mID: Tracing Screen Photos via Moiré Patterns

Yushi Cheng
Zhejiang University

Xiaoyu Ji*
Zhejiang University

Lixu Wang†
Zhejiang University

Qi Pang†
Zhejiang University

Yi-Chao Chen
Shanghai Jiao Tong University

Wenyuan Xu
Zhejiang University

Abstract

Cyber-theft of trade secrets has become a serious business threat. Digital watermarking is a popular technique to assist in identifying the source of the file leakage, whereby a unique watermark for each insider is hidden in sensitive files. However, malicious insiders may use their smartphones to photograph the secret file displayed on screens to remove the embedded hidden digital watermarks due to the optical noises introduced during photographing. To identify the leakage source despite such *screen-photo-based leakage attacks*, we leverage Moiré pattern, an optical phenomenon resulted from the optical interaction between electronic screens and cameras. As such, we present mID, a new watermark-like technique that can create a carefully crafted Moiré pattern on the photo when it is taken towards the screen. We design patterns that appear to be natural yet can be linked to the identity of the leaker. We implemented mID and evaluate it with 5 display devices and 6 smartphones from various manufacturers and models. The results demonstrate that mID can achieve an average bit error rate (BER) of 0.6% and can successfully identify an ID with an average accuracy of 96%, with little influence from the type of display devices, cameras, IDs, and ambient lights.

1 Introduction

Cyber-theft of trade secrets is the illegal leakage of sensitive business information, e.g., digital documents, images, or codes over cyberspace. It is estimated to cause a loss of €60 billion in economic growth and 289,000 jobs in Europe alone in 2018, and the losses are expected to be one million jobs by 2025 [26]. Such cyber-thefts are typically involved with insiders [44], whereby employees access confidential business files legally yet leak them to unauthorized parties via emails or messaging systems (e.g., WhatsApp). To identify and trace the source of the leakage, i.e., digital forensics, companies log files outbound from the network interface card or USB ports [28], and insert a digital watermark [1, 7, 12, 20, 23, 29] that is unique to an employee in each confidential file.

To avoid exposure, the adversary starts to photograph (usually with smartphones) the computer screen that displays the confidential information and leaks it out anonymously [28]. Hereafter, we name this kind of attack as **screen-photo-based leakage attack**. After such an attack, unfortunately, the digital watermark may no longer be recognizable due to the noises (e.g., the Gaussian and salt-and-pepper noises [4]) introduced by both the electronic screen and the camera sensors. Therefore, digital forensics for screen photos, i.e., photos taken towards screens, is in urgent need.

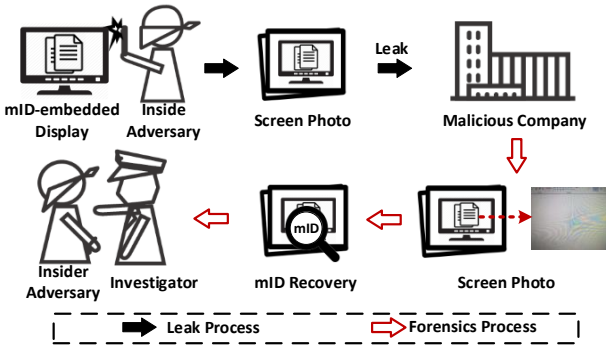


Figure 1: An illustration of mID for screen photo forensics: The identity (ID) of an adversary is embedded on the screen by subtly manipulating what is being displayed and can be recovered later by analyzing the Moiré patterns on the screen photos.

In this paper, we propose mID, a digital forensics mechanism against the aforementioned screen-photo-based leakage attack utilizing Moiré patterns [43]. Moiré patterns are optical phenomena generated during the process of photographing screens and are often observed in the photos of computer screens, TV screens, etc. Moiré patterns are ideal for screen photo forensics because they are natural optical phenomena and attract almost no, if any, attention of the adversary. As shown in Fig. 1, mID works as follows: once an adversary logs into a computer or an application (e.g., an email system) with her account, mID will modify the displayed content slightly based on her identity (ID), such that when she takes pictures of the screen, the modification will create Moiré patterns in the photos. Finally, the embedded Moiré patterns are decoded to obtain the ID.

