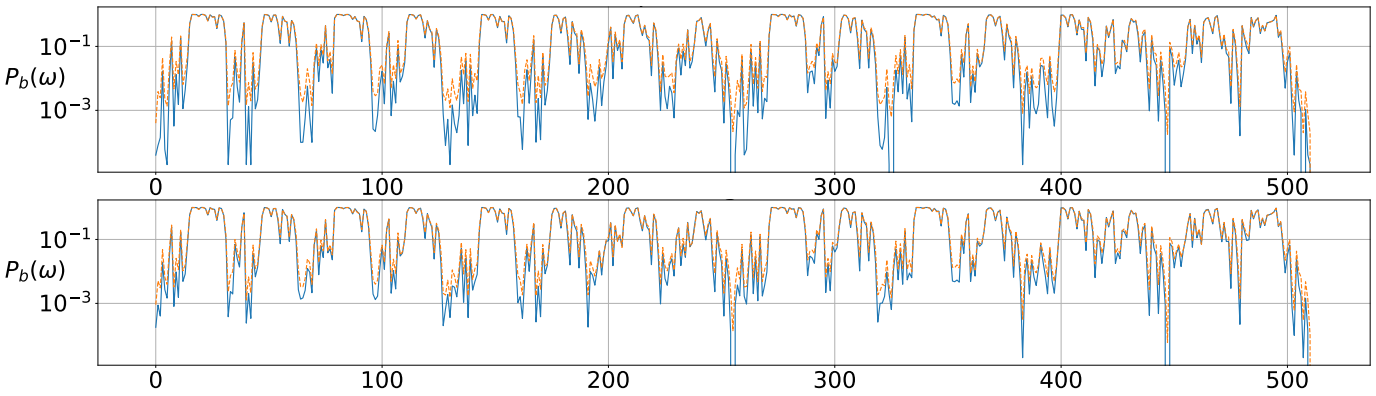


Fig. 8: Visualisation of trained codebooks for two different datasets: iPhone (top) and Samsung (bottom). The blue codebook is trained on originals, while the dashed orange codebook is trained on fakes. These results were obtained for the acquisition run 1 and random seed 0.

$\mathbf{t}$ are generated with a 50% density of black symbols. The templates $\mathbf{t}$, enrolled originals $\mathbf{x}$ and fakes $\mathbf{f}$ all have a size of 228×228 pixels. When testing the whole authentication scheme, we fix the training set size to 50 samples, validation set to 100 samples and test set to 500 samples.

B. Algorithm parameters

In order to learn the codebook, we need to fix a certain number of parameters and choose an estimator $\mathbf{x} \to \tilde{\mathbf{t}}$. As discussed in Section II, there are many different approaches to this problem. We decide to stick with Otsu's algorithm for binarization as deep binarization techniques tend to give unpredictable outcomes which are not usable for authentication. Moreover, we prefer to keep a simple explainable algorithm that can also be executed on any device. We fix the size of patterns ω in $\mathbf{t}$ to be of size 3×3 for the following reasons:

- This brings the total number of possible patterns down to $|\Omega| = 2^9 = 512$ which is small enough in comparison to the total number of patterns in a single template: $226^2 = 51'076$. We can thus expect to see every pattern appear roughly 100 times in each template.
- The printing process can produce some random deviations as discussed in Section III, but these deviations are local in the sense that they only affect neighbouring symbols in most cases. Thus, 3×3 patterns are sufficient to capture them (see Fig. 3 and 4 for an illustration).

VI. Experimental Results

A. Codebook estimation and pattern reliability

In this experiment, we train the codebook with Algorithm 1 and visualize the results for each pattern $\omega \in \Omega$. Each pattern is then associated to a number $0, 1, ..., 511$, simply based on its flattened binary representation for visualisation. In Fig. 8, the algorithm is run on four different datasets: iPhone originals, iPhone fakes, Samsung originals and Samsung fakes with 500 training samples. Although fake samples might not be available in real case scenarios, we decided to train the codebook algorithm both on originals and fakes in order to visualize the difference of behaviour between them. The x-axis represents the 512 different patterns ω and the y-axis represents the probability of bit-flip $P_b(\omega)$ for each pattern in log-scale.

A first observation clearly indicates that some patterns are much more reliable than others and thus confirms that the locality hypothesis (3) of the BPC model makes sense. For instance, the all-white pattern 511 has a probability of bit-flip of zero. The other two patterns with a zero probability of bit-flip are pattern 255 (one black symbol in top-left corner) and pattern 447 (one black symbol in top-right corner).

