Mathematical model of printing-imaging channel for blind detection of fake copy detection patterns

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*—Nowadays, copy detection patterns (CDP) appear as a very promising anti-counterfeiting technology for physical object protection. However, the advent of deep learning as a powerful attacking tool has shown that the general authentication schemes are unable to compete and fail against such attacks. In this paper, we propose a new mathematical model of printing-imaging channel for the authentication of CDP together with a new detection scheme based on it. The results show that even deep learning created copy fakes unknown at the training stage can be reliably authenticated based on the proposed approach and using only digital references of CDP during authentication.**

***Index Terms*—copy detection patterns, authentication, predictor channel, one-class classification, deep learning fakes.**

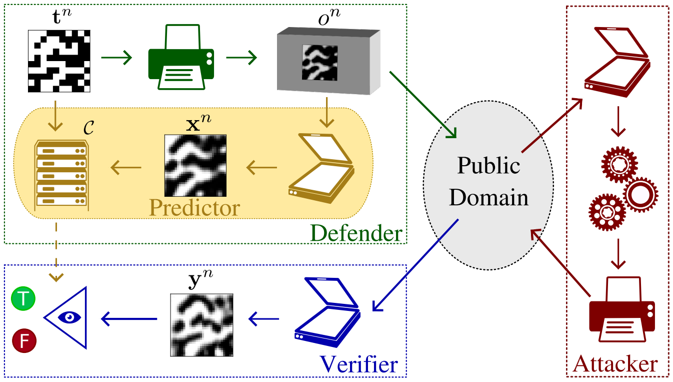

Fig. 1: Illustration of the PI channel seen as a 3-player game. The Defender (in green) generates and prints template $\mathbf{t}^n$ on an object package o^n and sends it to the public domain. The Attacker (in red) can use o^n to create a counterfeited version c^n of it. Finally, the Verifier (in blue) scans the object package and authentifies the probe $\mathbf{y}^n$ to decide whether it is an original package or a counterfeit. The novelty of our model is adding a predictor channel (in yellow) based on a codebook $\mathcal{C}$ which is trained by the Defender and used by the Verifier to enhance the classification results at the authentication stage.

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 inking, 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 uncloneable features of physical objects, such as randomness of ink blots or paper micro-structures [1]–[3]. Another technology is the Copy Detection Patterns (CDP) [4] which are random binary patterns of high entropy that are difficult to clone, such as very small sized QR codes. The advantages of CDP, in comparison to other technologies, are that they are cheap, easily integrable with a product into a structure of QR-code and digitally readable [5]. They are also easy to integrate in a track-and-trace distribution framework. The main challenge of this technology today is that, although being mainly robust to common copy attacks when simple decision rules are used based on the similarity to the reference template blueprint, it faces significant difficulties with the advanced machine-learning (ML) copy attacks. The possibility to use powerful deep classifiers in two-class classification allows one to reliably distinguish original CDPs 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 original and fakes [6]. In practice, the situation is further complicated by several factors:

- the high deviations in printing and imaging leading to large intra-class variabilities;
- ML attacks that are able to produce blueprint estimations with an accuracy score as high as 94% [7];
- 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.

Therefore, there is a critical need in a one-class (OC) authentication scheme able to operate in the generalized setup of the above printing-imaging channel without prior knowledge of the fakes. In this paper we adress these problems by:

- providing a new stochastic model describing the defender Printing-Imaging (PI) channel;

S. Voloshynovskiy is a corresponding author.

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

- proposing a new method of authentication based on the PI model able to perform authentication in the OC-classifier setup, i.e., under complete ignorance about the actual fakes;
- validating the proposed approach on a real dataset of CDPs of originals and ML-based fakes based on codes designed with 1×1 symbol and produced on two industrial printers;
- comparing the proposed method with traditional authentication techniques.

