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.2024.3361798

A Machine Learning-Based Digital Twin for Anti-Counterfeiting Applications with Copy Detection Patterns

Yury Belousov, Guillaume Quétant, Brian Pulfer, Roman Chaban, Joakim Tutt,
Olga Taran, Taras Holotyak and Slava Voloshynovskiy
Department of Computer Science, University of Geneva, Switzerland
{name.surname, svolos}@unige.ch

***Abstract*—In this paper, we present a new approach to model a printing-imaging channel using a machine learning-based "digital twin" for copy detection patterns (CDP). The CDP are considered as modern anti-counterfeiting features in multiple applications. Our digital twin is formulated within the information-theoretic framework of TURBO initially developed for high energy physics simulations, using variational approximations of mutual information for both encoder and decoder in the bidirectional exchange of information. This model extends various architectural designs, including paired pix2pix and unpaired CycleGAN, for image-to-image translation. Applicable to any type of printing and imaging devices, the model needs only training data comprising digital templates sent to a printing device and data acquired by an imaging device. The data can be paired, unpaired, or hybrid, ensuring architectural flexibility and scalability for multiple practical setups. We explore the influence of various architectural factors, metrics, and discriminators on the overall system's performance in generating and predicting printed CDP from their digital versions and vice versa. We also performed a comparison with several state-of-the-art methods for image-to-image translation applications. The simulation code and extended results are publicly available at https://gitlab.unige.ch/sip-group/digital-twin.**

***Index Terms*—Copy detection patterns, machine learning, digital twin, information theory, variational approximation.**

I. Introduction

In recent years copy detection patterns (CDP) [1], [2] attracted a lot of attention as a digital and machine-readable anti-counterfeiting technology. The CDP are broadly used for the anti-counterfeiting protection of product packaging, secure labels and documents. At the same time, a great deal of research has recently been carried out into the various factors influencing the accuracy of CDP authentication [3]–[16] and source attribution of printed documents [17]. To study the performance of CDP-based authentication, access to training and test sets of sufficient size and diversity to transfer academic research to an industrial level is required. However, the production of datasets of real CDP is a costly and time consuming process. It requires the printing and acquisition of original CDP and the production and acquisition of fakes, preferably on equipment close to the industrial one. Thus, the cost, time and needed deep domain knowledge considerably limit the study of the anti-counterfeiting aspects of CDP. A possible solution to reduce time and cost would be to develop a mathematical model of the printing and imaging channel. Nevertheless, the development of such model requires a lot of domain knowledge for each specific case. This, in turn, decreases the scalability of the system to new products, production technologies, and imaging devices, making the optimization process difficult, time-consuming, and expensive. Furthermore, even in the hypothetical scenario where such accurate models could be developed, the non-differentiable nature of these models could impose limitations on the optimization of CDP-based authentication systems.

It should be pointed out that the knowledge of the physical printing-imaging channel plays a very important role in anti-counterfeiting systems and is crucial for both the defender and the attacker. On the side of the defender, the knowledge of a model for this printing-imaging channel can: (a) enable the overall optimisation of the whole authentication system by end-to-end training of CDP generation, printing, imaging and authentication decoders and decision making, (b) simulate and predict the intra-class variabilities and (c) help generate synthetic samples of both originals and fakes that can be used to efficiently train authentication modules (classifiers).

The attacker can also benefit from such a model by: (a) optimising the estimation of digital templates from the physical samples in the scope of copy attacks and (b) developing adversarial samples for the physical domain.

At the same time, the design of *digital twins* of printing-imaging channels is not a trivial task. To simplify it somehow, one can consider printing and imaging systems separately.

Besides some works [18]–[20] addressing the physics of specific production systems, there is no generalized theory on how to model even straightforward printing systems characterized by a high level of stochasticity and nonlinearity. The printing process model of each printing technology, such as off-set, digital off-set, inkjet or flexo, representing the most significant interest for practical applications, is very complex and domain-specific. Moreover, such a model should consider not only hardware but also software particularities of drivers that significantly impact the printing outcome. Altogether, it requires a lot of domain-specific know-how and makes the model development for each printing system very time-consuming. Furthermore, the validation of the model is also expensive and might require tuning many parameters.

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.2024.3361798

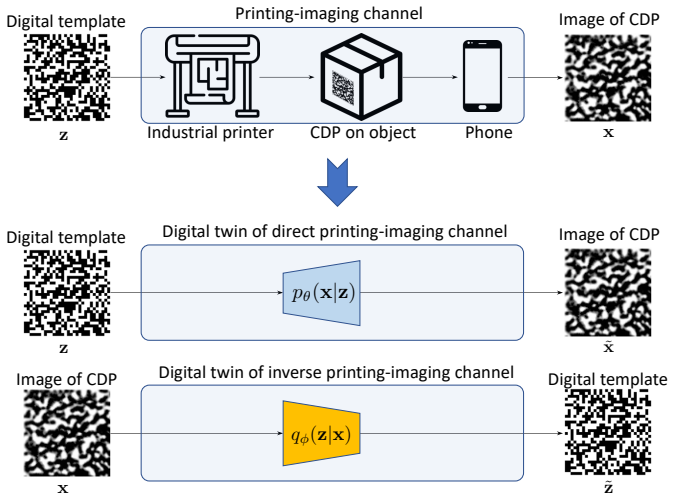


Fig. 1. Physical printing-imaging channel and its digital twin counterpart. In physical channel, a digital template $\mathbf{z}$ is reproduced on surface of a digital object in a form of CDP by an industrial printer. An imaging device represented by an end-user phone acquires an image of CDP $\mathbf{x}$. The acquired image is a degraded version of $\mathbf{z}$. The digital twin of the printing-imaging channel replaces a complex printing-imaging system by deep neural architectures thus allowing to estimate the CDP $\tilde{\mathbf{x}}$ from the template $\mathbf{z}$ in the direct consideration and vice versa in the inverse.

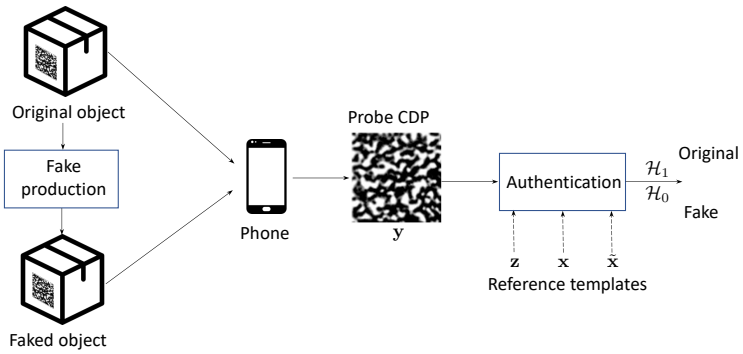


Fig. 2. Authentication protocol based on CDP. A counterfeiter aims at producing faked objects as close as possible to the original ones. It also concerns the reproduction of CDP. The verifier acquires a probe CDP $\mathbf{y}$ from an object presented for the authentication which can be an original object or faked one. The authentication compares the correspondence of statistics of probe $\mathbf{y}$ with those of digital template $\mathbf{z}$, physical template $\mathbf{x}$ or synthetically generated template $\tilde{\mathbf{x}}$ and makes the decision whether the object under the authentication is original (hypothesis $\mathcal{H}_1$) or fake (hypothesis $\mathcal{H}_0$).

Modeling the acquisition and imaging process is no less important. Besides some remarkable exceptions [21], [22] that present the models of noise in the CCD and CMOS imaging devices and practical methodologies of their validation, the simulation of the interaction between the incident light and reflecting object surface is not a trivial task [23]. The imaging device hardware components and drivers' settings such as type of sensor, resolution of sensors, optics, ISO, shutter time, denoising, white color balancing, compression, etc., greatly impact the output image features. Finally, similarly to the models of production systems, there is no guarantee that the imaging model will be interpretable and differentiable and thus suitable for the envisioned optimization tasks.

In this paper, we aim at addressing these challenges and shortcomings based on a machine-learning framework and introduce a concept of *digital twins* of complex printing-imaging systems. More specifically, we propose a *digital twin* system that simulates the entire chain from the digital template $\mathbf{z}$ to the acquired image of CDP $\mathbf{x}$. The proposed system is based on an auto-encoder (AE) structure.

The framework of *digital twins* might be used as a simulator of complex physical printing-imaging systems for:

- creation of differentiable models for the investigation of unexplored adversarial attacks in the physical world;
- generation of synthetic samples of original and fake CDP to train corresponding classifiers even when no fakes are known in advance by using synthetic samples as fakes;
- generation of synthetic physical templates for authentication as opposed to the authentication based on real physical templates;
- creation of augmentations for self-supervised learning (SSL) methods;
- investigation of variability and estimation of uncertainty in printing-imaging systems.

The current work is an extension of our previous experimental work [24] that demonstrated a proof of concept on a limited data set of CDP acquired by a scanner. The main contributions of the present work are:

- the complete information-theoretic problem formulation of printing-imaging digital twin based on the TURBO

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.2024.3361798

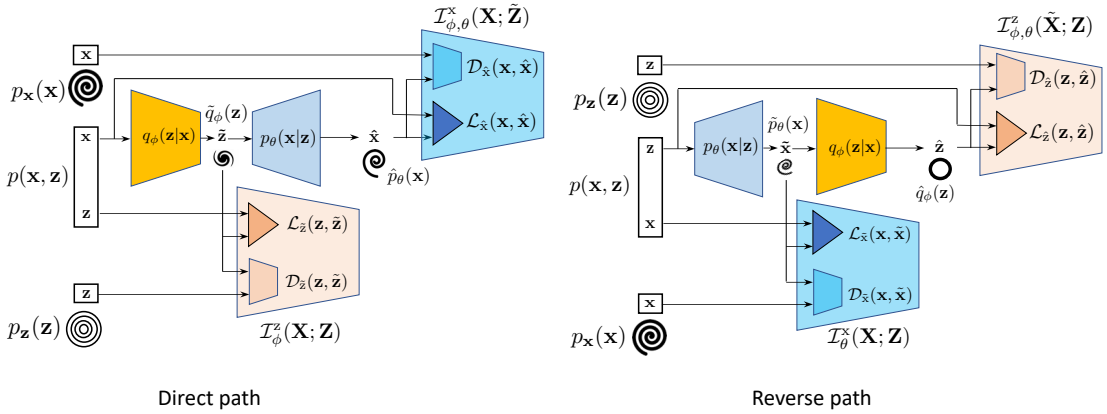


Fig. 3. Training of *TURBO digital twin system*: direct and reverse training path. The direct path is based on an AE that encodes the physical CDP $\mathbf{x}$ into the latent representation $\tilde{\mathbf{z}}$ end decodes back the physical CDP $\hat{\mathbf{x}}$. The reverse path is also based on an AE consisting of the same encoder and decoder but working in the reverse order. The reverse path AE encodes the digital template $\mathbf{z}$ into the physical CDP $\tilde{\mathbf{x}}$ and decodes back the digital template $\hat{\mathbf{z}}$. The encoder and decoder of direct and reverse paths are trained to maximize the mutual information between the encoded and reconstructed data with respect to the corresponding references.

framework [25];

- the demonstration of the link of the TURBO framework and Information Bottleneck and the state-of-the-art paired and unpaired image-to-image translation methods;
- the extended study of bidirectional generation between digital templates and CDP for images acquired by scanner, and two types of mobile phones iPhone 12 Pro (iOS) and Samsung Galaxy Note 20 Ultra (Android) with different imaging characteristics;
- investigation of the performance and sensitivity of the proposed Turbo framework to the data pre-processing and architectural particularities of TURBO encoders, decoders and losses.

Notations. We use the following notations: $\mathbf{z} \in \{0,1\}^{m\times m}$ denotes an original digital template of size $m\times m$; $\mathbf{x} \in [0,1]^{m\times m}$ corresponds to an image of original CDP, while $\mathbf{f} \in [0,1]^{m\times m}$ is used to denote an image of fake CDP; $\mathbf{y} \in [0,1]^{m\times m}$ stands for a probe that might be either original or fake. We use $\mathbb{E}_{p(\mathbf{x})}[.]$ to denote mathematical expectation with respect to a distribution $p(\mathbf{x})$, $D_{\mathrm{KL}}(.||.)$ denotes Kullback-Leibler (KL)-divergence and $I(.;.)$ stands for mutual information. We assume that a pair of digital template $\mathbf{z}$ and CDP $\mathbf{x}$ are distributed as $(\mathbf{x},\mathbf{z}) \sim p_{\mathrm{x,z}}(\mathbf{x},\mathbf{z})$ or simply as $p(\mathbf{x},\mathbf{z})$. We will use $\mathbf{z}$ and $\mathbf{x}$ to denote the real digital template and CDP, $\tilde{\mathbf{z}}$ and $\tilde{\mathbf{x}}$ their synthetic counterparts and $\hat{\mathbf{z}}$ and $\hat{\mathbf{x}}$ the reconstructed data from the corresponding synthetic counterparts $\tilde{\mathbf{x}}$ and $\tilde{\mathbf{z}}$. Accordingly, the marginal distributions of synthetic and reconstructed data are denoted as $\tilde{p}_\theta(\mathbf{x})$ and $\hat{p}_\theta(\mathbf{x})$ and $\tilde{q}_\phi(\mathbf{z})$ and $\hat{p}_\phi(\mathbf{z})$ for CDP and digital templates, respectively.