Photo forensics via Moiré patterns is promising yet challenging, since we have to encode IDs inside the Moiré patterns reliably yet keep the patterns as if they are naturally generated. In this case, a naive method [17, 30, 43, 45] that encodes IDs by manipulating the phases of images will not work, because it may change the display content (e.g., change a straight line into a wavy one) or create artificial patterns in the generated

*Corresponding author.
†Equal contribution.

Moiré stripes. Meanwhile, mID has to adjust the encoding in real-time as users modify the window sizes, e.g., maximize the file viewer. Last but not the least, decoding IDs from Moiré patterns in photos has to overcome the distortion caused by the angle of cameras, the photo content, etc.

To overcome the aforementioned challenges, we design the encoding and decoding schemes of mID. The key to encoding is to have as little influence as possible on the original display content and to find the best display areas for encoding such that the generated Moiré patterns remain sneaky. Thus, we first employ a vertical grating scheme to imitate the natural screen-camera channel. Then, we modify the intensity levels of pixels to generate designed Moiré patterns and exploit the discretized bipolar non-return-to-zero (NRZ) encoding method. Considering that humans perceive light and color in a non-linear manner [33], we further correct the luminance difference caused by the bipolar NRZ encoding to smoothen the visual effect of the generated grating image. Furthermore, mID automatically searches for suitable display areas for information embedding, such that it maximizes its possibility of being captured in the photos. To reliably decode the ID despite image distortion, we first extract the Moiré areas with image rectification and window scanning. Then, we transform the Moiré areas into the HSV (hue, saturation, value) color space [49], and perform saturation balance and enlargement for high decoding efficiency. After that, we use k-means clustering with the assistance of check codes to recover the embedded IDs. In summary, our contribution includes below:

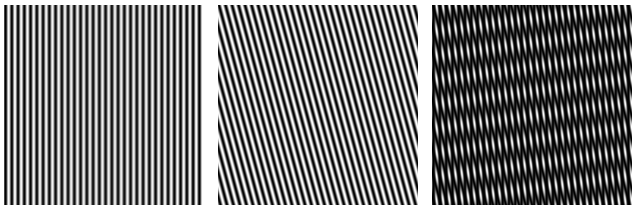
- We propose to exploit the natural Moiré phenomenon existing in the screen-camera channel for screen photo forensics. To the best of our knowledge, this is the first work that addresses screen photo forensics. We believe that mID is a promising technique and can work complementarily to several existing ones.
- We design mID, an effective digital forensics mechanism for file leakages via photos utilizing Moiré patterns.
- We evaluate mID with 5 display devices and 6 smartphones from various manufacturers and models. The results show that mID can achieve an average BER of 0.6% and an average NER (identity number error rate) of 4.0%. In addition, it can operate with little influence from display devices, cameras, IDs, and ambient lights.

2 Background

In this section, we begin with the principle and profiling of Moiré patterns. Then, we introduce the nonlinearity of the screen-camera channel that contributes to Moiré patterns in the screen photos.

2.1 Moiré Pattern

Moiré patterns or Moiré fringes are interference patterns created when opaque ruled patterns with transparent gaps are overlaid [2]. Natural Moiré phenomena can be seen by looking through the folds of a nylon curtain of small mesh, or at two sheets of graph paper twisted 20-30 degrees to each other. Moreover, a pattern on a TV screen, can interfere with the shape of light sensors when photographed by a digital camera and thus generate Moiré patterns. In this paper, we utilize such an effect for screen photo forensics.



(a) Periodical layer l_1 (b) Periodical layer l_2 (c) Superposition of l_1 and l_2

Figure 2: The superposition of periodical layers $l_1(x,y) = 0.5 + 0.5cos(y)$ (a) and $l_2(x,y) = 0.5 + 0.5cos(ycos(15°) + xsin(15°))$ (b) generates new frequency components (c).