The paper is organized as follows. Section II introduces the problem formulation and presents a stochastic model of PI channel for the defender that forms the basis of the OC-classification framework. Section III presents the algorithm of OC-classification for CDPs in two variations. Section IV presents the results of performance and comparison with standard metrics on the same dataset. Finally, Section V concludes the paper and discusses possible extensions and perspectives. All mathematical notations used in the paper can be found in Table I.

II. Problem formulation

A. The printing-authentication scheme

The production of an anti-counterfeit technology using CDP is best described as a 3-player game with a Defender, an Attacker and a Verifier as shown in Fig. 1.

The Defender protects his brand by using a family of digital CDP blueprints $\{\mathbf{t}^n\}_{n=1}^N$ stored in the form of a binary matrix $\mathbf{t}^n$, which is then printed on the object package 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 [7]–[10] for investigations of attacking techniques). At the authentication stage, the Verifier receives an unindentified package (either o^n or c^n) from which a digital image $\mathbf{y}^n$ is acquired, using any device such as a scanner or 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 $\mathbf{f}^n$, and on the reference template $\mathbf{t}^n$.

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 & reprinting attacks [4]. Due to the dot gain effect of printers, a portion of the information stored in the template blueprint $\mathbf{t}$ is lost in the probe $\mathbf{y}$ through the process of printing and scanning. 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 $\mathbf{y}$. A very common way to perform the estimation $\tilde{\mathbf{t}}$ is to use Otsu's binarization algorithm and then a majority voting for each symbol. Fig. 2 on the next page illustrates this technique.

TABLE I: Mathematical notations used in the paper.

	Mathematical notation	Meaning
CDPs	$\mathbf{t}$	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 $\mathbf{y}$
PI Model	$\mathbf{T}$	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 black symbol in $\mathbf{T}$
	$\omega \in \Omega$	set of all neighbourhoods
	$P(\omega)$	positive probability at ω
	$P_b(\omega)$	probability of bit-flipping at ω
	$\mathcal{C}$	codebook of probabilities
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 $\mathbf{x}$

Nowadays, new techniques emerge with the use of machine learning, allowing one to train deep classifiers [6], [11] and deep binarization techniques [7]–[10]. 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.

C. Stochastic model of Printing-Imaging channel

The PI channel can be described mathematically 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 black symbol;
- $\mathbf{X}$ is a random matrix of size $kL \times kL$, $X_{ij} \in [0,1]$ for some magnification factor[1] $k = 1, 2, 3, ...$;
- $\tilde{\mathbf{T}}$ is a random binary matrix of size $L \times L$.

In reality, when we pass a template $\mathbf{t}$ through the PI channel, some distorsions occur in $\mathbf{x}$ due to the dot-gain effect and printing-related natural randomness. Thus, when we try to estimate $\tilde{\mathbf{t}}$ from $\mathbf{x}$, we end up with some errors, dependant on the printer, type of paper, acquisition device and chosen estimator. In this paper, we are mostly interested in understanding the probability distribution $\mathbb{P}(\tilde{\mathbf{T}}|\mathbf{T})$, which we believe to be highly correlated with the particular choices of print-acquire-estimate system and is central when trying to estimate information loss.

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

In [12], the authors model this probability distribution as a Binary Symmetric Channel (BSC). This model assumes that each symbol T_{ij} in $\mathbf{T}$ has a certain probability P_b of bit-flipping, independently of its location (i,j). We conjecture that the BSC model is too simple to capture the random behaviour of printing, as it does not take into account the local dependency of neighbouring sites and rather learns an average probability of bit-error across the whole template. Another related model with multilevel symbols has been studied in [13]. Inspired by the BSC model, we introduce a new stochastic model with three key assumptions:

1) *Markovianity:* the posterior probability at a particular symbol location (i,j) only depends on the local neighbourhood ω_{ij} surrounding it:
$$\mathbb{P}(\tilde{T}_{ij}|\mathbf{T}) = \mathbb{P}(\tilde{T}_{ij}|\omega_{ij}), \tag{1}$$
where ω_{ij} is a small neighbourhood surrounding symbol T_{ij}, typically a square matrix centered around (i,j):
$$\omega_{ij} = \{T_{i\pm a, j\pm b} | 0 \leq a, b < h/2\},$$
where $h = 1, 3, 5, ...$ is fixed by the model and defines the size of the neighbourhood.
2) *Stationarity:* the posterior probability does not depend on the location inside the image. Similar patterns in $\mathbf{T}$ lead to similar probability values[2]:
$$\mathbb{P}(\tilde{T}_{ij}|\omega_{ij}) = \mathbb{P}(\tilde{T}_{rs}|\omega_{rs}), \text{ if } \omega_{ij} = \omega_{rs}. \tag{2}$$
3) *Posterior independance:* the joint posterior probability factorizes as:
$$\mathbb{P}(\tilde{\mathbf{T}}|\mathbf{T}) = \prod_{i,j} \mathbb{P}(\tilde{T}_{ij}|\mathbf{T}). \tag{3}$$

With assumptions (1) and (2), one can easily prove the expectation formula for the posterior distribution:
$$\mathbb{P}(\tilde{T}_{ij}|\omega_{ij}) = \mathbb{E}_{r,s:\omega_{rs}=\omega_{ij}}[\mathbb{P}(\tilde{T}_{rs}|\omega_{rs})]. \tag{4}$$
This formula is a key to the proposed authentication scheme as it can be estimated directly using Monte-Carlo method from a training dataset. For each type of neighbourhood $\omega \in \Omega$ (there can be at most 2^{h^2}), we learn the probability distribution which is highly correlated with the PI channel on which it was trained. Two measures associated with this distribution are the posterior probability of bit-flipping $P_b(\omega)$ and the positive posterior probability $P(\omega)$, which we define as:
$$P_b(\omega_{ij}) := \mathbb{P}(\tilde{T}_{ij} \neq T_{ij}|\omega_{ij}), \tag{5}$$
$$P(\omega_{ij}) := \mathbb{P}(\tilde{T}_{ij} = 1|\omega_{ij}). \tag{6}$$

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.

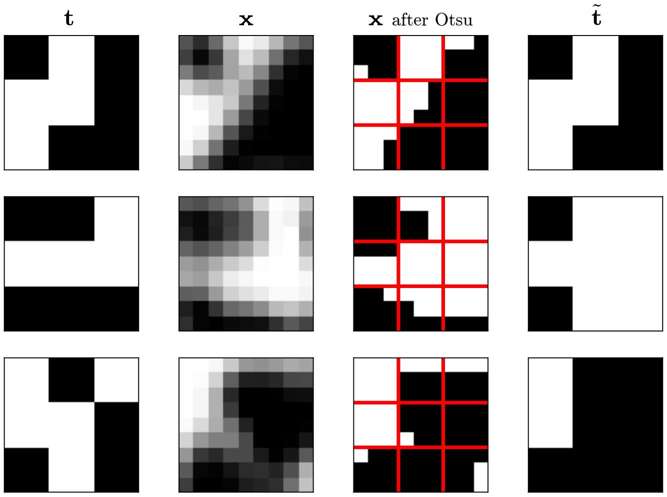

Fig. 2: Otsu's binarization technique followed by majority voting. The first column shows different neighbourhoods ω, the second column the printed originals $\mathbf{x}$, the third column $\mathbf{x}$ after Otsu's binarization and the fourth column, the estimated template $\tilde{\mathbf{t}}$ after majority voting. Red lines highlight the 3×3 patches in $\mathbf{x}$ corresponding to one symbol in $\mathbf{t}$. Different types of distorsions are illustrated leading to estimation errors in $\tilde{\mathbf{t}}$.

D. Metric in the PI channel

The introduced PI channel 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.