II. Printing-imaging channel in CDP authentication pipeline

The printing-imaging system under the study is shown in Fig. 1. The digital template $\mathbf{z}$ represents a random binary pattern with maximized entropy to complicate its prediction from the printed counterpart. It is generated from some key associated to a given physical object or can carry out some secretly encoded message. An industrial printer reproduces this digital template $\mathbf{z}$ on surface of an object that should be protected thus creating a CDP of the protected object. Alternatively, it can be reproduced on the object's packaging. An image of printed CDP can be acquired by some imaging device such as a phone. This image $\mathbf{x}$ can be stored in the database of the original CDP for further authentication.

It should be pointed out that printing and imaging represent a very complex physical channel from $\mathbf{z}$ to $\mathbf{x}$ with an unknown mathematical model and that it is also subject to various deviations during different sessions of printing and imaging. Thus, the development of an exact mathematical model of this printing-imaging process for each model of printer, substrate, phone, their settings and imaging conditions represents a very costly and time consuming problem. Therefore, it is not attractive for practical large scale applications. Instead, one can target to create a stochastic model $p(\mathbf{x}|\mathbf{z})$ of this channel or its parameterized version $p_\theta(\mathbf{x}|\mathbf{z})$ implemented in the form of a deep neural network with parameters θ.

Once objects protected by CDP are produced, the verifier can perform their authentication. A schematic diagram of the authentication process is shown in Fig. 2. Given a physical object protected by CDP, the counterfeiter will try to produce the closest replica of the original object using recent advancements in reproduction technologies. It also concerns reproduction or cloning of CDP using machine learning tools as demonstrated in [10]. The goal of the end-user (verifier) is to acquire a probe CDP $\mathbf{y}$ from the object under verification and to run an authentication test producing a decision in favor of the hypothesis $\mathcal{H}_1$ for the authentic object and $\mathcal{H}_0$ for the fake one. The authentication procedure might be based on various statistical tests that differentiate the features of original and faked CDP [1], [2], [8], [12]. Recently it was shown that the authentication can be performed on a generalized approach based on the digital template $\mathbf{z}$, physical template $\mathbf{x}$ or synthetic physical template $\tilde{\mathbf{x}}$ [9], [26] as shown in Fig. 2.

Authentication based on synthetic template has many advantages in large scale practical applications since it does not require the acquisition of CDP images from each physical

object while producing an authentication accuracy close to the one based on the physical templates [9]. At the same time, the synthetic physical template $\tilde{\mathbf{x}}$ should be generated from the digital template $\mathbf{z}$ that in turns requires an accurate simulator capable to simulate the printing-imaging channel. Additionally, the training of an accurate authentication system based on two-class or one-class classifiers might require a lot of training data acquired from both original and faked objects. This might be very costly or sometimes impossible in practice. That is why the development of digital twin system of printing-imaging channel looks very attractive.

III. Digital twin based on TURBO system

Since the exact mathematical model of the considered printing-imaging channel is unknown, we will instead proceed with its digital twin system. This digital twin system is shown in Fig. 1. It can be considered as two channels: (a) a direct printing-imaging channel $p_\theta(\mathbf{x}|\mathbf{z})$ producing synthetic CDP samples $\tilde{\mathbf{x}}$ from the digital template $\mathbf{z}$ and (b) an inverse printing-imaging channel $q_\phi(\mathbf{z}|\mathbf{x})$ producing synthetic digital template estimates $\tilde{\mathbf{z}}$ from physical CDP $\mathbf{x}$. These channels can be stochastic or deterministic $p_\theta(\mathbf{x}|\mathbf{z}) = \delta(\mathbf{x} - g_\theta(\mathbf{z}))$ and $q_\phi(\mathbf{z}|\mathbf{x}) = \delta(\mathbf{z} - f_\phi(\mathbf{x}))$, where $\delta(.)$ stands for the delta-function and $g_\theta(.)$ and $f_\phi(.)$ denote deterministic parameterized neural networks with the corresponding parameters θ and ϕ. The presence of direct and inverse digital twin channels can be very useful for both defender and attacker. The defender can use the direct channel for the production of synthetic physical templates for the authentication as considered in Fig. 2. Additionally, the synthetic templates can be considered as augmentation for the training of classifiers. The attacker can use the inverse channel for the estimation of digital templates from the scanned CDP of original objects with their following integration into the faked objects. Furthermore, the fact that the digital twin is fully differentiable opens for the attacker a possibility to design adversarial attacks for the physical world applications. The adversarial attacks against CDP in the physical world remain an open and little studied problem for now.

In this paper, we will consider a generalized approach named TURBO to the training of $p_\theta(\mathbf{x}|\mathbf{z})$ and $q_\phi(\mathbf{z}|\mathbf{x})$. This includes the training of both $p_\theta(\mathbf{x}|\mathbf{z})$ and $q_\phi(\mathbf{z}|\mathbf{x})$ simultaneously. Then a particular use of each digital twin depends on the targeted application.

A. The intuition behind TURBO system

The proposed digital twin system is based on an AE architecture and consists of stochastic encoder $q_\phi(\mathbf{z}|\mathbf{x})$ and decoder $p_\theta(\mathbf{x}|\mathbf{z})$ that are deep networks parametrized by the parameters ϕ and θ, respectively. The block diagram of the TURBO system is shown in Fig. 3.

Since it is assumed that the CDP image $\mathbf{x}$ and template $\mathbf{z}$ follow the joint distribution $(\mathbf{x}, \mathbf{z}) \sim p_{\mathrm{x,z}}(\mathbf{x}, \mathbf{z})$, it is natural to consider a generative process in two ways, i.e., the encoding and decoding of $\mathbf{x}$ and the encoding and decoding of $\mathbf{z}$. Such a consideration is justified by two ways of decomposition of the joint distribution based on the chain rule: (a) the direct way $p_{\mathrm{x,z}}(\mathbf{x}, \mathbf{z}) = p_{\mathbf{x}}(\mathbf{x})p(\mathbf{z}|\mathbf{x})$ and (b) the reverse way $p_{\mathrm{x,z}}(\mathbf{x}, \mathbf{z}) = p_{\mathbf{z}}(\mathbf{z})p(\mathbf{x}|\mathbf{z})$.

In contrast to this interpretation, classical AEs such as variational AE (VAE) [27], [28] or adversarial AE (AAE) [29] consider only the direct path, i.e., the encoding and decoding with respect to $\mathbf{x}$ only. Additionally, in contrast to these classical AEs, where the latent space of the encoder $q_\phi(\mathbf{z}|\mathbf{x})$ represented by $\mathbf{z}$ is considered to be a non-physical data governed by some easy to analyze and easy to sample from distribution $p(\mathbf{z})$ typically selected to be Gaussian probability density function (pdf), the TURBO framework considers the representation $\mathbf{z}$ to be statistically related to $\mathbf{x}$ according to $p_{\mathrm{x,z}}(\mathbf{x}, \mathbf{z})$. That is why the encoder $q_\phi(\mathbf{z}|\mathbf{x})$ and decoder $p_\theta(\mathbf{x}|\mathbf{z})$ that form the TURBO architecture are trained in two ways referred to as *direct path* and *reverse path*[1]. At the direct path, the encoder $q_\phi(\mathbf{z}|\mathbf{x})$ receives $\mathbf{x}$ as an input and produces an estimation of $\tilde{\mathbf{z}}$ as close as possible to the corresponding $\mathbf{z}$. The decoder receives $\tilde{\mathbf{z}}$ as input and produces a reconstructed version $\hat{\mathbf{x}}$ as close as possible to $\mathbf{x}$. Mutual information is used to measure the correspondence between the pair of $\tilde{\mathbf{z}}$ and $\mathbf{z}$ and the pair of $\hat{\mathbf{x}}$ and $\mathbf{x}$ as shown in Fig. 3. At the reverse path, the decoder $p_\theta(\mathbf{x}|\mathbf{z})$ receives $\mathbf{z}$ as an input and generates $\tilde{\mathbf{x}}$ as a new latent representation to be as close as possible to $\mathbf{x}$ and the encoder $q_\phi(\mathbf{z}|\mathbf{x})$ produces a reconstructed version of $\hat{\mathbf{z}}$ as close as possible to $\mathbf{z}$. Mutual information should ensure the correspondence between the pair of $\tilde{\mathbf{x}}$ and $\mathbf{x}$ and the pair of $\hat{\mathbf{z}}$ and $\mathbf{z}$.

B. Definition of TURBO loss

To train the encoder's $q_\phi(\mathbf{z}|\mathbf{x})$ and decoder's $p_\theta(\mathbf{x}|\mathbf{z})$ parameters ϕ, θ, the TURBO loss is defined as:

$$\mathcal{L}_{\text{TURBO}}\left(\phi, \theta\right) = \mathcal{L}^{\text{Direct}}\left(\phi, \theta\right) + \lambda_T \mathcal{L}^{\text{Reverse}}\left(\phi, \theta\right), \quad (1)$$

where λ_T is a trade-off parameter between the two terms and the direct path loss $\mathcal{L}^{\text{Direct}}\left(\phi, \theta\right)$ and the reverse path loss $\mathcal{L}^{\text{Reverse}}\left(\phi, \theta\right)$ are:

$$\mathcal{L}^{\text{Direct}}\left(\phi, \theta\right) = -\mathcal{I}^{\mathrm{z}}_{\phi}(\mathbf{X}; \mathbf{Z}) - \lambda_D \mathcal{I}^{\mathrm{x}}_{\phi,\theta}(\mathbf{X}; \tilde{\mathbf{Z}}), \quad (2)$$

$$\mathcal{L}^{\text{Reverse}}\left(\phi, \theta\right) = -\mathcal{I}^{\mathrm{x}}_{\theta}(\mathbf{X}; \mathbf{Z}) - \lambda_R \mathcal{I}^{\mathrm{z}}_{\phi,\theta}(\tilde{\mathbf{X}}; \mathbf{Z}). \quad (3)$$

The terms $\mathcal{I}^{\mathrm{z}}_{\phi}(\mathbf{X}; \mathbf{Z})$ and $\mathcal{I}^{\mathrm{x}}_{\phi,\theta}(\mathbf{X}; \tilde{\mathbf{Z}})$ impose the constraints on the latent and reconstruction spaces of the direct path, respectively, and symmetrically the terms $\mathcal{I}^{\mathrm{x}}_{\theta}(\mathbf{X}; \mathbf{Z})$ and $\mathcal{I}^{\mathrm{z}}_{\phi,\theta}(\tilde{\mathbf{X}}; \mathbf{Z})$ impose the constraints on the latent and reconstruction spaces of the reverse path. These terms will be introduced below. The parameters λ_D and λ_R trade-off the latent and reconstruction spaces' constraints in the direct and reverse paths.

The training of the TURBO system is considered as a maximization of mutual information problem, which translates into the minimization of loss problem:

$$(\hat{\phi}, \hat{\theta}) = \arg\min_{\phi,\theta} \mathcal{L}_{\text{TURBO}}\left(\phi, \theta\right). \quad (4)$$

The TURBO framework assumes the training of the encoder $q_\phi(\mathbf{z}|\mathbf{x})$ and decoder $p_\theta(\mathbf{x}|\mathbf{z})$ to maximize the mutual

[1]One should distinguish the direct and inverse digital twin printing-imaging channels shown in Fig. 1 from the direct and reverse paths of TURBO training as shown in Fig. 3.

information between the predicted and physically observable components. Since the practical computation of mutual information is challenging, we provide variational approximations to the mutual information terms as described in Appendices A and B. The TURBO loss contains four variational terms as defined in (2) and (3).