Another very important observation is a difference in the performances between the original CDP and fake CDP. Indeed, as expected, fake samples have a higher probability of bit-flip than originals (the orange curve is always above the blue curve). The strength of the authentication scheme of the BPC model relies on this difference and our ultimate goal is to select only the patterns where this difference is big. Notice also that this difference is bigger for the iPhone dataset than for the Samsung dataset which explains why the authentication scheme is less powerful on the latter.

Finally, another very interesting observation is that the behaviour of the codebook is very similar for both smartphones. This allows us to test cross-device performances where the codebook is trained on one smartphone and the authentication is performed using the other smartphone.

B. Codebook sensitivity to training set

The goal of this second experiment is to test the stability of the codebook estimation with respect to the number of training samples N_{train}. As discussed in Section V-B, every pattern appears 100 times on average in each template, it thus makes sense to run the algorithm on very small training sets. In order to measure the performance of a codebook $\mathcal{C}$ learned on a training set $\{(\mathbf{t}^n, \mathbf{x}^n)\}_{n=1}^{N_{train}}$, we compare it with a reference codebook $\mathcal{C}^{ref}$ learned on a big dataset of 500 pairs $\{(\mathbf{t}^n, \mathbf{x}^n)\}_{n=1}^{500}$. The comparison is simply done by computing an average ℓ_1-distance between the codebooks:

$$d_1(\mathcal{C}, \mathcal{C}^{ref}) = \frac{1}{|\Omega|} \sum_{\omega \in \Omega} |P_b(\omega) - P_b^{ref}(\omega)|. \qquad (11)$$

This article has been accepted for publication in IEEE Transactions on Information Forensics and Security. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIFS.2024.3359510