2.2 Moiré Pattern Profiling

Moiré patterns are usually generated by the superposition of periodic layers [2] and appear as new structures that do not exist in any of the original layers. The periodic layer could be an image, a nylon curtain, an optical filter, etc. Assume l_1 and l_2 are two periodical layers and s is the generated superposition pattern, where:

$$s(x,y) = l_1(x,y) \times l_2(x,y) \tag{1}$$

The multiplication of two periodic functions results in nonlinearity in the frequency domain. As illustrated in Fig. 2, l_1 and l_2 are two cosine functions with the frequency of f_1 and f_2 respectively. Then, the generated structure s can be calculated as follows:

$$\begin{aligned} s &= l_1 \times l_2 \\ &= (a_1 + b_1 cos(2\pi f_1 t)) \times (a_2 + b_2 cos(2\pi f_2 t)) \\ &= a_1 a_2 + a_1 b_2 cos(2\pi f_2 t) + a_2 b_1 cos(2\pi f_1 t) \\ &\quad + b_1 b_2 cos(2\pi (f_1 + f_2) t) + b_1 b_2 cos(2\pi (f_1 - f_2) t) \end{aligned} \tag{2}$$

which contains two new components $(f_1 + f_2)$ and $(f_1 - f_2)$ in the frequency domain. Since human eyes are more sensitive to low frequency signals, the new component $(f_1 - f_2)$ becomes noticeable as Moiré patterns if it is lower than the cutoff frequency of human visual system (HVS) [51] and meanwhile has a significant amplitude.

2.3 Moiré Pattern of Screen-camera Channel

Digital cameras often cause Moiré phenomenon when taking pictures of digital screens, e.g., TV screens or liquid-crystal displays (LCDs). The nonlinearity arises from the interference of digital screens and the Color Filter Array (CFA) on the camera image sensors, which we call the screen-camera channel. The process is depicted in Fig. 3.

Screen Image. The unit structure of digital screens, e.g., LCD screens, usually consists of tri-color (red (R), green (G) and blue (B)) filters and emits corresponding light separately,

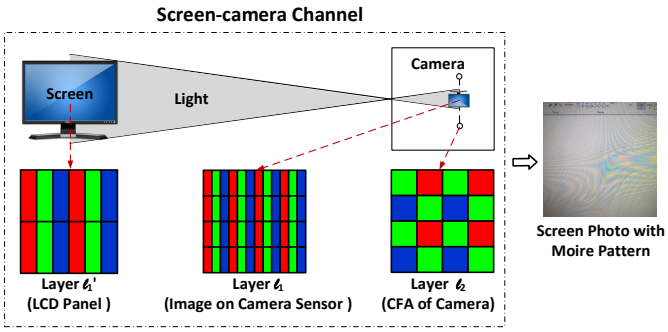


Figure 3: An illustration of the imaging process of the screen (LCD)-camera (CFA) channel and the resulted screen photo with Moiré patterns.

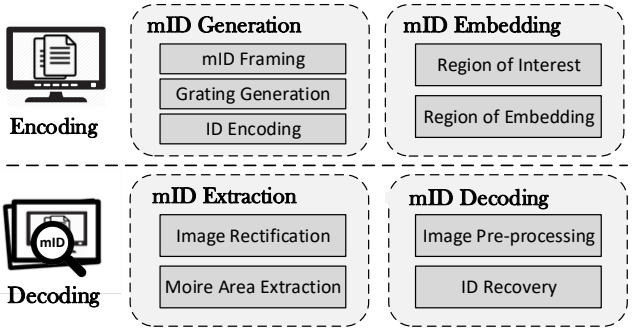


Figure 4: System overview of mID scheme.

e.g., LCD panel shown on the left of Fig. 3. When taking a picture towards an LCD screen, the unit structures of the LCD panel are projected onto the camera sensors and form a layer of spatial patterns, i.e., the image of the LCD screen. We denote the image of the LCD screen on the camera sensors as layer l_1 with a frequency of f_1, which interacts with the CFA directly to generate Moiré patterns. To distinguish, we denote the layer formed by the original LCD screen as layer $l_1^{'}$ with a frequency of $f_1^{'}$. Note that other displays such as LED screens are also applicable.

CFA. In the screen-camera channel, the light emitted by the screen is received by the camera image sensors. A CFA (i.e., a mosaic of tiny color filters) is placed over the camera image sensor to capture the color information. Bayer filter is the most common filter on smartphones' built-in cameras [46], which gives information about the intensity of light in RGB wavelength regions in a 2×2 array (e.g., CFA of camera shown in the middle of Fig. 3). As a result, the CFA forms another layer of spatial patterns, which we denote as layer l_2 with a frequency of f_2.