1) The loss of the direct path: The first term $\mathcal{I}_{\phi}^{\mathrm{z}}(\mathbf{X};\mathbf{Z})$ of the direct path (2) represents a variational approximation to the mutual information between the estimated $\tilde{\mathbf{z}}$ obtained from $\mathbf{x} \sim p_{\mathrm{x}}(\mathbf{x})$ via the encoder $q_{\phi}(\mathbf{z}|\mathbf{x})$ and true template $\mathbf{z}$ corresponding to the pair $\{\mathbf{x},\mathbf{z}\} \sim p_{\mathrm{x,z}}(\mathbf{x},\mathbf{z})$ as shown in Fig. 3. The variational approximation to this mutual information introduced in Appendix A is defined as:

$$\mathcal{I}_{\phi}^{\mathrm{z}}(\mathbf{X};\mathbf{Z}) := \underbrace{\mathbb{E}_{p(\mathbf{x},\mathbf{z})}\left[\log q_{\phi}(\mathbf{z}|\mathbf{x})\right]}_{-\mathcal{L}_{\tilde{\mathrm{z}}}(\mathbf{z},\tilde{\mathbf{z}})} - \underbrace{D_{\mathrm{KL}}\left(p_{\mathrm{z}}(\mathbf{z})\|\tilde{q}_{\phi}(\mathbf{z})\right)}_{\mathcal{D}_{\tilde{\mathrm{z}}}(\mathbf{z},\tilde{\mathbf{z}})}. \tag{5}$$

The first term represents the conditional cross-entropy for the distribution $q_{\phi}(\mathbf{z}|\mathbf{x})$ and the second term stands for the KL-divergence denoted as $\mathcal{D}_{\tilde{\mathrm{z}}}(\mathbf{z},\tilde{\mathbf{z}})$ between the marginal distribution of true data $p_{\mathrm{z}}(\mathbf{z})$ and marginal distribution $\tilde{q}_{\phi}(\mathbf{z})$ of the output of the encoder. One can assume that $q_{\phi}(\mathbf{z}|\mathbf{x}) \propto \exp\left(-\alpha \left\|\mathbf{z} - f_{\boldsymbol{\phi}}(\mathbf{x})\right\|_1\right)$ follows a Laplacian distribution with a scale parameter α, $\|.\|_1$ denoting the ℓ_1-norm and $f_{\boldsymbol{\phi}}(.)$ representing a parametrized encoder network. Thus, one can define a pair-wise estimation loss $\mathcal{L}_{\tilde{\mathrm{z}}}(\mathbf{z},\tilde{\mathbf{z}}) := \alpha\mathbb{E}_{p_{\mathrm{x,z}}(\mathbf{x},\mathbf{z})}\left[\left\|\mathbf{z} - f_{\boldsymbol{\phi}}(\mathbf{x})\right\|_1\right]^2$. Therefore, by maximizing the variational approximation term $\mathcal{I}_{\phi}^{\mathrm{z}}(\mathbf{X};\mathbf{Z})$ at the direct path, one minimizes the pair-wise estimation loss $\mathcal{L}_{\tilde{\mathrm{z}}}(\mathbf{z},\tilde{\mathbf{z}})$ and KL-divergence $\mathcal{D}_{\tilde{\mathrm{z}}}(\mathbf{z},\tilde{\mathbf{z}})$ in (5).

The second term $\mathcal{I}_{\phi,\theta}^{\mathrm{x}}(\mathbf{X};\tilde{\mathbf{Z}})$ of the direct path ensures the reconstruction of data $\hat{\mathbf{x}}$ from the latent representation $\tilde{\mathbf{z}}$, i.e., we consider a chain $\mathbf{x} \to \tilde{\mathbf{z}} \to \hat{\mathbf{x}}$ that corresponds to Fig. 3. This term corresponds to a variation approximation of mutual information between the true $\mathbf{x}$ and its reconstructed version $\hat{\mathbf{x}}$ as developed in Appendix A:

$$\begin{aligned}\mathcal{I}_{\phi,\theta}^{\mathrm{x}}(\mathbf{X};\tilde{\mathbf{Z}}) := &\underbrace{\mathbb{E}_{p_{\mathrm{x}}(\mathbf{x})}\left[\mathbb{E}_{q_{\phi}(\mathbf{z}|\mathbf{x})}\left[\log p_{\theta}(\mathbf{x}|\mathbf{z})\right]\right]}_{-\mathcal{L}_{\hat{\mathrm{x}}}(\mathbf{x},\hat{\mathbf{x}})} \\ &- \underbrace{D_{\mathrm{KL}}\left(p_{\mathrm{x}}(\mathbf{x})\|\hat{p}_{\theta}(\mathbf{x})\right)}_{\mathcal{D}_{\hat{\mathrm{x}}}(\mathbf{x},\hat{\mathbf{x}})}.\end{aligned} \tag{6}$$

The first term of (6) represents the conditional cross-entropy and can be practically computed similarly to the reconstruction loss considered above. One can assume that $p_{\theta}(\mathbf{x}|\mathbf{z}) \propto \exp\left(-\beta \left\|\mathbf{x} - g_{\boldsymbol{\theta}}(\mathbf{z})\right\|_1\right)$ with a scale parameter β and $g_{\boldsymbol{\theta}}(.)$ representing a parametrized decoder network. In this case, one can define a pair-wise reconstruction loss $\mathcal{L}_{\hat{\mathrm{x}}}(\mathbf{x},\hat{\mathbf{x}}) := \beta\mathbb{E}_{p_{\mathrm{x}}(\mathbf{x})}\left[\mathbb{E}_{q_{\phi}(\mathbf{z}|\mathbf{x})}\left[\left\|\mathbf{x} - g_{\boldsymbol{\theta}}(\mathbf{z})\right\|_1\right]\right]$. The KL-divergence term denoted as $\mathcal{D}_{\hat{\mathrm{x}}}(\mathbf{x},\hat{\mathbf{x}})$ ensures the proximity between the distributions of true data $p_{\mathrm{x}}(\mathbf{x})$ and reconstructed data $\hat{p}_{\theta}(\mathbf{x})$. Thus, the maximization of $\mathcal{I}_{\phi,\theta}^{\mathrm{x}}(\mathbf{X};\tilde{\mathbf{Z}})$ in the direct loss (2) corresponds to the minimization of the reconstruction loss $\mathcal{L}_{\hat{\mathrm{x}}}(\mathbf{x},\hat{\mathbf{x}})$ and the KL-divergence $\mathcal{D}_{\hat{\mathrm{x}}}(\mathbf{x},\hat{\mathbf{x}})$.

[2]In case of Gaussian assumption on the estimation error between $\mathbf{z}$ and $\tilde{\mathbf{z}} = f_{\phi}(\mathbf{x})$, the ℓ_2-norm will be used instead of ℓ_1-norm.

2) The loss of the reverse path: The reverse path of TURBO uses the same encoder and decoder as at the direct path but operates in the reverse order, i.e., the AEs' input is $\mathbf{z}$ instead of $\mathbf{x}$ as for the direct path. Therefore, the first term $\mathcal{I}_{\theta}^{\mathrm{x}}(\mathbf{X};\mathbf{Z})$ of the reverse path in (3) is defined in Appendix B as:

$$\mathcal{I}_{\theta}^{\mathrm{x}}(\mathbf{X};\mathbf{Z}) := \underbrace{\mathbb{E}_{p(\mathbf{x},\mathbf{z})}\left[\log p_{\theta}(\mathbf{x}|\mathbf{z})\right]}_{-\mathcal{L}_{\tilde{\mathrm{x}}}(\mathbf{x},\tilde{\mathbf{x}})} - \underbrace{D_{\mathrm{KL}}\left(p_{\mathrm{x}}(\mathbf{x})\|\tilde{p}_{\theta}(\mathbf{x})\right)}_{\mathcal{D}_{\tilde{\mathrm{x}}}(\mathbf{x},\tilde{\mathbf{x}})}. \tag{7}$$

Similarly to the direct path considered above, the first term of (7) represents the conditional entropy and can be practically considered as a pair-wise loss $\mathcal{L}_{\tilde{\mathrm{x}}}(\mathbf{x},\tilde{\mathbf{x}}) := \gamma\mathbb{E}_{p_{\mathrm{x,z}}(\mathbf{x},\mathbf{z})}\left[\left\|\mathbf{x} - g_{\boldsymbol{\theta}}(\mathbf{z})\right\|_1\right]$ under the Laplacian assumption $p_{\theta}(\mathbf{x}|\mathbf{z}) \propto \exp\left(-\gamma \left\|\mathbf{x} - g_{\boldsymbol{\theta}}(\mathbf{z})\right\|_1\right)$ with a scale parameter γ. The second term denoted as $\mathcal{D}_{\tilde{\mathrm{x}}}(\mathbf{x},\tilde{\mathbf{x}})$ represents the KL-divergence between the marginal distribution $p_{\mathrm{x}}(\mathbf{x})$ and marginal distribution of estimated data $\tilde{p}_{\theta}(\mathbf{x})$.

Finally, the second term $\mathcal{I}_{\phi,\theta}^{\mathrm{z}}(\tilde{\mathbf{X}};\mathbf{Z})$ of the reverse path in the loss (3) ensures the reconstruction of $\hat{\mathbf{z}}$ from the latent representation $\tilde{\mathbf{x}}$ according to a considered chain $\mathbf{z} \to \tilde{\mathbf{x}} \to \hat{\mathbf{z}}$ that corresponds to Fig. 3. This term corresponds to a variational approximation of mutual information between the true $\mathbf{z}$ and its reconstructed version $\hat{\mathbf{z}}$ as developed in Appendix B:

$$\begin{aligned}\mathcal{I}_{\phi,\theta}^{\mathrm{z}}(\tilde{\mathbf{X}};\mathbf{Z}) := &\underbrace{\mathbb{E}_{p_{\mathbf{z}}(\mathbf{z})}\left[\mathbb{E}_{p_{\theta}(\mathbf{x}|\mathbf{z})}\left[\log q_{\phi}(\mathbf{z}|\mathbf{x})\right]\right]}_{-\mathcal{L}_{\hat{\mathrm{z}}}(\mathbf{z},\hat{\mathbf{z}})} \\ &- \underbrace{D_{\mathrm{KL}}\left(p_{\mathrm{z}}(\mathbf{z})\|\hat{q}_{\phi}(\mathbf{z})\right)}_{\mathcal{D}_{\hat{\mathrm{z}}}(\mathbf{z},\hat{\mathbf{z}})}.\end{aligned} \tag{8}$$

The first term of (8) represents the conditional cross-entropy that can be practically computed as $\mathcal{L}_{\hat{\mathrm{z}}}(\mathbf{z},\hat{\mathbf{z}}) := \nu\mathbb{E}_{p_{\mathrm{z}}(\mathbf{z})}\left[\mathbb{E}_{p_{\theta}(\mathbf{x}|\mathbf{z})}\left[\left\|\mathbf{z} - f_{\boldsymbol{\phi}}(\mathbf{x})\right\|_1\right]\right]$ under the assumption of Laplacian distribution of reconstruction error $q_{\phi}(\mathbf{z}|\mathbf{x}) \propto \exp\left(-\nu \left\|\mathbf{z} - f_{\boldsymbol{\phi}}(\mathbf{x})\right\|_1\right)$ with a scale parameter ν. The KL-divergence term denoted as $\mathcal{D}_{\hat{\mathrm{z}}}(\mathbf{z},\hat{\mathbf{z}})$ ensures the proximity between the distributions of true data $p_{\mathrm{z}}(\mathbf{z})$ and reconstructed data $\hat{q}_{\phi}(\mathbf{z})$. Accordingly, the maximization of $\mathcal{I}_{\phi,\theta}^{\mathrm{z}}(\tilde{\mathbf{X}};\mathbf{Z})$ in the reverse loss (3) corresponds to the minimization of the reconstruction loss $\mathcal{L}_{\hat{\mathrm{z}}}(\mathbf{z},\hat{\mathbf{z}})$ and the KL-divergence $\mathcal{D}_{\hat{\mathrm{z}}}(\mathbf{z},\hat{\mathbf{z}})$.

Therefore, the complete direct and reverse losses of the TURBO framework are:

$$\begin{aligned}\mathcal{L}^{\text{Direct}}(\phi,\theta) = &\mathcal{L}_{\tilde{\mathrm{z}}}(\mathbf{z},\tilde{\mathbf{z}}) + \mathcal{D}_{\tilde{\mathrm{z}}}(\mathbf{z},\tilde{\mathbf{z}}) \\ &+ \lambda_D\mathcal{L}_{\hat{\mathrm{x}}}(\mathbf{x},\hat{\mathbf{x}}) + \lambda_D\mathcal{D}_{\hat{\mathrm{x}}}(\mathbf{x},\hat{\mathbf{x}}),\end{aligned} \tag{9}$$

$$\begin{aligned}\mathcal{L}^{\text{Reverse}}(\phi,\theta) = &\mathcal{L}_{\tilde{\mathrm{x}}}(\mathbf{x},\tilde{\mathbf{x}}) + \mathcal{D}_{\tilde{\mathrm{x}}}(\mathbf{x},\tilde{\mathbf{x}}) \\ &+ \lambda_R\mathcal{L}_{\hat{\mathrm{z}}}(\mathbf{z},\hat{\mathbf{z}}) + \lambda_R\mathcal{D}_{\hat{\mathrm{z}}}(\mathbf{z},\hat{\mathbf{z}}).\end{aligned} \tag{10}$$

C. Link to Information Bottleneck

The TURBO framework can be viewed from an information bottleneck (IBN) perspective, considering either the direct path or the reverse path. The IBN originally suggested by Tishby *et. al.* [30] assumes the minimization of mutual information between the input of system $\mathbf{x}$ and some latent representation $\mathbf{z}$ while preserving mutual information between $\mathbf{z}$ and some utility attribute $\mathbf{c}$ that can represent class labels, segmentation maps, etc. The IBN was extended to a variational formulation in [31] allowing to practically compute mutual information.

Finally, the IBN was generalized to AE formulation [32] to address the self-encoding and reconstruction. This allowed to generalize a family of AE models such as VAE [27], [28], β-VAE [33], InfoVAE [34] and others based on a concept of bounded information bottleneck AE (BIB-AE) [32]. Along the same way, the AE formulation was also extended to semi-supervised learning [35] that allowed to generalize self-encoding and classification systems such as CatGAN [36], VAE (M1 + M2) [37] and SeGMA [38].