Lemma II.1. *In the PI channel model, the posterior log-likelihood can be computed as:*
$$\log \mathbb{P}(\tilde{\mathbf{T}} = \tilde{\mathbf{t}}|\mathbf{T}) = \sum_{i,j} \log(1 - |\tilde{t}_{ij} - P(\omega_{ij})|). \tag{7}$$

Proof. The proof relies on two steps. The first one is to use conditional independence of the symbols in $\tilde{\mathbf{T}}$ given $\mathbf{T}$ and Markovianity:
$$\begin{aligned}\log \mathbb{P}(\tilde{\mathbf{T}} = \tilde{\mathbf{t}}|\mathbf{T}) &= \sum_{i,j} \log \mathbb{P}(\tilde{T}_{ij} = \tilde{t}_{ij}|\mathbf{T}) \\ &= \sum_{i,j} \log \mathbb{P}(\tilde{T}_{ij} = \tilde{t}_{ij}|\omega_{ij}).\end{aligned}$$

The second step is then a simple case study for $\tilde{t}_{ij} \in \{0,1\}$:
$$\mathbb{P}(\tilde{T}_{ij} = \tilde{t}_{ij}|\omega_{ij}) = 1 - |\tilde{t}_{ij} - P(\omega_{ij})|.$$
■

III. ONE-CLASS CDP CLASSIFICATION ALGORITHMS

The core idea of building an authentication system based on the PI channel model is to introduce *the predictor channel*, which is trained using both digital templates $\mathbf{t}$ and acquired originals $\mathbf{x}$ and to learn a codebook $\mathcal{C}$ of probabilities for each neighbourhood $\omega \in \Omega$.

[2] The printing and scanning 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. (2) 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).

To train the predictor, we create two dictionaries $\mathbb{D}$ and $\mathbb{D}_b$ whose keys are the different types of neighbourhoods. For each $\omega_{ij} \in \Omega$, $\mathbb{D}[\omega_{ij}]$ lists the corresponding values of symbol $\tilde{t}_{ij}$ and $\mathbb{D}_b[\omega_{ij}]$ lists the boolean values $(\tilde{t}_{ij} \neq t_{ij})$. Finally, we compute the codebook $\mathcal{C}$, which is a database storing the statistics $P(\omega)$ and $P_b(\omega)$ for each type of neighbourhood ω. A pseudo-code is given in Algorithm 1.

A. The likelihood score model

The first authentication scheme is a direct implementation of (7). It starts by learning the codebook $\mathcal{C}$, running Algorithm 1 on the training set. For the authentication of a probe $\mathbf{y}$, we perform the following steps:

1) estimate $\tilde{\mathbf{t}}$ from the probe $\mathbf{y}$;
2) with the reference template $\mathbf{t}$, search the probability $P(\omega_{ij})$ in $\mathcal{C}$, for each neighbourhood ω_{ij} in $\mathbf{t}$;
3) compute the likelihood score of $\tilde{\mathbf{t}}$ applying (7);
4) compare the score with a chosen threshold fixed on the validation set to decide whether $\mathbf{y}$ is original or fake.

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 in 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 attention model

The attention model is similar in essence to the preceding model but differs in several ways. The idea here is to use the probability bit-error map $P_b(\omega_{ij})$ as a mask, only keeping symbols that have a low probability of bit-error on the training set. In this way, we remove all regions in $\mathbf{y}$ that are known to produce high error for original samples $\mathbf{x}$. Training is done similarly to the likelihood score model above. For the authentication, we do:

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 $m_{ij} := (P_b(\omega_{ij}) < \mu)$ 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) weight the chosen metric $d(\mathbf{t}, \mathbf{y})$ by using the binary mask, upsampling it if needed:

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

IV. Results

A. Dataset choice

For our experiments, we use the Indigo 1×1 base dataset, presented in [7]. It is constituted of 720 different templates $\mathbf{t}$ printed with two different printers: HP Indigo 5500 DS (HPI55) and HP Indigo 7600 DS (HPI76) at 812.8 dpi, which we refer to as $\mathbf{x}^{55}$ and $\mathbf{x}^{76}$. It also includes ML-based fakes of four different types: $\mathbf{f}^{55/55}$, $\mathbf{f}^{76/55}$, $\mathbf{f}^{55/76}$ and $\mathbf{f}^{76/76}$ where fake $\mathbf{f}^{mm/nn}$ is obtained from $\mathbf{x}^{nn}$ by the process of deepnet-based binarization, printed using HPImm and rescanned.