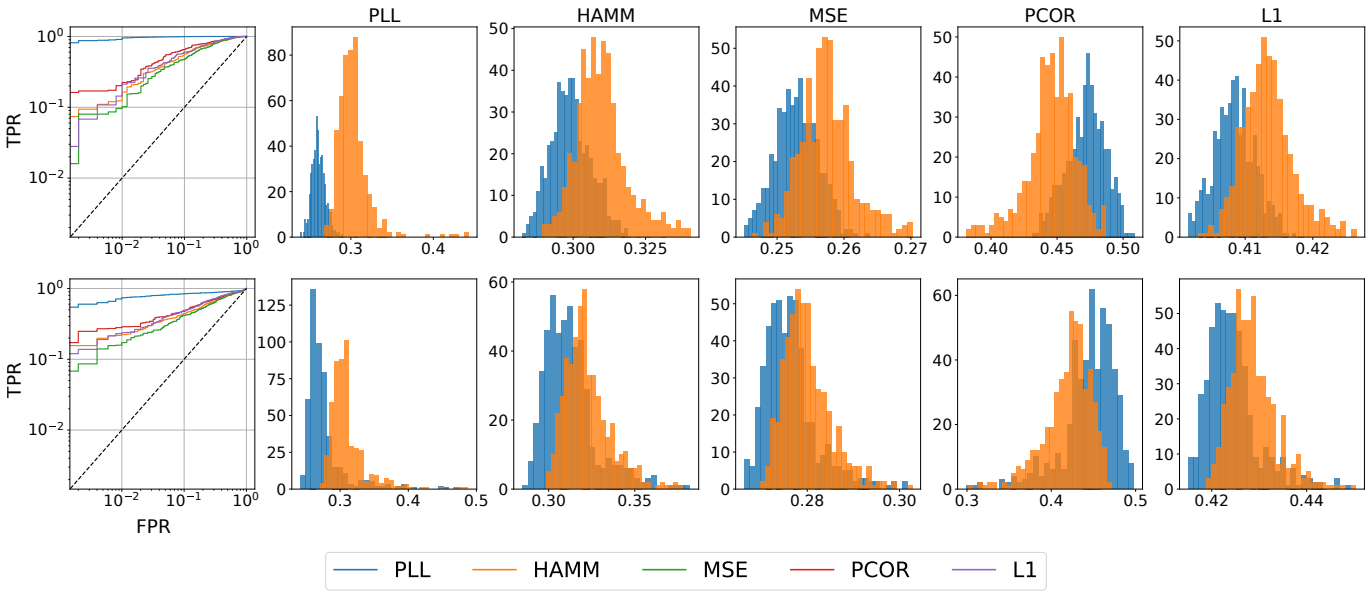


Fig. 9: Histograms of measures $d(\mathbf{t}, \mathbf{y})$ or $d(\mathbf{t}, \tilde{\mathbf{t}})$ for various metrics on originals (in blue) and fakes (in orange). The top row is measured on the iPhone dataset and the bottom row on the Samsung dataset. The left-most column shows the ROC curve of these histograms in terms of False Positive Rate (FPR) and True Positive Rate (TPR). These results were obtained for the acquisition run 1 and random seed 0.

Fig. 10 shows the results of this study for different training sets size with a number of samples going from 1 to 100 randomly selected. The curves show the average distance and the standard deviation for 30 different runs. What we can see is that when using 50 samples, the probabilities in the codebook $\mathcal{C}$ differ with the reference by less than 1% on average and the variability is very small. This explains why we decided to use 50 training samples in our experiments. We also notice that the Samsung dataset produces higher variability and less accurate estimations, again pointing out the fact that this dataset is less stable. This behaviour can be explained by the higher variability of images acquired by Samsung in terms of blur and focusing comparatively to the images acquired by iPhone. Overall, images captured with Samsung have a lower quality than those captured with iPhone.

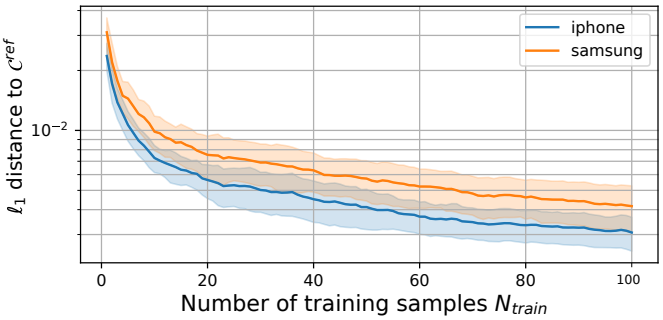


Fig. 10: Stability of the codebook estimation as a function of the number of samples used at training time. The solid curves shows the mean distance to the reference codebook and the shaded area the standard deviation for 30 random initialisations of the dataset.

C. Performance of PLL measure

In this experiment, we test the performances of the authentication scheme based on the posterior log-likelihood measure detailed in Section IV-A. The PLL measure is based on a codebook trained with 50 samples selected randomly and the testing set consists of 500 triples $\{(\mathbf{t}^n, \mathbf{x}^n, \mathbf{f}^n)\}_{n=1}^{500}$ for both datasets. The PLL measure is compared to various standard metrics such as:

- the Hamming distance (HAMM) between $\tilde{\mathbf{t}}$ and $\mathbf{t}$;
- the mean-squared-error (MSE) between $\mathbf{y}$ and $\mathbf{t}$;
- the Pearson correlation (PCOR) between $\mathbf{y}$ and $\mathbf{t}$;
- the Manhattan distance (L1) between $\mathbf{y}$ and $\mathbf{t}$.

Fig. 9 shows the histograms of the different metrics on originals and fakes. The first observation that we can make is that the standard metrics are very bad at separating originals and fakes. This is further confirmed by examining the receiver operating characteristic (ROC) curves of these metrics. On the other hand, the PLL metric is already performing quite good in this setup, with an almost non-overlapping histogram in the case of the iPhone dataset.

D. Authentication based on reliable patterns

We test here the second scheme of authentication based on the BPC model and measure the metrics presented above only restricted to reliable patterns as presented in Section IV-B.

To this end, we test several different thresholds $P_b(\omega) \leq \mu$ and compute a masked version of the metrics using (10). Fig. 11 shows the different ROC curves obtained for the various metrics.

This article has been accepted for publication in IEEE Transactions on Information Forensics and Security. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIFS.2024.3359510

TABLE II: Same-device and cross-device test of separability between originals and fakes in terms of minimal average probability of error (12) for each dataset and various metrics (in percent). The best results for each dataset are highlighted. Note that the trained codebook is only used for PLL and masked metrics. These results were obtained by averaging over 30 different random initialisations.