Besides some remarkable similarities between the TURBO and the IBN frameworks, there are a number of fundamental differences that can be summarized as follows:

- **optimization objectives**: IBN targets to compress the latent representation using *minimization* of mutual information between the system input and latent representation which follows a chosen distribution $p_{\mathrm{z}}(\mathbf{z})$ while TURBO targets *maximization* of the above mutual information with the distribution that follows from the joint pdf $p_{\mathrm{x},\mathrm{z}}(\mathbf{x},\mathbf{z})$, i.e., from the physical observation model;
- **physically meaningful latent space**: the latent space distribution of IBN is chosen to be easy to sample from and to analyse in terms of calculation of KL-divergence as for example in VAE, i.e., it does not bring any physical interpretation, while the latent space of TURBO is considered as a physically observable representation matched in dimensions and statistical behavior with the corresponding variables;
- **two-way consideration**: IBN typically considers only one directional path to train a model consisting of the encoder and decoder while TURBO has two paths each for its own observable data but with a common model.

D. Link to the state-of-the-art translation methods

The proposed TURBO model can also be considered in the scope of the image-to-image translation problem for two paths. Along this consideration, TURBO generalizes models such as pix2pix [39] and CycleGAN [40] and is conceptually linked to Contrastive Unpaired Translation (CUT) model [41].

1) Paired setup: pix2pix [39] is a representative of paired setup, i.e., when training data is represented by N paired samples $\{\mathbf{x}_i,\mathbf{z}_i\}_{i=1}^N$. The goal of pix2pix image-to-image translation is to train a model $q_\phi(\mathbf{z}|\mathbf{x}) = \delta(\mathbf{z} - f_\phi(\mathbf{x}))$ that would produce an estimate $\tilde{\mathbf{z}} = f_\phi(\mathbf{x})$. Since the considered setup is paired, the reconstruction loss $\mathcal{L}_{\tilde{\mathrm{z}}}(\mathbf{z},\tilde{\mathbf{z}})$ is imposed along the distribution matching loss $\mathcal{D}_{\tilde{\mathrm{z}}}(\mathbf{z},\tilde{\mathbf{z}})$ a.k.a. adversarial loss requiring correspondence between the distributions $\tilde{q}_\phi(\mathbf{z})$ and $p_{\mathrm{z}}(\mathbf{z})$. Symmetrically, one can consider the translation problem from $\mathbf{z}$ to $\mathbf{x}$.

The pix2pix [39] image-to-image translation model can be viewed as a particular case of the TURBO approach where only the encoder $q_\phi(\mathbf{z}|\mathbf{x})$ is used for the direct path (2) and $\lambda_D = 0$ or where only the decoder $p_\theta(\mathbf{x}|\mathbf{z})$ is used for the reverse path (3) and $\lambda_R = 0$. Accordingly, the direct and reverse formulations of pix2pix [39] can be written as:

$$\mathcal{L}^{\mathrm{Direct}}_{\mathrm{pix2pix}}(\phi) = \mathcal{L}_{\tilde{\mathrm{z}}}(\mathbf{z},\tilde{\mathbf{z}}) + \mathcal{D}_{\tilde{\mathrm{z}}}(\mathbf{z},\tilde{\mathbf{z}}), \tag{11}$$

$$\mathcal{L}^{\mathrm{Reverse}}_{\mathrm{pix2pix}}(\theta) = \mathcal{L}_{\tilde{\mathrm{x}}}(\mathbf{x},\tilde{\mathbf{x}}) + \mathcal{D}_{\tilde{\mathrm{x}}}(\mathbf{x},\tilde{\mathbf{x}}). \tag{12}$$

One advantage of the TURBO model compared to the pix2pix is simultaneous training in both directions at once, making the encoder and decoder consistent with each other. Also, we link networks to the information-theoretic framework and add cycle-consistency losses.

A popular super-resolution framework SRGAN [42] can be considered as a particular case of one-way TURBO framework (11). Finally, one can also consider only the paired loss $\mathcal{L}_{\tilde{\mathrm{x}}}(\mathbf{x},\tilde{\mathbf{x}})$ while skipping the distribution matching part $\mathcal{D}_{\tilde{\mathrm{x}}}(\mathbf{x},\tilde{\mathbf{x}})$ for simplicity of training.

2) Unpaired setup: CycleGAN [40] image-to-image translation is developed for those cases when only two sets of unpaired data $\{\mathbf{x}_i\}_{i=1}^N$ and $\{\mathbf{z}_j\}_{j=1}^M$ are available for training. In this situation, a cycle consistency is needed to ensure the encoding of $\mathbf{x}$ into $\tilde{\mathbf{z}}$ and back decoding to $\hat{\mathbf{x}}$ for the direct path and the encoding of $\mathbf{z}$ into $\tilde{\mathbf{x}}$ and back decoding to $\hat{\mathbf{z}}$ for the reverse one. The accuracy of encoding and decoding is ensured by two cycle reconstruction losses corresponding to $\mathcal{L}_{\hat{\mathrm{x}}}(\mathbf{x},\hat{\mathbf{x}})$ and $\mathcal{L}_{\hat{\mathrm{z}}}(\mathbf{z},\hat{\mathbf{z}})$ in (9) and (10). This reflects the part of the "cycle" in CycleGAN.

Since the considered setup is unpaired, the distributions of latent representations are controlled by the KL-divergence terms $\mathcal{D}_{\tilde{\mathrm{z}}}(\mathbf{z},\tilde{\mathbf{z}})$ and $\mathcal{D}_{\tilde{\mathrm{x}}}(\mathbf{x},\tilde{\mathbf{x}})$ in (9) and (10) that corresponds to "GAN" type of constraints.

Therefore, the total loss of CycleGAN corresponds to a particular case of the TURBO framework:

$$\begin{aligned}\mathcal{L}_{\mathrm{CycleGAN}}(\phi,\theta) = \mathcal{D}_{\tilde{\mathrm{z}}}(\mathbf{z},\tilde{\mathbf{z}}) + \lambda_D\mathcal{L}_{\hat{\mathrm{x}}}(\mathbf{x},\hat{\mathbf{x}}) \\ + \lambda_T\mathcal{D}_{\tilde{\mathrm{x}}}(\mathbf{x},\tilde{\mathbf{x}}) + \lambda_T\lambda_R\mathcal{L}_{\hat{\mathrm{z}}}(\mathbf{z},\hat{\mathbf{z}}).\end{aligned} \tag{13}$$

While CycleGAN was created only for unpaired data, the TURBO model can be applied to the paired or even mixed setup as well. By including a discriminator's loss for reconstruction and pairwise estimation loss in the latent space it is possible to achieve more stable training and better performance.

Therefore, TURBO generalizes both paired and unpaired systems in both families of translation models.

Additionally, considering individually each path, one can notice that the TURBO setup also generalizes AAE [29], which imposes KL-divergence constraint on the distribution of the latent space and ensures the reconstruction constraint after the decoder.

Finally, one can also consider a link to contrastive unpaired translation (CUT) system [43], where the mutual information between the positive pairs is maximized. Conceptually, it corresponds to the direct path of Turbo where $\mathbf{x}$ is considered as the reference and $\mathbf{z}$ as its positive pair. At the same time, the mutual information decomposition of TURBO differs from those used in CUT, which is based on the InfoNCE framework [44]. Thus, these systems cannot be compared directly.

IV. Architectural and Training Details

The TURBO system offers a high level of flexibility, providing many degrees of freedom. In the current section, we describe different architectural solutions for encoder and decoder, adversarial losses and techniques for improving the stability of training. A full list of used hyperparameters is presented in Appendix D.

TABLE I
MODELS PERFORMANCE ON THE SCANNER DATA

Model	FID $_{\mathbf{x} \to \tilde{\mathbf{z}}}$ ↓	Hamming distance ↓	FID $_{\mathbf{z} \to \tilde{\mathbf{x}}}$ ↓	MSE ↓	SSIM↑
W/O processing	*304.13*	*0.24*	*304.01*	*0.181*	*0.48*
CUT	3.86	0.20	5.29	0.061	0.70
pix2pix	3.37	0.11	8.57	0.045	0.76
CycleGAN	3.87	0.15	**4.45**	0.049	0.73
TURBO$^{\text{unpaired}}_{\text{CNN-RESNET-CNN}}$	3.57	0.16	5.91	0.059	0.68
TURBO$^{\text{paired (w } \mathcal{D})}_{\text{UNET}}$	4.27	0.11	8.94	0.043	0.77
TURBO$^{\text{paired (w } \mathcal{D})}_{\text{CNN-RESNET-CNN}}$	**3.16**	**0.09**	6.60	0.040	**0.78**
Shallow TURBO$^{\text{paired (w/o } \mathcal{D})}_{\text{CNN-RESNET-CNN}}$	53.31	0.15	37.82	0.037	0.77
Deep TURBO$^{\text{paired (w/o } \mathcal{D})}_{\text{CNN-RESNET-CNN}}$	7.77	0.12	24.09	0.036	0.78
TURBO$^{\text{paired}}_{\text{UNET}}$	6.21	0.10	28.11	**0.036**	0.78
TURBO$^{\text{paired}}_{\text{Eff-UNET-2}}$	11.33	0.09	28.48	0.037	0.77
TURBO$^{\text{paired}}_{\text{Eff-UNET-7}}$	11.83	0.09	28.95	0.037	0.77

A. Structure of encoder and decoder

Our proposed approach does not enforce any specific constraints on the architecture of the encoder $q_\phi(\mathbf{z}|\mathbf{x})$ and decoder $p_\theta(\mathbf{x}|\mathbf{z})$, allowing for a wide range of technical choices. In our work, we have considered several most widely used architectures for encoders and decoders, namely:

- CNN-RESNET-CNN adapted from CycleGAN and StarGAN [45] models, consisting of two convolutional layers for downsampling, nine residual blocks [46], and two transposed convolutional layers for upsampling;
- UNET [47] with skip-connections layers;
- Eff-UNET [48] that uses the EfficientNet backbone [49] to extract high-level features from the input image and then using the UNET decoder for the reconstruction.

The instance normalization [50] was used to stabilize training together with Adam optimizer [51].

B. Adversarial loss and structure of discriminators

The choice of the adversarial loss function, which implements $D_{\text{KL}}(.||.)$ terms in (5)-(8), could be crucial for the success of the training for the considered models [52]. In our previous work [24], we examined three commonly used adversarial losses: LSGAN [53], HINGE [54] and WGAN [55] with gradient penalty [56]. Our experiments demonstrated that WGAN is the most efficient in achieving desirable outcomes. Consequently, in our current work, we exclusively use the WGAN adversarial loss.

Based on the investigation performed in [24], we use the ImageGAN discriminator based on residual networks [46].

C. Stability of training

Adversarial training with discriminators is known to be quite unstable due to the mode collapse and vanishing of gradients. To address these challenges, we employ several techniques to enhance training stability:

- First, we balance the iterations between the discriminator and the generator by controlling the number of discriminator iterations per generator iteration, denoted as n_{D}.
- Additionally, we update the discriminators using a history of generated images rather than solely relying on images generated at the last iteration [57]. This approach helps provide more diverse and informative training examples for the discriminators.
- To further enhance stability, we introduce occasional label flipping during discriminator training with a probability of p_{flip} [58]. This technique involves randomly changing the labels to create a more robust discriminator.
- Moreover, we incorporate artificial noise into the discriminator's inputs with a probability of p_{noise} and a weight w_{noise} [59]. This noise injection helps prevent the discriminator from overfitting and encourages it to learn more generalized representations.

By integrating these strategies, we have observed improved stability and performance in our training process.

V. EXPERIMENTAL RESULTS

A. Dataset

The experimental results are divided into two parts: (i) experiments conducted on data acquired by a high-resolution scanner and (ii) experiments conducted on data acquired by modern mobile phones.

The experiments on the scanner data are an extension of our previous work [24]. In this respect, the same Indigo 1×1 base dataset [10][3] was used. This dataset consists of 720 digital CDP of size 228×228 with 1×1 pixel symbol size. One pixel with the value 1 in the digital template represents a symbol of size 1×1 that corresponds to a printed black spot on the substrate of size approximately 30 μm in diameter when printed at the resolution of 812.8 dpi. No halftoning is applied during the printing. The digital CDP were printed at HP Indigo 7600 industrial printer at a resolution of 812.8 dpi and enrolled by Epson Perfection V850 Pro scanner at a resolution of 2400 ppi. Taking into account the printing and acquisition resolutions, the obtained CDP are of size 684×684 meaning that 1×1 pixel in a digital template corresponds to

[3]http://sip.unige.ch/projects/snf-it-dis/datasets/indigo-base

3×3 block in a printed CDP and the final codes are 16-bit gray-scaled images.