Algorithm 1 Algorithm for predictor training

```
Input: training set $\{(\mathbf{t}^n, \mathbf{x}^n)\}_{n=1}^N$
Output: learned codebook $\mathcal{C} = (\omega, P(\omega), P_b(\omega))_{\omega \in \Omega}$
    Initialisation:
    create two dictionaries $\mathbb{D}$ and $\mathbb{D}_b$ with the set $\Omega$ as keys
    and empty lists as values.
    for $n = 1$ to $N$ do
        estimate $\tilde{\mathbf{t}}^n$ from $\mathbf{x}^n$
        for symbol $t_{ij}^n$ in $\mathbf{t}^n$ do
            extract neighourhood $\omega_{ij}^n$ in $\mathbf{t}^n$
            extract symbol $\tilde{t}_{ij}^n$ in $\tilde{\mathbf{t}}^n$
            append value $\tilde{t}_{ij}^n$ in dictionary $\mathbb{D}$ at key $\omega_{ij}^n$
            append boolean value $(\tilde{t}_{ij}^n \neq t_{ij}^n)$ in dictionary $\mathbb{D}_b$ at
            key $\omega_{ij}^n$
        end for
    end for
    for $\omega$ in $\Omega$ do
        compute mean value: $P(\omega) = \text{mean}(\mathbb{D}[\omega])$
        compute mean value: $P_b(\omega) = \text{mean}(\mathbb{D}_b[\omega])$
        store the triple $(\omega, P(\omega), P_b(\omega))$
    end for
    return codebook $\mathcal{C} = (\omega, P(\omega), P_b(\omega))_{\omega \in \Omega}$
```

In this work, we only concentrate on the templates with 50% density of black symbols. The templates $\mathbf{t}$ have a size of 228×228 symbols while $\mathbf{x}$ and $\mathbf{f}$ have a size of 684×684, that is a magnification by a factor $k = 3$. We fix the training set size to 50 samples, validation set to 100 samples and test set to 500 samples.

B. Predictor algorithm parameters

In order to train the predictor, we fix a certain number of parameters. The first one is the estimator $\mathbf{x} \to \tilde{\mathbf{t}}$. As we saw in Section II, there are many different approaches to it. We decide to use Otsu's algorithm for binarization followed by majority voting on each 3×3 patch corresponding to one symbol in $\mathbf{t}$. We fix the size of neighbourhoods ω in $\mathbf{t}$ to be of size 3×3 for the following reasons:

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

TABLE II: Results in percent of the Area Under Curve (AUC) for each type of originals and fakes and various metrics. Best results for each type of fakes are highlighted. Average results are shown per printer and in total.

	HPI55 originals x^{55}					HPI76 originals x^{76}					
	$f^{55/55}$	$f^{55/76}$	$f^{76/55}$	$f^{76/76}$	Average	$f^{55/55}$	$f^{55/76}$	$f^{76/55}$	$f^{76/76}$	Average	Total
metrics											
LLS	99.88	99.89	100	100	99.94	87.24	85.69	**99.97**	**99.98**	93.22	96.58
MSE	59.60	59.05	35.85	27.77	45.57	97.47	**99.03**	85.85	82.12	91.12	68.34
PCOR	87.11	88.41	94.75	92.36	90.66	88.97	90.49	95.79	93.88	92.28	91.47
HAMM	63.39	63.82	69.46	61.12	64.45	85.36	86.45	89.35	83.14	86.08	75.26
masked											
M-LLS	**99.98**	99.96	**100**	**100**	99.99	**99.29**	98.97	99.94	99.84	**99.51**	**99.75**
M-MSE	99.97	99.95	100	100	99.98	97.04	94.76	99.94	99.85	97.9	98.94
M-PCOR	96.30	92.46	99.35	98.58	96.67	97.11	95.45	98.42	97.72	97.17	96.92
M-HAMM	99.98	**99.97**	100	100	**99.99**	99.22	98.89	99.94	99.85	99.48	99.73