Datasets		Non-masked metrics $d(t,y)$					Masked metrics $d^m(t,y)$ with $\mu = 0.01$				
test	codebook	PLL	HAMM	MSE	PCOR	L1	PLL	HAMM	MSE	PCOR	L1
iphone	iphone	1.88	21.15	22.91	17.12	20.07	**0.80**	0.87	12.52	1.99	6.64
iphone	samsung	2.52	-	-	-	-	1.09	**1.07**	15.49	1.93	8.44
samsung	iphone	14.63	32.69	33.65	29.87	30.20	**10.91**	12.24	22.43	13.29	14.11
samsung	samsung	11.93	-	-	-	-	**7.89**	8.78	23.09	10.36	14.67

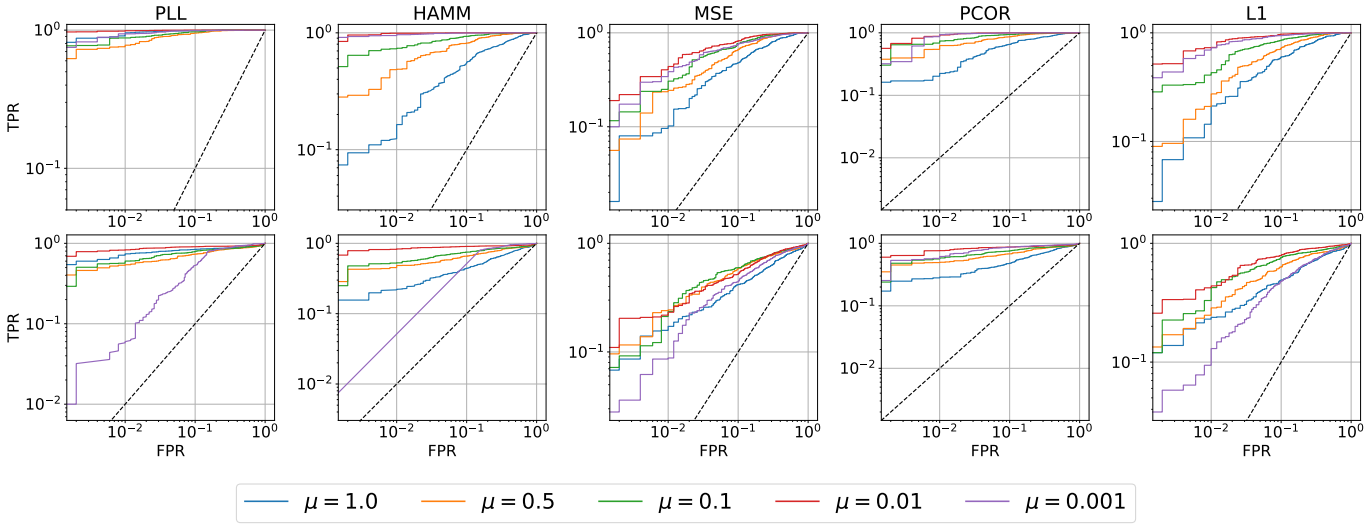


Fig. 11: ROC curves for masked measures on reliable patterns $d^m(\mathbf{t},\mathbf{y})$. The top row is measured on the iPhone dataset and the bottom row on the Samsung dataset. The different colors represent different choices of threshold μ, meaning that the similarity measure (10) is only performed on reliable patterns with probability of bit-flipping $P_b(\omega) \leq \mu$.

The top row shows results on the iPhone dataset and the bottom row, results on the Samsung dataset. As one can see, the restriction of the metric to patterns with low probability of bit-flipping drastically enhances the performance of the metrics. For instance, the standard Hamming distance is generally performing poorly on the iPhone dataset, but with a correct choice of threshold μ, it is able to separate originals and fakes almost perfectly. Even the PLL metric, which was already outperforming other metrics is enhanced by the mask. Another point to notice is that if we set the threshold value too low, the performances of the authentication scheme are getting worse, as there is only a few patterns left for authentication. Note that the choice $\mu = 1.0$ corresponds to the standard metrics, where all patterns contribute to the final aggregated score.

E. Authentication accuracy of BPC model

To compare all different approaches in a unified way and to establish the achievable authentication accuracy, we measure the performance of the different authentication schemes by using the minimal average probability of error, which measures the separability between originals and fakes:

$$P_{error}^{min} = \min_{\gamma} \frac{(1 - TPR^{\gamma}) \cdot N_{orig} + FPR^{\gamma} \cdot N_{fake}}{N_{orig} + N_{fake}}. \quad (12)$$

This measure computes an average between the probability of miss $(1 - TPR^{\gamma})$ and the probability of false acceptance FPR^{γ}, weighted by the number of original and fake samples for a decision threshold set at γ. We then choose the value of γ that minimizes this probability of error. We find this value to be much more indicative of the authentication capability of a given scheme than the Area Under Curve (AUC).