The experiments on the mobile phone data were performed on the recently created Indigo 1x1 variability dataset [11][4] that consists of 1440 digital CDP of size 228×228 with 1×1 pixel symbol size. The CDP were printed at HP Indigo 5500 industrial printer at a resolution of 812.8 dpi and enrolled by iPhone 12 Pro and Samsung Galaxy Note 20 Ultra cell phones. The obtained CDP are of size 228×228 and encoded as 8-bit RGB-images.

Both mentioned datasets contain the original and fake CDP. For our experiments, we used only the original codes. However, it should be noted that in both cases the fakes were produced on the same printing and acquisition equipment as the original codes. In this respect, the model trained on the original codes can be efficiently applied to generate fake codes.

B. Metrics

The following metrics were used to evaluate the quality of the synthesised twins:

- Hamming distance between the original digital template $\mathbf{z}$ and the binarized estimation $\tilde{\mathbf{z}}$.
- Mean square error (MSE) between the original printed code $\mathbf{x}$ and the synthesized twin $\tilde{\mathbf{x}}$.
- To address an issue that the MSE is not highly indicative of the perceived similarity of images, we calculate the Structural Similarity Index (SSIM) [60] between the original printed code $\mathbf{x}$ and the synthesized twin $\tilde{\mathbf{x}}$.
- Fréchet Inception Distance (FID): $\text{FID}_{\mathbf{z} \to \tilde{\mathbf{x}}}$ and $\text{FID}_{\mathbf{x} \to \tilde{\mathbf{z}}}$ proposed in [61]. Instead of a simple pixel-by-pixel comparison of images, FID estimates the mean and standard deviation of one of the deep layers in the pretrained convolutional neural network. It became one of the most widely used metric for image-to-image translation task. We suppose that the usage of deep network statistics can be helpful not only as a measure of human perception of image similarity but also to assess the difficulty of distinguishing the generated images from the real ones since the network activations are similar at a metric close to zero.

C. Setups under investigation

To demonstrate the flexibility of the proposed TURBO framework, we will consider both paired and unpaired setups. It might be trained in both direct and reverse paths simultaneously or only in one path, for example, from the template $\mathbf{z}$ to the printed code $\mathbf{x}$ (reverse path) or vice versa. At the same time, different encoder-decoder architectures might be used. In our experiments, we compared the performance of the paired and unpaired setups under the same encoder-decoder architecture and investigated the performance of the paired setup under different encoder-decoder architectures. More particularly, we study the following setups:

[4]http://sip.unige.ch/projects/snf-it-dis/datasets/indigo-variability

- $\text{TURBO}^{\text{unpaired}}_{\text{CNN-RESNET-CNN}}$ with the CNN-RESNET-CNN based encoder-decoder and the total loss:

$$\begin{aligned}\mathcal{L}^{\text{unpaired}}_{\text{CNN-RESNET-CNN}}(\phi,\theta) = \mathcal{D}_{\tilde{\mathbf{z}}}(\mathbf{z},\tilde{\mathbf{z}}) + \lambda_D\mathcal{L}_{\hat{\mathbf{x}}}(\mathbf{x},\hat{\mathbf{x}}) \\ + \lambda_D\mathcal{D}_{\hat{\mathbf{x}}}(\mathbf{x},\hat{\mathbf{x}}) + \lambda_T\mathcal{D}_{\tilde{\mathbf{x}}}(\mathbf{x},\tilde{\mathbf{x}}) \\ + \lambda_T\lambda_R\mathcal{L}_{\hat{\mathbf{z}}}(\mathbf{z},\hat{\mathbf{z}}) + \lambda_T\lambda_R\mathcal{D}_{\hat{\mathbf{z}}}(\mathbf{z},\hat{\mathbf{z}}).\end{aligned}$$

- $\text{TURBO}^{\text{paired (w } \mathcal{D})}_{\text{CNN-RESNET-CNN}}$ with the CNN-RESNET-CNN based encoder-decoder and the total loss:

$$\begin{aligned}\mathcal{L}^{\text{paired (w } \mathcal{D})}_{\text{CNN-RESNET-CNN}}(\phi,\theta) = \mathcal{L}_{\tilde{\mathbf{z}}}(\mathbf{z},\tilde{\mathbf{z}}) + \mathcal{D}_{\tilde{\mathbf{z}}}(\mathbf{z},\tilde{\mathbf{z}}) \\ + \lambda_D\mathcal{L}_{\hat{\mathbf{x}}}(\mathbf{x},\hat{\mathbf{x}}) + \lambda_D\mathcal{D}_{\hat{\mathbf{x}}}(\mathbf{x},\hat{\mathbf{x}}) \\ + \lambda_T\mathcal{L}_{\tilde{\mathbf{x}}}(\mathbf{x},\tilde{\mathbf{x}}) + \lambda_T\mathcal{D}_{\tilde{\mathbf{x}}}(\mathbf{x},\tilde{\mathbf{x}}) \\ + \lambda_T\lambda_R\mathcal{L}_{\hat{\mathbf{z}}}(\mathbf{z},\hat{\mathbf{z}}) + \lambda_T\lambda_R\mathcal{D}_{\hat{\mathbf{z}}}(\mathbf{z},\hat{\mathbf{z}}).\end{aligned}$$

- $\text{TURBO}^{\text{paired (w/o } \mathcal{D})}_{\text{CNN-RESNET-CNN}}$ shallow and deep (14'936 and 8'402'304 models' parameters) with the CNN-RESNET-CNN based encoder-decoder and the total loss:

$$\begin{aligned}\mathcal{L}^{\text{paired (w/o } \mathcal{D})}_{\text{CNN-RESNET-CNN}}(\phi,\theta) = \mathcal{L}_{\tilde{\mathbf{z}}}(\mathbf{z},\tilde{\mathbf{z}}) + \lambda_D\mathcal{L}_{\hat{\mathbf{x}}}(\mathbf{x},\hat{\mathbf{x}}) \\ + \lambda_T\mathcal{L}_{\tilde{\mathbf{x}}}(\mathbf{x},\tilde{\mathbf{x}}) + \lambda_T\lambda_R\mathcal{L}_{\hat{\mathbf{z}}}(\mathbf{z},\hat{\mathbf{z}}).\end{aligned}$$

- $\text{TURBO}^{\text{paired}}_{\text{UNET}}$ with the UNET based encoder-decoder and the same loss as $\text{TURBO}^{\text{paired (w/o } \mathcal{D})}_{\text{CNN-RESNET-CNN}}$.
- $\text{TURBO}^{\text{paired}}_{\text{Eff-UNET-2}}$ with the Eff-UNET-2 based encoder-decoder and the same loss as $\text{TURBO}^{\text{paired (w/o } \mathcal{D})}_{\text{CNN-RESNET-CNN}}$.
- $\text{TURBO}^{\text{paired}}_{\text{Eff-UNET-7}}$ with the Eff-UNET-7 based encoder-decoder and the same loss as $\text{TURBO}^{\text{paired (w/o } \mathcal{D})}_{\text{CNN-RESNET-CNN}}$.
- W/O processing setup is used to estimate the baseline performance where we assume $\tilde{\mathbf{z}} = \mathbf{x}$ in the direct path and, in the reverse path, an ideal printing-imaging without any distortions, i.e., $\tilde{\mathbf{x}} = \mathbf{z}$.

D. Analysis of obtained results

We perform the investigation of the proposed TURBO framework in different configurations on CDP enrolled by the scanner and mobile phones. From the point of view of synthetic code generation the scanner data are more challenging since they contain more fine details, compared to the mobile data. From the point of view of verification by the end customers, the mobile data present a bigger value as a more practical scenario. From the point of view of the attacker, the scanner data is a more practical scenario since it allows them to perform a more accurate estimation of the original digital templates. Thus, both scanner and mobile data have important practical significance.

It should also be noted that these results were obtained without any post-processing and are the direct output of deep networks. Additional post-processing might increase the accuracy of digital template estimation and generation. However, to preserve the scalability to any artwork and fair comparison, we report all results without any refinements.

1) Scanner data: The results obtained on the scanner data are given in Table I. Comparing the unpaired and paired setups it should be noted that the paired setup is better for the estimation of the digital template $\mathbf{z}$ for both considered metrics, i.e., $\text{FID}_{\mathbf{x} \to \tilde{\mathbf{z}}}$ and Hamming distance. For the printed

TABLE II
MODELS PERFORMANCE ON THE IPHONE DATA

	Model	FID $_{\mathbf{x} \to \tilde{\mathbf{z}}}$ ↓	Hamming distance ↓	FID $_{\mathbf{z} \to \tilde{\mathbf{x}}}$ ↓	MSE ↓	SSIM↑
Non-normalized	*W/O processing*	*288.09*	*0.30*	*288.08*	*0.257*	*0.23*
	pix2pix	13.61	0.23	12.76	0.006	0.90
	CycleGAN	24.78	0.27	14.17	0.015	0.72
	$\text{TURBO}^{\text{unpaired}}_{\text{CNN-RESNET-CNN}}$	10.30	0.27	17.13	0.017	0.69
	$\text{TURBO}^{\text{paired (w } \mathcal{D})}_{\text{UNET}}$	7.36	0.23	11.72	0.005	0.91
	$\text{TURBO}^{\text{paired (w } \mathcal{D})}_{\text{CNN-RESNET-CNN}}$	**6.83**	0.24	**10.78**	0.005	0.91
	Shallow $\text{TURBO}^{\text{paired (w/o } \mathcal{D})}_{\text{CNN-RESNET-CNN}}$	113.62	0.23	50.82	0.005	0.90
	Deep $\text{TURBO}^{\text{paired (w/o } \mathcal{D})}_{\text{CNN-RESNET-CNN}}$	11.88	0.22	17.32	0.005	0.92
	$\text{TURBO}^{\text{paired}}_{\text{UNET}}$	36.90	**0.21**	16.93	**0.005**	**0.92**
	$\text{TURBO}^{\text{paired}}_{\text{Eff-UNET-2}}$	62.76	0.21	18.95	0.005	0.92
	$\text{TURBO}^{\text{paired}}_{\text{Eff-UNET-7}}$	60.66	0.21	18.30	0.005	0.92
Normalized	*W/O processing*	*289.68*	*0.30*	*289.68*	*0.254*	*0.25*
	pix2pix	11.82	0.23	11.64	0.005	0.91
	CycleGAN	20.69	0.27	12.59	0.014	0.78
	$\text{TURBO}^{\text{unpaired}}_{\text{CNN-RESNET-CNN}}$	11.73	0.28	14.71	0.020	0.70
	$\text{TURBO}^{\text{paired (w } \mathcal{D})}_{\text{UNET}}$	6.81	0.23	12.45	0.005	0.92
	$\text{TURBO}^{\text{paired (w } \mathcal{D})}_{\text{CNN-RESNET-CNN}}$	**6.56**	0.24	**10.20**	0.005	0.91
	Shallow $\text{TURBO}^{\text{paired (w/o } \mathcal{D})}_{\text{CNN-RESNET-CNN}}$	113.62	0.23	48.75	0.005	0.91
	Deep $\text{TURBO}^{\text{paired (w/o } \mathcal{D})}_{\text{CNN-RESNET-CNN}}$	11.85	0.22	13.76	0.005	0.92
	$\text{TURBO}^{\text{paired}}_{\text{UNET}}$	35.35	**0.21**	12.22	**0.004**	**0.92**
	$\text{TURBO}^{\text{paired}}_{\text{Eff-UNET-2}}$	61.27	0.21	17.02	0.004	0.92
	$\text{TURBO}^{\text{paired}}_{\text{Eff-UNET-7}}$	64.58	0.21	16.64	0.005	0.92

twins generation, the paired setup shows better results in terms of the pixel metrics, i.e., MSE and SSIM, while the unpaired setup achieves better generalization in terms of non-pixel FID metric. Comparing the $\text{TURBO}^{\text{paired (w } \mathcal{D})}_{\text{CNN-RESNET-CNN}}$ and $\text{TURBO}^{\text{paired (w/o } \mathcal{D})}_{\text{CNN-RESNET-CNN}}$ setups, one can see that the use of discriminators is beneficial. As for $\text{TURBO}^{\text{paired (w } \mathcal{D})}_{\text{CNN-RESNET-CNN}}$ and UNET based setups, the CNN-RESNET-CNN based encoder-decoder architecture is better on average. Regarding the performance of the related state-of-the-art approaches, it should be pointed out that the unpaired CycleGAN demonstrates the best generalization in terms of FID $_{\mathbf{z} \to \tilde{\mathbf{x}}}$. The paired pix2pix is inferior in effectiveness to the proposed $\text{TURBO}^{\text{paired (w } \mathcal{D})}_{\text{CNN-RESNET-CNN}}$ that demonstrates the best performance.