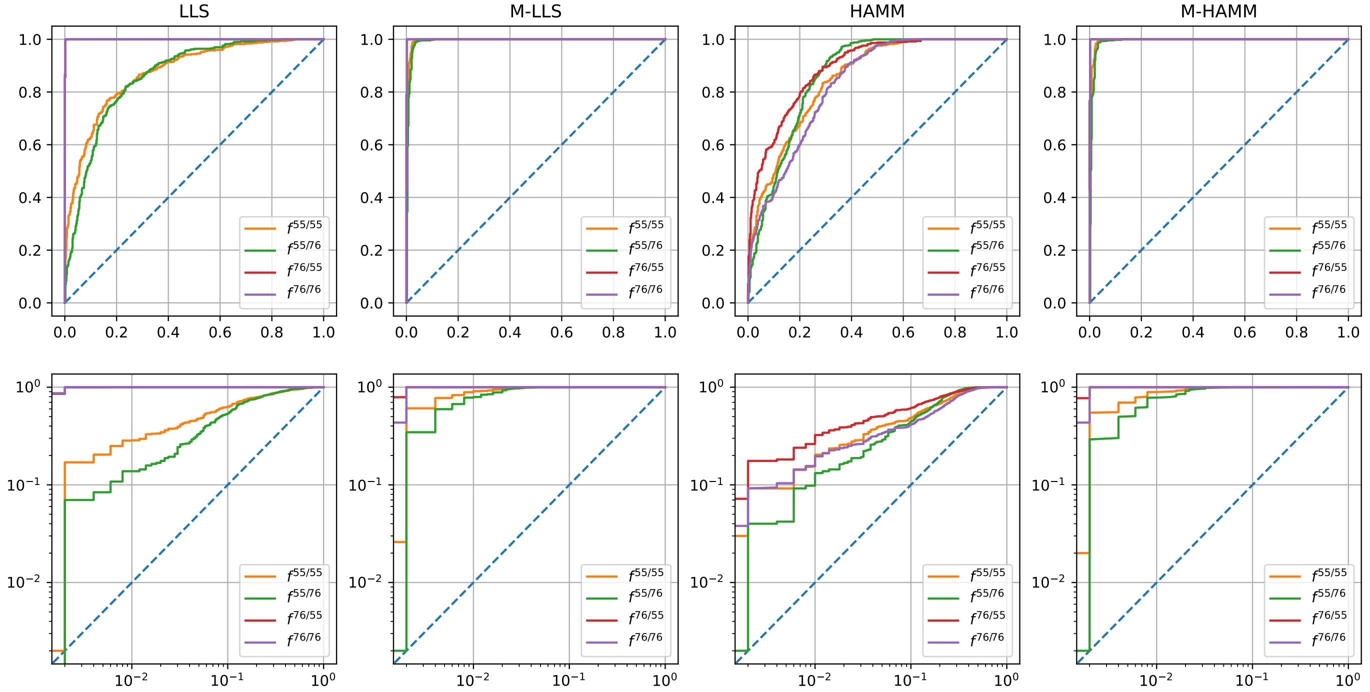

Fig. 3: Visualisation of ROC curves for original x^{76} and all four types of fakes. First column is the LLS metric, second column the masked LLS, third column the Hamming distance and fourth column the masked Hamming distance. The second row shows the same plots with logarithmic scale on both axes.

C. Discussion

To compare all different approaches in a unified way, we test both originals $\mathbf{x}^{55}$ and $\mathbf{x}^{76}$ separately against all four kind of ML fakes $\mathbf{f}^{55/55}$, $\mathbf{f}^{55/76}$, $\mathbf{f}^{76/55}$ and $\mathbf{f}^{76/76}$. The metrics that we use are:

- the log-likelihood score (LLS) described in Section III-A;
- the mean-squared-error (MSE) between $\mathbf{y}$ and $\mathbf{t}$;
- the Pearson correlation (PCOR) between $\mathbf{y}$ and $\mathbf{t}$;
- the Hamming distance (HAMM) between $\mathbf{t}$ and $\tilde{\mathbf{t}}$;
- the same four metrics mentioned above with a mask as described in Section III-B.