The results are presented in Table II. The codebook is trained on either the iPhone or the Samsung dataset and this codebook is then used for the authentication with both smartphones to test cross-device performances. Results are presented both for standard metrics and for masked metrics with a fixed threshold $\mu = 0.01$. The results are averaged over all 6 runs with randomization of training/testing set.

The first observation we can make is that the proposed authentication schemes significantly outperform the traditional approaches. Indeed, the error in classification drops from an average 20% of error for classical metrics, which is useless in practical applications, to a solid 1% error for the iPhone dataset for the best metrics.

The performances on Samsung dataset are not as impressive as the quality of the images are poorer than those acquired with iPhone but, nevertheless, we still see a clear improvement by using our approach. Cross-device performances, i.e., training the codebook on one device and performing authentication on another, are comparable to same-device performances but slightly less accurate. As such, this demonstrates how important it is to carefully choose the acquisition device.

F. Classifier baseline

In order to show the soundness of the codebook approach jointly with machine learning, we perform the following experiment involving a Support Vector Machine (SVM) classifier [21] trained directly on codebooks. The goal of this Machine-Learning-based experiment is to demonstrate that the information contained in the codebooks is sufficient to perform reliable classification. Our approach can be summarized in the following way for both one-class and two-class SVM models: for each pair $(\mathbf{t}^n, \mathbf{x}^n)$ and, if available, $(\mathbf{t}^n, \mathbf{f}^n)$ in the dataset where $n = 1, ..., N$:

1) Run the codebook estimation Algorithm 1 based on the pair $(\mathbf{t}^n, \mathbf{x}^n)$ or $(\mathbf{t}^n, \mathbf{f}^n)$ only. This outputs two lists of codebooks $\{\mathcal{C}^n_{\mathbf{x}}\}_{n=1}^N$ and $\{\mathcal{C}^n_{\mathbf{f}}\}_{n=1}^N$ which act as statistics of the particular couples $(\mathbf{t}^n, \mathbf{x}^n)$ and $(\mathbf{t}^n, \mathbf{f}^n)$;
2) Divide these lists into a training set $\{(\mathcal{C}^n_{\mathbf{x}}, \mathcal{C}^n_{\mathbf{f}})\}_{n=1}^{N_{train}}$ and a testing set $\{(\mathcal{C}^n_{\mathbf{x}}, \mathcal{C}^n_{\mathbf{f}})\}_{n=1}^{N_{test}}$;
3) • *For one-class SVM:* Train the model based on $\{\mathcal{C}^n_{\mathbf{x}}\}_{n=1}^{N_{train}}$;
   • *For two-class SVM:* Train the model based on $\{(\mathcal{C}^n_{\mathbf{x}}, \mathcal{C}^n_{\mathbf{f}})\}_{n=1}^{N_{train}}$;
4) Use the test set to compute the probability of error P_{error} score which is reported in Table III.

This experiment was done in two setups. A first setup, referred to as "full SVM", where the model is trained on the full codebook with all 512 binary patterns taken as input, that is both reliable and unreliable channels. The second setup, referred to as "masked SVM", where the model is trained only on reliable patterns, with $\mu = 0.01$. For this experiment, we select non-overlapping traih launch game changed how many saw open world experiences. Vast in size, deeply systemic, with a completely freeform approach to exploration, combat and puzzles. Few games match its scope for experimental play opportunities, and few give the player so much sense of authority over their own disconing set and testing set with $N_{train} = N_{test} = 500$ chosen randomly across the full dataset, and we use the radial-basis function (RBF) kernel. The hyperparameters of the SVM are optimized on the testing set using grid-search.