2) Mobile phones: In contrast to the scanner data, for the mobile data, the enrolled image normalization might play an important role allowing to compensate for lighting irregularities and other important factors such as handshaking. To investigate the sensitivity of the proposed TURBO framework in different configurations to the input data irregularities we evaluate the performance of the setup under investigation on the non-normalized and normalized data[5]. From the obtained results given in Tables Table II and Table III one can see that, on average, the proposed TURBO framework, as well as the considered state-of-the-art pix2pix and CycleGAN methods, are quite robust to the enrollment imperfections. We can observe small improvements on the normalized data but, generally speaking, the results on normalized and non-normalized data are very close. Moreover, similarly to the scanner data, for both mobile phones the $\text{TURBO}^{\text{paired (w } \mathcal{D})}_{\text{CNN-RESNET-CNN}}$ outperforms $\text{TURBO}^{\text{unpaired}}_{\text{CNN-RESNET-CNN}}$ setup. The Shallow $\text{TURBO}^{\text{paired (w/o } \mathcal{D})}_{\text{CNN-RESNET-CNN}}$ shows the worst FID $_{\mathbf{x} \to \tilde{\mathbf{z}}}$ among all tested setups. The CNN-RESNET-CNN based encoder-decoder architecture shows better generalization in terms of FID metrics. While the UNEN-based TURBO is better in terms of Hamming distance, MSE and SSIM. The classical UNET is better than Eff-UNET. Similarly to the scanner data, the proposed TURBO outperforms the state-of-the-art pix2pix and CycleGAN.

It should be noted that on Samsung data all the methods show several times worse FID $_{\mathbf{z} \to \tilde{\mathbf{x}}}$ generalization than on the iPhone data. At the same time, MSE between the true printed and synthesized codes are an order of magnitude less than on scanner data. SSIM index is also higher for the mobile data which is related to the less detailed enrolled data.

E. Visualization of synthetic samples

Examples of the synthetic twins generated by the $\text{TURBO}^{\text{paired (w } \mathcal{D})}_{\text{CNN-RESNET-CNN}}$ model in the direct and reverse paths are shown in Table IV and Table V, respectively. It should be noted that the synthetic digital templates $\tilde{\mathbf{z}}$ generated in the direct path are not fully binary and contain a certain amount of grey pixels. That is expectable since the deep networks are not capable to produce pure binary output. Thus, post-binarization,

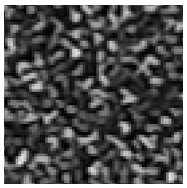
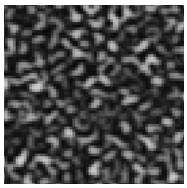
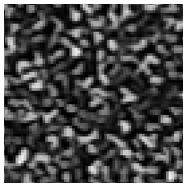
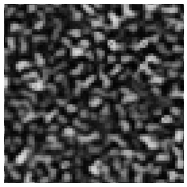
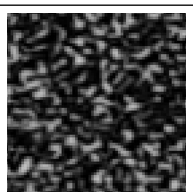
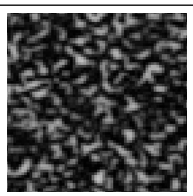
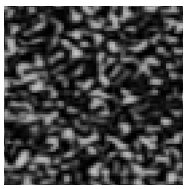
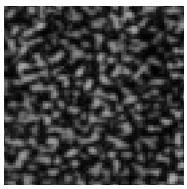
[5]As a normalization technique we used histogram matching to the selected reference CDP with the most uniform histogram.

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.2024.3361798

TABLE III
MODELS PERFORMANCE ON THE SAMSUNG DATA

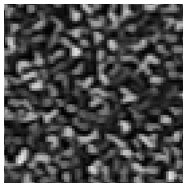
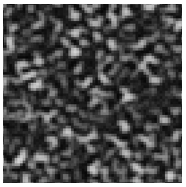
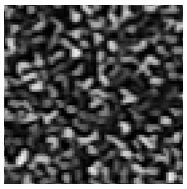
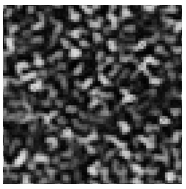
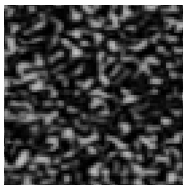
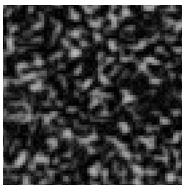
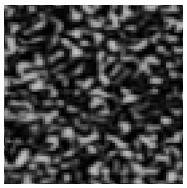
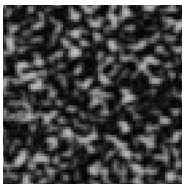
	Model	$\text{FID}_{\mathbf{x} \to \tilde{\mathbf{z}}} \downarrow$	Hamming distance ↓	$\text{FID}_{\mathbf{z} \to \tilde{\mathbf{x}}} \downarrow$	MSE ↓	SSIM↑
Non-normalized	*W/O processing*	*376.58*	*0.32*	*376.58*	*0.295*	*0.17*
	pix2pix	9.53	0.24	21.53	0.004	0.90
	CycleGAN	13.68	0.28	22.00	0.010	0.77
	$\text{TURBO}^{\text{unpaired}}_{\text{CNN-RESNET-CNN}}$	16.10	0.28	26.04	0.015	0.66
	$\text{TURBO}^{\text{paired (w } \mathcal{D})}_{\text{UNET}}$	9.65	0.24	18.49	0.004	0.90
	$\text{TURBO}^{\text{paired (w } \mathcal{D})}_{\text{CNN-RESNET-CNN}}$	**8.97**	0.25	**17.43**	0.004	0.90
	Shallow $\text{TURBO}^{\text{paired (w/o } \mathcal{D})}_{\text{CNN-RESNET-CNN}}$	117.98	0.25	135.52	0.004	0.89
	Deep $\text{TURBO}^{\text{paired (w/o } \mathcal{D})}_{\text{CNN-RESNET-CNN}}$	14.69	0.23	41.39	0.004	0.91
	$\text{TURBO}^{\text{paired}}_{\text{UNET}}$	54.12	**0.21**	32.74	0.004	0.91
	$\text{TURBO}^{\text{paired}}_{\text{Eff-UNET-2}}$	67.21	0.22	32.80	0.004	0.91
	$\text{TURBO}^{\text{paired}}_{\text{Eff-UNET-7}}$	61.12	0.21	33.59	**0.004**	**0.91**
Normalized	*W/O processing*	*381.44*	*0.31*	*381.44*	*0.278*	*0.19*
	pix2pix	8.53	0.24	20.18	0.004	0.91
	CycleGAN	8.85	0.28	22.85	0.015	0.69
	$\text{TURBO}^{\text{unpaired}}_{\text{CNN-RESNET-CNN}}$	11.71	0.29	22.89	0.015	0.70
	$\text{TURBO}^{\text{paired}}_{\text{UNET}}$	8.56	0.24	19.51	0.004	0.91
	$\text{TURBO}^{\text{paired (w } \mathcal{D})}_{\text{CNN-RESNET-CNN}}$	**7.01**	0.25	**17.35**	0.004	0.91
	Shallow $\text{TURBO}^{\text{paired (w/o } \mathcal{D})}_{\text{CNN-RESNET-CNN}}$	127.02	0.25	135.95	0.004	0.90
	Deep $\text{TURBO}^{\text{paired (w/o } \mathcal{D})}_{\text{CNN-RESNET-CNN}}$	14.27	0.23	37.37	0.003	0.92
	$\text{TURBO}^{\text{paired}}_{\text{UNET}}$	54.80	**0.21**	28.88	**0.003**	**0.92**
	$\text{TURBO}^{\text{paired}}_{\text{Eff-UNET-2}}$	63.25	0.21	42.70	0.003	0.92
	$\text{TURBO}^{\text{paired}}_{\text{Eff-UNET-7}}$	70.08	0.21	42.33	0.003	0.92

TABLE IV
EXAMPLES OF SYNTHETIC SAMPLES GENERATED BY $\text{TURBO}^{\text{PAIRED (W } \mathcal{D})}_{\text{CNN-RESNET-CNN}}$ IN THE DIRECT PATH

Device	Hist. matching	Original CDP $\mathbf{x}$	Original digital $\mathbf{z}$	Synthetic digital $\tilde{\mathbf{z}}$	Reconstructed CDP $\hat{\mathbf{x}}$
iPhone	✗				
	✓				
Samsung	✗				
	✓				

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.2024.3361798

TABLE V
EXAMPLES OF SYNTHETIC SAMPLES GENERATED BY $\text{TURBO}^{\text{PAIRED (W } \mathcal{D})}_{\text{CNN-RESNET-CNN}}$ IN THE REVERSE PATH

Device	Hist. matching	Original digital $\mathbf{z}$	Original CDP $\mathbf{x}$	Synthetic CDP $\tilde{\mathbf{x}}$	Reconstructed digital $\hat{\mathbf{z}}$
iPhone	✗				
	✓				
Samsung	✗				
	✓				

such as Otsu thresholding, could be applied if necessary. The synthetic printed CDP $\tilde{\mathbf{x}}$ generated in the reverse path visually is almost identical to the original printed CDP $\mathbf{x}$ for both phones. At the same time, the reconstructed digital $\hat{\mathbf{z}}$ contains more artefacts than the original digital $\mathbf{z}$. However, in the reverse path, it is not critical, since the reconstruction plays a role of an additional regularization on the whole process similar to the reconstruction of $\hat{\mathbf{x}}$ in the direct path. To conclude, the synthesised twins are of high quality for both used mobile phones and there is no big model's sensitivity to the data normalization, i.e., histogram matching.

VI. CONCLUSIONS

The current work is dedicated to modelling the complex physical printing-imaging process using a machine learning based model known as a "digital twin" for anti-counterfeiting applications based on CDP. The proposed model opens new possibilities for the investigation of unexplored adversarial attacks in the physical world, facilitates the data augmentation for the supervised, self-supervised and even unsupervised, i.e., one class classification, learning methods.

Along with the detailed empirical investigation of the proposed TURBO framework on the data enrolled by the high quality scanner and two modern mobile phones, we provide the complete proofs of information-theoretic formulations of loss functions used in TURBO framework. In addition, we also show a link to Information Bottleneck and to the state-of-the-art translation methods like pix2pix and CycleGAN.

The obtained results show the high robustness of the proposed TURBO framework to the input data imperfections caused by the mobile phones acquisition and the synthesized codes are of high quality and visually indistinguishable from the corresponding real printed codes.

For future work, we will consider the usage of the generated examples to build a classifier based on the augmented synthetic samples of both original CDP and fakes. Additionally, issues of stochasticity and usage in hybrid settings, where only part of the data is paired, remain open for future research.

REFERENCES

[1] J. Picard, "Digital authentication with copy-detection patterns," in *Optical Security and Counterfeit Deterrence Techniques V*, vol. 5310. SPIE, 2004, pp. 176–183.

[2] 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.

[3] 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.

[4] C. Baras and F. Cayre, "Towards a realistic channel model for security analysis of authentication using graphical codes," in *2013 IEEE International Workshop on Information Forensics and Security (WIFS)*. IEEE, 2013, pp. 115–119.

[5] A. T. P. Ho, B. A. H. Mai, W. Sawaya, and P. Bas, "Document authentication using graphical codes: impacts of the channel model," in *ACM Workshop on Information Hiding and Multimedia Security*, 2013, pp. ACM–978.

[6] 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.

[7] 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.

[8] O. Taran, J. Tutt, T. Holotyak, R. Chaban, S. Bonev, and S. Voloshynovskiy, "Mobile authentication of copy detection patterns: how critical is to know fakes?" in *IEEE International Workshop on Information Forensics and Security (WIFS)*, Montpellier, France, December 2021.

[9] 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.

[10] 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 *2021 IEEE International Workshop on Information Forensics and Security (WIFS)*. IEEE, 2021, pp. 1–6.

[11] R. Chaban, O. Taran, J. Tutt, Y. Belousov, B. Pulfer, T. Holotyak, and S. Voloshynovskiy, "Printing variability of copy detection patterns," in *IEEE International Workshop on Information Forensics and Security (WIFS)*, Shanghai, China, December 2022.

[12] B. Pulfer, R. Chaban, Y. Belousov, J. Tutt, O. Taran, T. Holotyak, and S. Voloshynovskiy, "Authentication of copy detection patterns under machine learning attacks: A supervised approach," in *IEEE International Conference on Image Processing (ICIP)*, Bordeaux, France, October 2022.

[13] 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.

[14] O. Taran, S. Bonev, T. Holotyak, and S. Voloshynovskiy, "Adversarial detection of counterfeited printable graphical codes: Towards" adversarial games" in physical world," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 2812–2816.

[15] I. Tkachenko, A. Tremeau, and T. Fournel, "Protecting documents using printed anticopy elements," 2022.

[16] I. Tkachenko, W. Puech, C. Destruel, O. Strauss, J.-M. Gaudin, and C. Guichard, "Two-level qr code for private message sharing and document authentication," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 3, pp. 571–583, 2015.

[17] A. Ferreira and M. Barni, "Attacking and defending printer source attribution classifiers in the physical domain," in *Pattern Recognition, Computer Vision, and Image Processing. ICPR 2022 International Workshops and Challenges*, J.-J. Rousseau and B. Kapralos, Eds. Cham: Springer Nature Switzerland, 2023, pp. 347–363.

[18] Q. T. Nguyen, Y. Delignon, F. Septier, and A. T. Phan-Ho, "Probabilistic modelling of printed dots at the microscopic scale," *Signal Processing: Image Communication*, vol. 62, pp. 129–138, 2018.

[19] 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.