For each metric, we compute the associated ROC curves and report the AUC score. The AUC score is averaged over ten runs with randomization of training/testing set. All results are summarized in Table II.

A first observation at the results in Table II shows that discriminating between originals and fakes is more accurate for $\mathbf{x}^{76}$ than for $\mathbf{x}^{55}$. In general, the results show that the metric LLS outperforms the other metrics. On average, M-LLS, its masked version, appears as the best metric with a very reliable AUC score on all types of fakes.

The masked metrics show a great improvement in AUC score over all their non-masked counterparts. This is further illustrated in Fig. 3, where we compare side-by-side masked and non-masked metrics for LLS and Hamming metrics.

Surprisingly, MSE proves to be the best metric for discriminating $\mathbf{x}^{76}$ and $\mathbf{f}^{55/76}$. This result should however be mitigated by the following observations:

- metric M-LLS performs very close to MSE and even outperformed it on certain runs;
- the high variability in performance of MSE on different types of fakes makes it highly unreliable for authentication, as shown by its average score.

D. Model stability

Another question that we investigated is the stability of Algorithm 1 with respect to the size of the training set. We already discussed, in Section IV-B, the fact that every neighbourhood appears 100 times on average in each template. Thus, it 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 $\{(\mathbf{t}^n, \mathbf{x}^n)\}$, we compare it with a reference codebook $\mathcal{C}^{ref}$ learned on the whole dataset of 720 pairs $\{(\mathbf{t}^n, \mathbf{x}^n)\}$. We then simply compute an average ℓ_1-distance between the predictions:

$$d_1(\mathcal{C}, \mathcal{C}^{ref}) = \frac{1}{|\Omega|} \sum_{\omega \in \Omega} |P(\omega) - P^{ref}(\omega)|. \tag{8}$$

Fig. 4 shows the results of this study for different training sets size with a number of samples going from 1 to 100. 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.

V. Conclusion

In this paper, we introduced a new mathematical model for the description of the Printing-Imaging channel based on local statistics.

We proposed two novel OC-authentication schemes based on this model which outperform the standard metrics used nowadays, while still maintaining full interpretability of the results. We showed that even ML-based attacks cannot fool our new authentication system. In constrast with modern deep learning approaches, our model requires very few training data and does not require much time 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 as the information-theoretic aspects can be deeper investigated. We also plan to replace the simple estimator with more sophisticated techniques based on neural networks and perform the comparison of the proposed approach with deep classifiers. Finally, we plan to extend the results on a new dataset acquired by several types of mobile phones which will bring more variability and new challenges for the PI channel model.

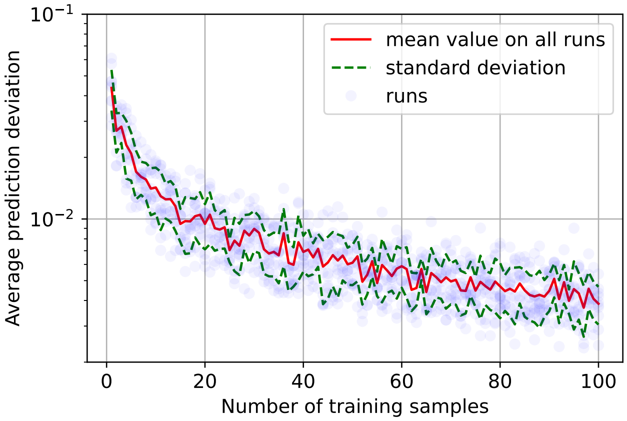

Fig. 4: Variability of codebook $\mathcal{C}$ with respect to the size of the training set.

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] 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.

[7] 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.

[8] 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.

[9] 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.

[10] 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.

[11] 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.

[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] 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.