The results are presented in Table III. One-class SVM trained on the full codebook performs similarly to standard non-masked metrics shown in Table II at the task of discriminating original and fake samples. On the other hand, training the model only on reliable patterns greatly improves the capacity of the SVM model to classify them correctly. The results of two-class SVM are also very interesting. Indeed, classification based on codebooks shows perfect separability demonstrating that the codebooks represent a sufficient statistic for the classification task. However, training the two-class SVM requires the knowledge of fakes whereas training one-class SVM is agnostic to this requirement. In general, focusing only on reliable patterns tends to decrease the accuracy of the two-class SVM. Our current hypothesis is that, by putting a hard threshold selection on patterns $P_b(\omega) \leq \mu$, we discard some patterns ω that still contain relevant features.

These observations should however be mitigated by the fact that the two-class SVM does not represent practical situations as having a full access to fake samples is an unrealistic assumption. In this regard, one-class SVM represents a more realistic scenario for which our approach shows an improvement.

TABLE III: one-class SVM and two-class SVM trained on codebook features. The table shows the average probability of error (in percent) in classification for a trained model. The SVM column is trained on the full codebook while the masked SVM column is trained only on reliable patterns. The results are averaged across five random selections of training and testing set.

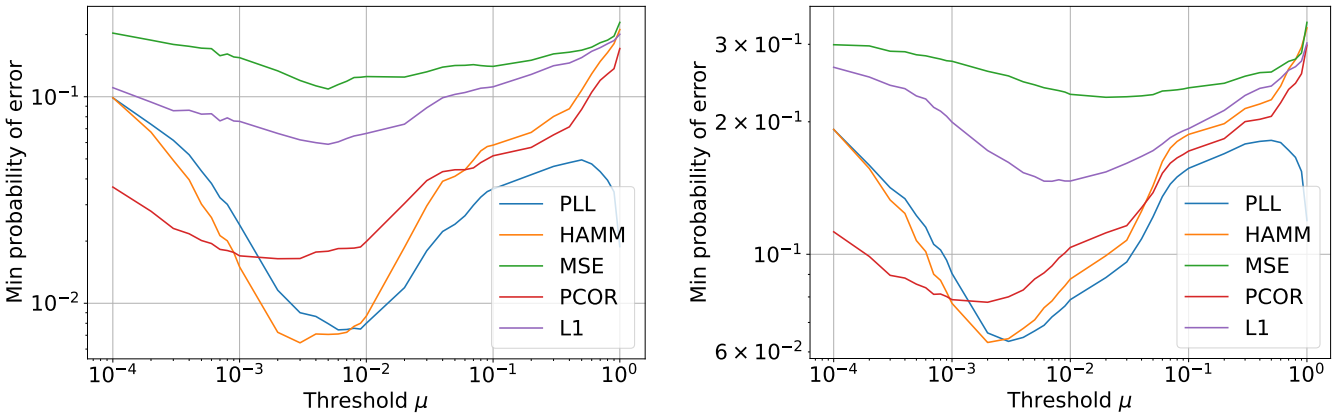
	full SVM		masked SVM	
Codebook	one-class	two-class	one-class	two-class
iphone	21.86	0.00	2.30	0.10
samsung	31.56	0.00	17.24	1.06

G. Pattern-wise contribution to authentication

The final experiment we conduct on this dataset is a more in-depth study of the choice of optimal threshold $P_b(\omega) \leq \mu$. Indeed, the test with different thresholds has demonstrated that the optimal choice of μ relies on a tradeoff between keeping too many unreliable patterns that are uninformative in terms of authentication score ($\mu \uparrow 1$) and having too few of them thus decreasing the overall capacity of distinguising originals and fakes ($\mu \downarrow 0$). This tradeoff is well illustrated in Fig. 12 where we represent the choice of threshold μ on the x-axis and the minimum probability of error as P^{min}_{error} on the y-axis (see the Section VI-E for a description of this measure of performance). This plot clearly indicates that the probability of error reaches a minimal value around $\mu = 0.01$ for all different metrics. Results based on iPhone dataset and Samsung dataset have similar behaviour.

Yet another way of measuring the contribution of each individual pattern $\omega \in \Omega$ to the authentication capacity is to measure a masked metric $d^{m_\omega}(\mathbf{t}, \mathbf{y})$ where the mask m_ω is non-zero only on the central pixel of a chosen pattern $\omega \in \Omega$.