[20] P. Roberts, "A mathematical model for inkjet printer characterisation," in *Color and Imaging Conference*, vol. 2000, no. 1. Society for Imaging Science and Technology, 2000, pp. 134–139.

[21] L. Azzari and A. Foi, "Gaussian-cauchy mixture modeling for robust signal-dependent noise estimation," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2014, pp. 5357–5361.

[22] A. Foi, M. Trimeche, V. Katkovnik, and K. Egiazarian, "Practical poissonian-gaussian noise modeling and fitting for single-image raw-data," *IEEE Transactions on Image Processing*, vol. 17, no. 10, pp. 1737–1754, 2008.

[23] C.-W. Wong and M. Wu, "Counterfeit detection using paper puf and mobile cameras," in *2015 IEEE International Workshop on Information Forensics and Security (WIFS)*. IEEE, 2015, pp. 1–6.

[24] 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.

[25] G. Quétant, M. Drozdova, V. Kinakh, T. Golling, and S. Voloshynovskiy, "Turbo-sim: a generalised generative model with a physical latent space," 2021.

[26] S. Voloshynovskiy, O. Taran, T. Joakim, and T. Holotyak, "Object authentication using digital blueprints and physical fingerprints," Dec. 15 2022, uS Patent App. 17/835,674.

[27] D. Kingma and M. Welling, "Auto-encoding variational bayes," *stat*, vol. 1050, p. 1, 2014.

[28] D. J. Rezende, S. Mohamed, and D. Wierstra, "Stochastic backpropagation and approximate inference in deep generative models," in *International conference on machine learning*. PMLR, 2014, pp. 1278–1286.

[29] A. Makhzani, J. Shlens, N. Jaitly, I. Goodfellow, and B. Frey, "Adversarial autoencoders," *arXiv preprint arXiv:1511.05644*, 2015.

[30] N. Tishby, F. C. Pereira, and W. Bialek, "The information bottleneck method," *arXiv preprint physics/0004057*, 2000.

[31] A. A. Alemi, I. Fischer, J. V. Dillon, and K. Murphy, "Deep variational information bottleneck," *arXiv preprint arXiv:1612.00410*, 2016.

[32] S. Voloshynovskiy, M. Kondah, S. Rezaeifar, O. Taran, T. Hotolyak, and D. J. Rezende, "Information bottleneck through variational glasses," in *NeurIPS Workshop on Bayesian Deep Learning*, Vancouver, Canada, December 2019.

[33] I. Higgins, L. Matthey, A. Pal, C. Burgess, X. Glorot, M. Botvinick, S. Mohamed, and A. Lerchner, "beta-vae: Learning basic visual concepts with a constrained variational framework," in *International conference on learning representations*, 2017.

[34] S. Zhao, J. Song, and S. Ermon, "Infovae: Information maximizing variational autoencoders," *arXiv preprint arXiv:1706.02262*, 2017.

[35] S. Voloshynovskiy, O. Taran, M. Kondah, T. Holotyak, and D. Rezende, "Variational information bottleneck for semi-supervised classification," in *Entropy Journal special issue "Information Bottleneck: Theory and Applications in Deep Learning"*, August 2020.

[36] J. T. Springenberg, "Unsupervised and semi-supervised learning with categorical generative adversarial networks," *arXiv preprint arXiv:1511.06390*, 2015.

[37] D. P. Kingma, D. J. Rezende, S. Mohamed, and M. Welling, "Semi-supervised learning with deep generative models," in *Proceedings of the 27th International Conference on Neural Information Processing Systems-Volume 2*, vol. 27, 2014, pp. 3581–3589.

[38] M. Śmieja, M. Wołczyk, J. Tabor, and B. C. Geiger, "Segma: Semi-supervised gaussian mixture autoencoder," *IEEE transactions on neural networks and learning systems*, vol. 32, no. 9, pp. 3930–3941, 2020.

[39] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," *CVPR*, 2017.

[40] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *IEEE International Conference on Computer Vision (ICCV)*, 2017.

[41] T. Park, A. A. Efros, R. Zhang, and J.-Y. Zhu, "Contrastive learning for unpaired image-to-image translation," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part IX 16*. Springer, 2020, pp. 319–345.

[42] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang *et al.*, "Photo-realistic single image super-resolution using a generative adversarial network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4681–4690.

[43] T. Park, A. A. Efros, R. Zhang, and J.-Y. Zhu, "Contrastive learning for unpaired image-to-image translation," in *Proceedings of (ECCV) European Conference on Computer Vision*, August 2020, pp. 319–345.

[44] A. v. d. Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," *arXiv preprint arXiv:1807.03748*, 2018.

[45] Y. Choi, M. Choi, M. Kim, J.-W. Ha, S. Kim, and J. Choo, "Stargan: Unified generative adversarial networks for multi-domain image-to-image translation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[46] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[47] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.

[48] B. Baheti, S. Innani, S. Gajre, and S. Talbar, "Eff-unet: A novel architecture for semantic segmentation in unstructured environment," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2020, pp. 358–359.

[49] M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *International conference on machine learning*. PMLR, 2019, pp. 6105–6114.

[50] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Instance normalization: The missing ingredient for fast stylization," *arXiv preprint arXiv:1607.08022*, 2016.

[51] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *International Conference on Learning Representations (ICLR)*, 2015.

[52] M. Lucic, K. Kurach, M. Michalski, S. Gelly, and O. Bousquet, "Are gans created equal? a large-scale study," *Advances in neural information processing systems*, vol. 31, 2018.

[53] X. Mao, Q. Li, H. Xie, R. Y. Lau, Z. Wang, and S. Paul Smolley, "Least squares generative adversarial networks," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[54] J. H. Lim and J. C. Ye, "Geometric gan," *arXiv preprint arXiv:1705.02894*, 2017.

[55] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein generative adversarial networks," in *International conference on machine learning*. PMLR, 2017, pp. 214–223.

[56] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville, "Improved training of wasserstein gans," *Advances in neural information processing systems*, vol. 30, 2017.

[57] A. Shrivastava, T. Pfister, O. Tuzel, J. Susskind, W. Wang, and R. Webb, "Learning from simulated and unsupervised images through adversarial training," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2107–2116.

[58] S. Chintala, "How to train a gan? tips and tricks to make gans work," 2016, starter from "How to Train a GAN?" at NIPS2016.

[59] C. K. Sønderby, J. Caballero, L. Theis, W. Shi, and F. Huszár, "Amortised map inference for image super-resolution," *arXiv preprint arXiv:1610.04490*, 2016.

[60] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE transactions on image processing*, vol. 13, no. 4, pp. 600–612, 2004.

[61] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "Gans trained by a two time-scale update rule converge to a local nash equilibrium," in *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., vol. 30. Curran Associates, Inc., 2017.

[62] O. Yadan, "Hydra - a framework for elegantly configuring complex applications," Github, 2019.

[63] L. Biewald, "Experiment tracking with weights and biases," 2020, software available from wandb.com.

APPENDIX A
INFORMATION THEORETIC FORMULATION OF THE DIRECT PATH LOSS FUNCTION

The encoder loss of the direct path is defined as:

$$
\begin{aligned}
I(\mathbf{X};\mathbf{Z}) &= \mathbb{E}_{p(\mathbf{x},\mathbf{z})}\left[\log \frac{p(\mathbf{x},\mathbf{z})}{p_{\mathbf{x}}(\mathbf{x})p_{\mathbf{z}}(\mathbf{z})}\right] \\
&= \mathbb{E}_{p(\mathbf{x},\mathbf{z})}\left[\log \frac{p(\mathbf{z}|\mathbf{x})}{p_{\mathbf{z}}(\mathbf{z})}\frac{q_\phi(\mathbf{z}|\mathbf{x})}{q_\phi(\mathbf{z}|\mathbf{x})}\right] \\
&= \mathbb{E}_{p(\mathbf{x},\mathbf{z})}\left[\log \frac{q_\phi(\mathbf{z}|\mathbf{x})}{p_{\mathbf{z}}(\mathbf{z})}\right] + D_{\mathrm{KL}}\left(p(\mathbf{z}|\mathbf{x})\|q_\phi(\mathbf{z}|\mathbf{x})\right) \\
&\geq \mathbb{E}_{p(\mathbf{x},\mathbf{z})}\left[\log \frac{q_\phi(\mathbf{z}|\mathbf{x})}{p_{\mathbf{z}}(\mathbf{z})}\frac{\tilde{q}_\phi(\mathbf{z})}{\tilde{q}_\phi(\mathbf{z})}\right] \\
&= \mathbb{E}_{p(\mathbf{x},\mathbf{z})}\left[\log q_\phi(\mathbf{z}|\mathbf{x})\right] - D_{\mathrm{KL}}\left(p_{\mathbf{z}}(\mathbf{z})\|\tilde{q}_\phi(\mathbf{z})\right) \\
&\quad + H\left(p_{\mathbf{z}}(\mathbf{z});\tilde{q}_\phi(\mathbf{z})\right) \\
&\geq \underbrace{\mathbb{E}_{p(\mathbf{x},\mathbf{z})}\left[\log q_\phi(\mathbf{z}|\mathbf{x})\right]}_{-\mathcal{L}_{\tilde{z}}(\mathbf{z},\tilde{\mathbf{z}})} - \underbrace{D_{\mathrm{KL}}\left(p_{\mathbf{z}}(\mathbf{z})\|\tilde{q}_\phi(\mathbf{z})\right)}_{\mathcal{D}_{\tilde{z}}(\mathbf{z},\tilde{\mathbf{z}})} \\
&=: \mathcal{I}^{\mathrm{z}}_\phi(\mathbf{X};\mathbf{Z}),
\end{aligned} \tag{14}
$$

where the marginal distribution on the output of the encoder of the direct path is defined as $\tilde{q}_\phi(\mathbf{z}) = \mathbb{E}_{p_{\mathbf{x}}(\mathbf{x})}\left[q_\phi(\mathbf{z}|\mathbf{x})\right]$. The first inequality holds because the KL-divergence $D_{\mathrm{KL}}\left(p(\mathbf{z}|\mathbf{x})\|q_\phi(\mathbf{z}|\mathbf{x})\right) \geq 0$ and the second holds because the cross-entropy $H\left(p_{\mathbf{z}}(\mathbf{z});\tilde{q}_\phi(\mathbf{z})\right) \geq 0$.

The decoder loss of the direct path is defined as:

$$
\begin{aligned}
I_\phi(\mathbf{X};\tilde{\mathbf{Z}}) &= \mathbb{E}_{q_\phi(\mathbf{x},\mathbf{z})}\left[\log \frac{q_\phi(\mathbf{x},\mathbf{z})}{p_{\mathbf{x}}(\mathbf{x})\tilde{q}_\phi(\mathbf{z})}\right] \\
&= \mathbb{E}_{q_\phi(\mathbf{x},\mathbf{z})}\left[\log \frac{q_\phi(\mathbf{x}|\mathbf{z})}{p_{\mathbf{x}}(\mathbf{x})}\frac{p_\theta(\mathbf{x}|\mathbf{z})}{p_\theta(\mathbf{x}|\mathbf{z})}\right] \\
&= \mathbb{E}_{q_\phi(\mathbf{x},\mathbf{z})}\left[\log \frac{p_\theta(\mathbf{x}|\mathbf{z})}{p_{\mathbf{x}}(\mathbf{x})}\right] + D_{\mathrm{KL}}\left(q_\phi(\mathbf{x}|\mathbf{z})\|p_\theta(\mathbf{x}|\mathbf{z})\right) \\
&\geq \mathbb{E}_{q_\phi(\mathbf{x},\mathbf{z})}\left[\log \frac{p_\theta(\mathbf{x}|\mathbf{z})}{p_{\mathbf{x}}(\mathbf{x})}\frac{\hat{p}_\theta(\mathbf{x})}{\hat{p}_\theta(\mathbf{x})}\right] \\
&= \mathbb{E}_{q_\phi(\mathbf{x},\mathbf{z})}\left[\log p_\theta(\mathbf{x}|\mathbf{z})\right] - D_{\mathrm{KL}}\left(p_{\mathbf{x}}(\mathbf{x})\|\hat{p}_\theta(\mathbf{x})\right) \\
&\quad + H\left(p_{\mathbf{x}}(\mathbf{x});\hat{p}_\theta(\mathbf{x})\right) \\
&\geq \underbrace{\mathbb{E}_{q_\phi(\mathbf{x},\mathbf{z})}\left[\log p_\theta(\mathbf{x}|\mathbf{z})\right]}_{-\mathcal{L}_{\hat{x}}(\mathbf{x},\hat{\mathbf{x}})} - \underbrace{D_{\mathrm{KL}}\left(p_{\mathbf{x}}(\mathbf{x})\|\hat{p}_\theta(\mathbf{x})\right)}_{\mathcal{D}_{\hat{x}}(\mathbf{x},\hat{\mathbf{x}})} \\
&=: \mathcal{I}^{\mathrm{x}}_{\phi,\theta}(\mathbf{X};\tilde{\mathbf{Z}}),
\end{aligned} \tag{15}
$$

where $q_\phi(\mathbf{x},\mathbf{z}) = p_{\mathbf{x}}(\mathbf{x})q_\phi(\mathbf{z}|\mathbf{x})$ and where the marginal distribution on the output of the decoder of the direct path is defined as $\hat{p}_\theta(\mathbf{x}) = \mathbb{E}_{p_{\mathbf{x}}(\mathbf{x})}\left[\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\left[p_\theta(\mathbf{x}|\mathbf{z})\right]\right]$. The first inequality holds because the KL-divergence $D_{\mathrm{KL}}\left(q_\phi(\mathbf{x}|\mathbf{z})\|p_\theta(\mathbf{x}|\mathbf{z})\right) \geq 0$ and the second holds because the cross-entropy $H\left(p_{\mathbf{x}}(\mathbf{x});\hat{p}_\theta(\mathbf{x})\right) \geq 0$.