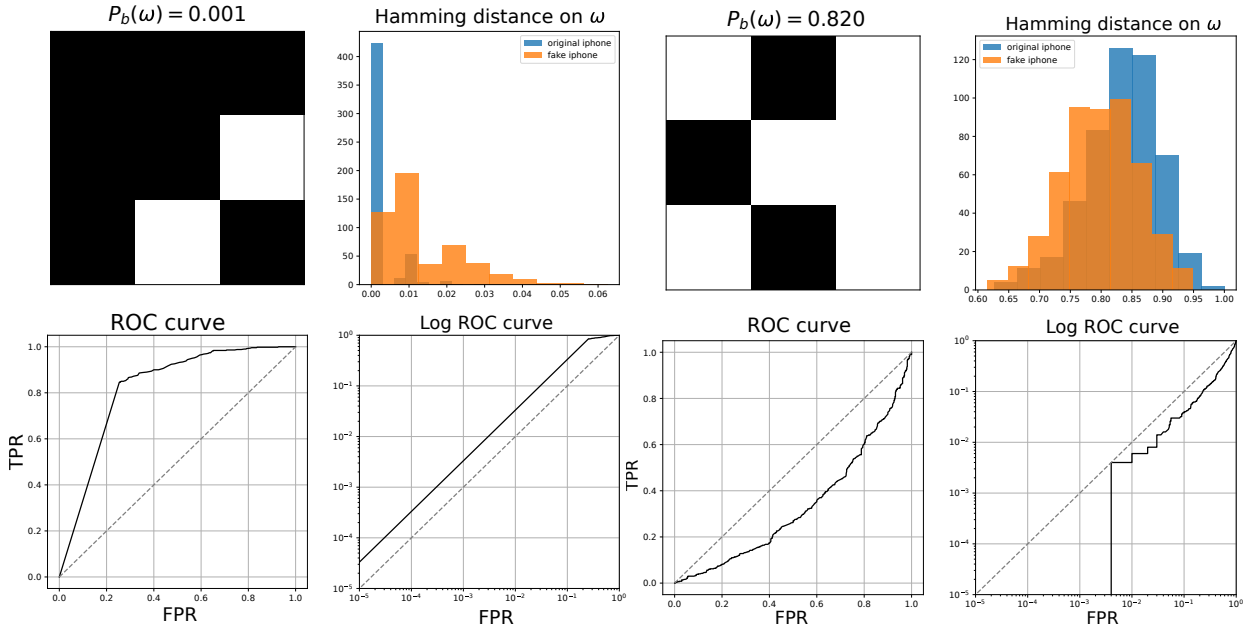
This article has been accepted for publication in IEEE Transactions on Information Forensics and Security. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIFS.2024.3359510



(a) iPhone dataset

(b) Samsung dataset

Fig. 12: Visualisation of the minimal probability of error (12) as a function of threshold μ on reliable patterns for iPhone dataset. Results are shown for various masked metrics.



(a) Reliable pattern

(b) Unreliable pattern

Fig. 13: Visualisation of the pattern-wise Hamming distance $d^{m_\omega}(t, \tilde{t})$. The pattern on the left has a small probability of bit-flip while the pattern on the right has a high probability of bit-flip. The ROC curve measures the separability between originals and fakes in both cases.

This is done in Fig. 13 for the Hamming distance. The graphics here show that reliable patterns in terms of probability of bit-flipping help discriminate between originals and fakes, while unreliable patterns mix them up.

VII. Conclusion

In this paper, we introduced the Binary Pattern-based Channel model, a new mathematical model for the description of the Printing-Imaging pipeline of copy detection patterns.

We proposed two novel authentication schemes for smartphones based on this model which give a viable alternative to the standard metrics, while still maintaining full interpretability of the results. We showed that our new authentication scheme can correctly detect ML-based attacks, in contrast to the standard metrics based on digital templates. Compared to modern deep learning approaches, our model requires very few training data and is very efficient to be run in practice, while still offering great performances against powerful ML attacks.

For future work, we aim at continuing to explore this model and further investigate the link between authentication accuracy and reliability in terms of bit-flipping. We are also interested in generalizing this approach to other printed unclonable features and different printing technologies.

REFERENCES

[1] 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*, 2003, pp. 145–154.

[2] G. Adams, S. Pollard, and S. Simske, "A study of the interaction of paper substrates on printed forensic imaging," in *Proceedings of the 11th ACM symposium on Document engineering*, 2011, pp. 263–266.

[3] S. Voloshynovskiy, M. Diephuis, F. Beekhof, O. Koval, and B. Keel, "Towards reproducible results in authentication based on physical non-cloneable functions: The forensic authentication microstructure optical set (famos)," in *2012 IEEE International Workshop on Information Forensics and Security (WIFS)*. IEEE, 2012, pp. 43–48.

[4] J. Picard, "Digital authentication with copy-detection patterns," in *Optical Security and Counterfeit Deterrence Techniques V*, vol. 5310. International Society for Optics and Photonics, 2004, pp. 176–183.

[5] J. Picard, P. Landry, and M. Bolay, "Counterfeit detection with qr codes," in *Proceedings of the 21st ACM Symposium on Document Engineering*, 2021, pp. 1–4.

[6] R. Chaban, O. Taran, J. Tutt, T. Holotyak, S. Bonev, and S. Voloshynovskiy, "Machine learning attack on copy detection patterns: are 1x1 patterns cloneable?" in *IEEE International Workshop on Information Forensics and Security (WIFS)*, December 2021.

[7] O. Taran, J. Tutt, T. Holotyak, R. Chaban, S. Bonev, and S. Voloshynovskiy, "Mobile authentication of copy detection patterns," *arXiv preprint arXiv:2203.02397*, 2022.

[8] B. Pulfer, Y. Belousov, J. Tutt, R. Chaban, O. Taran, T. Holotyak, and S. Voloshynovskiy, "Anomaly localization for copy detection patterns through print estimation," in *IEEE International Workshop on Information Forensics and Security (WIFS)*, Shanghai, China, December 2022.

[9] Y. Belousov, B. Pulfer, R. Chaban, J. Tutt, O. Taran, T. Holotyak, and S. Voloshynovskiy, "Digital twins of physical printing-imaging channel," in *IEEE International Workshop on Information Forensics and Security (WIFS)*, Shanghai, China, December 2022.

[10] J. Tutt, O. Taran, R. Chaban, B. Pulfer, Y. Belousov, T. Holotyak, and S. Voloshynovskiy, "Mathematical model of printing-imaging channel for blind detection of fake copy detection patterns," in *IEEE International Workshop on Information Forensics and Security (WIFS)*, Shanghai, China, 12 2022.

[11] T. Cover and J. Thomas, *Elements of Information Theory*. Wiley, 2012.

[12] S. Voloshynovskiy, T. Holotyak, and P. Bas, "Physical object authentication: detection-theoretic comparison of natural and artificial randomness," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2016, pp. 2029–2033.

[13] T. Holotyak, S. Voloshynovskiy, F. Beekhof, and O. Koval, "Fast identification of highly distorted images," in *Proceedings of SPIE / Media Forensics and Security XII*, San Jose, USA, January 21–24 2010.

[14] T. Holotyak, S. Voloshynovskiy, F. Farhadzadeh, O. Koval, and F. Beekhof, "Fast physical object identification based on unclonable features and soft fingerprinting," in *International Conference on Acoustics, Speech and Signal Processing ICASSP2011*, Prague, Czech Republic, May, 22-27 2011.

[15] E. Khermaza, I. Tkachenko, and J. Picard, "Can copy detection patterns be copied? evaluating the performance of attacks and highlighting the role of the detector," in *2021 IEEE International Workshop on Information Forensics and Security (WIFS)*. IEEE, 2021, pp. 1–6.

[16] R. Yadav, I. Tkachenko, A. Trémeau, and T. Fournel, "Estimation of copy-sensitive codes using a neural approach," in *Proceedings of the ACM Workshop on Information Hiding and Multimedia Security*, 2019, pp. 77–82.

[17] O. Taran, S. Bonev, and S. Voloshynovskiy, "Clonability of anti-counterfeiting printable graphical codes: a machine learning approach," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 2482–2486.

[18] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 9, no. 1, pp. 62–66, 1979.

[19] Z. Cui, W. Li, C. Yu, and N. Yu, "A new type of two-dimensional anti-counterfeit code for document authentication using neural networks," in *Proceedings of the 2020 4th International Conference on Cryptography, Security and Privacy*, 2020, pp. 68–73.

[20] R. Villán, S. Voloshynovskiy, O. Koval, and T. Pun, "Multilevel 2-d bar codes: Toward high-capacity storage modules for multimedia security and management," *IEEE Transactions on Information Forensics and Security*, vol. 1, no. 4, pp. 405–420, 2006.

[21] C.-C. Chang and C.-J. Lin, "Libsvm: A library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, may 2011. [Online]. Available: https://doi.org/10.1145/1961189.1961199