Finally, the total direct path loss is defined as:

$$
\mathcal{L}^{\mathrm{Direct}}(\phi,\theta) = -\mathcal{I}^{\mathrm{z}}_\phi(\mathbf{X};\mathbf{Z}) - \lambda_D \mathcal{I}^{\mathrm{x}}_{\phi,\theta}(\mathbf{X};\tilde{\mathbf{Z}}), \tag{16}
$$

where λ_D is the trade-off parameter.

APPENDIX B
INFORMATION THEORETIC FORMULATION OF THE REVERSE PATH LOSS FUNCTION

The decoder loss of the reverse path is defined as:

$$
\begin{aligned}
I(\mathbf{X};\mathbf{Z}) &= \mathbb{E}_{p(\mathbf{x},\mathbf{z})}\left[\log \frac{p(\mathbf{x},\mathbf{z})}{p_{\mathbf{x}}(\mathbf{x})p_{\mathbf{z}}(\mathbf{z})}\right] \\
&= \mathbb{E}_{p(\mathbf{x},\mathbf{z})}\left[\log \frac{p(\mathbf{x}|\mathbf{z})}{p_{\mathbf{x}}(\mathbf{x})}\frac{p_\theta(\mathbf{x}|\mathbf{z})}{p_\theta(\mathbf{x}|\mathbf{z})}\right] \\
&= \mathbb{E}_{p(\mathbf{x},\mathbf{z})}\left[\log \frac{p_\theta(\mathbf{x}|\mathbf{z})}{p_{\mathbf{x}}(\mathbf{x})}\right] + D_{\mathrm{KL}}\left(p(\mathbf{x}|\mathbf{z})\|p_\theta(\mathbf{x}|\mathbf{z})\right) \\
&\geq \mathbb{E}_{p(\mathbf{x},\mathbf{z})}\left[\log \frac{p_\theta(\mathbf{x}|\mathbf{z})}{p_{\mathbf{x}}(\mathbf{x})}\frac{\tilde{p}_\theta(\mathbf{x})}{\tilde{p}_\theta(\mathbf{x})}\right] \\
&= \mathbb{E}_{p(\mathbf{x},\mathbf{z})}\left[\log p_\theta(\mathbf{x}|\mathbf{z})\right] - D_{\mathrm{KL}}\left(p_{\mathbf{x}}(\mathbf{x})\|\tilde{p}_\theta(\mathbf{x})\right) \\
&\quad + H\left(p_{\mathbf{x}}(\mathbf{x});\tilde{p}_\theta(\mathbf{x})\right) \\
&\geq \underbrace{\mathbb{E}_{p(\mathbf{x},\mathbf{z})}\left[\log p_\theta(\mathbf{x}|\mathbf{z})\right]}_{-\mathcal{L}_{\tilde{x}}(\mathbf{x},\tilde{\mathbf{x}})} - \underbrace{D_{\mathrm{KL}}\left(p_{\mathbf{x}}(\mathbf{x})\|\tilde{p}_\theta(\mathbf{x})\right)}_{\mathcal{D}_{\tilde{x}}(\mathbf{x},\tilde{\mathbf{x}})} \\
&=: \mathcal{I}^{\mathrm{x}}_\theta(\mathbf{X};\mathbf{Z}),
\end{aligned} \tag{17}
$$

where the marginal distribution on the output of the decoder of the reverse path is defined as $\tilde{p}_\theta(\mathbf{x}) = \mathbb{E}_{p_{\mathbf{z}}(\mathbf{z})}\left[p_\theta(\mathbf{x}|\mathbf{z})\right]$. The first inequality holds because the KL-divergence $D_{\mathrm{KL}}\left(p(\mathbf{x}|\mathbf{z})\|p_\theta(\mathbf{x}|\mathbf{z})\right) \geq 0$ and the second holds because the cross-entropy $H\left(p_{\mathbf{x}}(\mathbf{x});\tilde{p}_\theta(\mathbf{x})\right) \geq 0$.

TABLE VI
MODELS STANDARD DEVIATIONS ON THE NON-NORMALIZED IPHONE DATASET

Model	FID $_{\mathbf{x}\to\tilde{\mathbf{z}}}$ ↓	Hamming distance ↓	FID $_{\mathbf{z}\to\tilde{\mathbf{x}}}$ ↓	MSE ↓	SSIM↑
pix2pix	14.04 ± 0.43	0.24 ± 0.01	13.5 ± 0.74	0.006 ± 0.0002	0.89 ± 0.01
TURBO$^{\text{paired}}_{\text{UNET}}$	36.39 ± 1.15	0.21 ± 0.01	17.07 ± 0.19	0.005 ± 0.0001	0.92 ± 0.01

The encoder loss of the reverse path is defined as:

$$
\begin{aligned}
I_\theta(\tilde{\mathbf{X}};\mathbf{Z}) &= \mathbb{E}_{p_\theta(\mathbf{x},\mathbf{z})}\left[\log\frac{p_\theta(\mathbf{x},\mathbf{z})}{\tilde{p}_\theta(\mathbf{x})p_{\mathbf{z}}(\mathbf{z})}\right] \\
&= \mathbb{E}_{p_\theta(\mathbf{x},\mathbf{z})}\left[\log\frac{p_\theta(\mathbf{z}|\mathbf{x})}{p_{\mathbf{z}}(\mathbf{z})}\frac{q_\phi(\mathbf{z}|\mathbf{x})}{q_\phi(\mathbf{z}|\mathbf{x})}\right] \\
&= \mathbb{E}_{p_\theta(\mathbf{x},\mathbf{z})}\left[\log\frac{q_\phi(\mathbf{z}|\mathbf{x})}{p_{\mathbf{z}}(\mathbf{z})}\right] + D_{\mathrm{KL}}\left(p_\theta(\mathbf{z}|\mathbf{x})\|q_\phi(\mathbf{z}|\mathbf{x})\right) \\
&\geq \mathbb{E}_{p_\theta(\mathbf{x},\mathbf{z})}\left[\log\frac{q_\phi(\mathbf{z}|\mathbf{x})}{p_{\mathbf{z}}(\mathbf{z})}\frac{\hat{q}_\phi(\mathbf{z})}{\hat{q}_\phi(\mathbf{z})}\right] \\
&= \mathbb{E}_{p_\theta(\mathbf{x},\mathbf{z})}\left[\log q_\phi(\mathbf{z}|\mathbf{x})\right] - D_{\mathrm{KL}}\left(p_{\mathbf{z}}(\mathbf{z})\|\hat{q}_\phi(\mathbf{z})\right) \\
&\quad + H\left(p_{\mathbf{z}}(\mathbf{z});\hat{q}_\phi(\mathbf{z})\right) \\
&\geq \underbrace{\mathbb{E}_{p_\theta(\mathbf{x},\mathbf{z})}\left[\log q_\phi(\mathbf{z}|\mathbf{x})\right]}_{-\mathcal{L}_{\hat{z}}(\mathbf{z},\hat{\mathbf{z}})} - \underbrace{D_{\mathrm{KL}}\left(p_{\mathbf{z}}(\mathbf{z})\|\hat{q}_\phi(\mathbf{z})\right)}_{\mathcal{D}_{\hat{z}}(\mathbf{z},\hat{\mathbf{z}})} \\
&=: \mathcal{I}^z_{\phi,\theta}(\tilde{\mathbf{X}};\mathbf{Z}),
\end{aligned}
\tag{18}
$$

where $p_\theta(\mathbf{x},\mathbf{z}) = p_{\mathbf{z}}(\mathbf{z})p_\theta(\mathbf{x}|\mathbf{z})$ and where the marginal distribution on the output of the encoder of the reverse path is defined as $\hat{q}_\phi(\mathbf{z}) = \mathbb{E}_{p_{\mathbf{z}}(\mathbf{z})}\left[\mathbb{E}_{p_\theta(\mathbf{x}|\mathbf{z})}\left[q_\phi(\mathbf{z}|\mathbf{x})\right]\right]$. The first inequality holds because the KL-divergence $D_{\mathrm{KL}}\left(p_\theta(\mathbf{z}|\mathbf{x})\|q_\phi(\mathbf{z}|\mathbf{x})\right) \geq 0$ and the second holds because the cross-entropy $H\left(p_{\mathbf{z}}(\mathbf{z});\hat{q}_\phi(\mathbf{z})\right) \geq 0$.

Finally, the total reverse path loss is defined as:

$$
\mathcal{L}^{\text{Reverse}}(\phi,\theta) = -\mathcal{I}^x_\theta(\mathbf{X};\mathbf{Z}) - \lambda_R\mathcal{I}^z_{\phi,\theta}(\tilde{\mathbf{X}};\mathbf{Z}), \tag{19}
$$

where λ_R is the trade-off parameter.

APPENDIX C
TRAINING DETAILS

For all our experiments, we utilized PyTorch. Additionally, we used Hydra [62], a framework for structured and easy-to-maintain configuration files to manage our experiment configurations. Finally, to log our experimental results, we used wandb [63], a web-based platform that allows for the easy tracking and visualization of experimental metrics.

The length of one training cycle per model can range from four hours to four days, depending on the specific configuration used. This training process was performed using either four RTX 2080 Ti graphics cards or a single A100 80 GB card. We trained all models from scratch and did not perform fine-tuning of any pretrained models. Although the total training time for all experiments was 306 GPU days, the inference time for the Turbo model applied to 280 test images is just 15 seconds on a single A100 GPU with 80GB. This makes the proposed framework usable for real-world tasks since after training the model the time needed to run it is just 0.05 seconds per image on average.

We report typical standard deviations for selected models trained on the non-normalized iPhone dataset in Table VI.

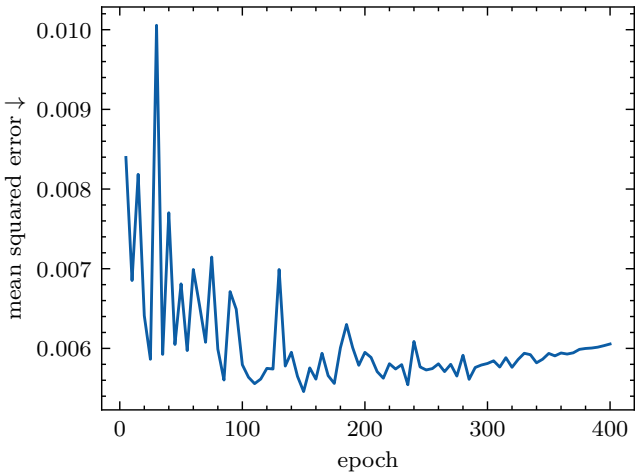


Fig. 4. Evolution of the mean-squared error during training of the TURBO$^{\text{paired}}_{\text{UNET}}$ model on the non-normalized iPhone dataset. The x-axis represents the 400 epochs during training with an evaluation frequency equal to 5. The y-axis represents the MSE error between $\mathbf{x}$ and $\tilde{\mathbf{x}}$ at the current epoch.

APPENDIX D
HYPERPARAMETERS OF MODELS' TRAINING USED IN OUR EXPERIMENTS

We list here all used hyperparameters for the experiments:

- Acquisition device: Epson Perfection V850 Pro scanner, iPhone 12 Pro, Samsung Galaxy Note 20 Ultra
- Normalization: non-normalized, normalized
- Dataset setup: paired, unpaired
- Optimizer: Adam with $\beta_1 = 0.5, \beta_2 = 0.999$
- Learning rate: 0.0002
- Number of epochs: 200 with initial learning rate, 200 to linearly decay learning rate to zero
- Batch size: 10
- Models: CUT, pix2pix, CycleGAN, TURBO
- Architecture of Generator: CNN-RESNET-CNN, UNET, Eff-UNET
- Architecture of Discriminator: PatchGAN, ImageGAN
- Adversarial loss: LSGAN, HINGE, WGAN-GP
- Thresholding functions: local, mean, minimum, Otsu, triangle
- Number of discriminator updates per generator updates n_{D}: 1, 5
- Probability of label flipping for batch p_{flip}: 0, 0.025
- Probability of adding noise to discriminator's input for batch p_{noise}: 0, 0.2
- Weight of added noise to discriminator's input for batch w_{noise}: 0.3
- Weight of pairwise estimation loss in the latent space α, ν: 1, 3, 6, 9, 12, 15, 19, 59, 109, 309.