Nonlinear Optical Interaction. According to the Moiré pattern profiling, the superposition of l_1 (image of the screen) and l_2 (CFA of the camera) can generate new components in the frequency domain. When the camera is positioned at a proper distance and angle, the generated component $(f_1 - f_2)$ falls in the observable frequency range and appears as ripple patterns on the captured screen photo, i.e., the Moiré patterns caused by the screen-camera channel (shown in Fig. 3).

Inspired by the natural Moiré phenomenon existing in the screen-camera channel, we propose to exploit the nonlinear optical interaction between the CFA of the camera and the well-designed camouflaging periodical patterns displayed on the screen, to embed Moiré-pattern-based ID, i.e., mID, into the screen photo, to trace the source of file leakages.

3 Threat Model

For the screen-photo-based leakage attacks, the adversary's goal is to leak confidential information via the photo taken by smartphones. The photo can be delivered to unauthorized parties from networking applications such as WhatsApp or a portable disk. In this attack scenario, we assume the company who wants to trace the screen photo, i.e., file forensics, has full control over the confidential file. In other words, they can modify the hardware and software such as screen configuration. For the adversary, we have the following assumptions:

- **Screen-capturing with Smartphones.** To avoid being logged and caught, the attacker tends to take a picture of the screen displaying the confidential information with her smartphones. The adversary wishes to capture the confidential content completely and clearly, and therefore they should place their smartphones close enough at a good angle.
- **Untraceability over Internet.** The adversary is able to leak the screen photos anonymously via open networks, e.g., public Wi-Fi. As a result, the path of the photo leakage cannot be traced by the company.
- **Photo Processing.** To reduce the risk of being traced, we assume the adversary may process the captured screen photos. The possible processing operations include photo duplication, photo compression, image up/downscaling, format conversion, image cut, etc.

4 Design

4.1 Design Requirement

To trace the source of file leakages via Moiré patterns, mID shall satisfy the following requirements.

Subtle Visual Difference to User. The embedded mID shall have no obvious visual impact to users for the sake of user experience. In other words, users should not be able to recognize what mID has modified to the display.

Vision Insensitivity to Adversary. The crafted Moiré patterns appeared in the photos shall look similar to the ones naturally generated by the screen-camera channel. Otherwise, the adversary may notice the existence of mID and abandon the image to avoid being traced.

4.2 Overview of mID

The basic idea is to generate mID by embedding identity numbers into the superimposed Moiré pattern via its intensity

levels, and the scheme consists of mID encoding and decoding phases with four modules: (a) mID generation, (b) mID embedding, (c) mID extraction, and (d) mID decoding, as shown in Fig. 4.

In the encoding phase, the mID Generation module first creates the modification that will be applied to the original display based on the IDs, and the mID Embedding module will find the best areas to apply such modification. The design goal of the encoding phase is that the modification cannot be observed visually by users but will be captured by cameras and form a seemingly natural Moiré pattern, i.e., mID. The mID Generation consists of (a) mID Framing that forms a proper frame, (b) Grating Generation that helps to create Moiré patterns, and (c) ID Encoding that adds the information of IDs to the Moiré patterns. Note that designing grating is similar to finding the carrier signals and the ID encoding is similar to finding the modulation scheme in communication.

To generate Moiré patterns, the screen pixels are manipulated to form a display grating, which has a periodic structure and may appear as stripes. To make the patterns looks as if they are naturally generated, the display grating is designed to be vertical since the LCD panel has a vertical grating structure. Second, to encode the mID into the display grating without noticed by users, we propose a discretized bipolar non-return-to-zero encoding method, which manipulates the intensity levels of the generated Moiré patterns to represent information. As humans perceive light and color in a non-linear manner, we correct the luminance difference caused by the discretized encoding to ensure user visual uniformity. Third, to embed the generated gratings into the screen and maximize their possibility of being captured in the photos, we automatically analyze the current page of the screen and search for suitable regions for embedding.

In the decoding phase, for a given screen photo that contains embedded mID, the mID Extraction module tries to remove the camera distortion with image rectification and extracts the regions of Moiré patterns, i.e., Moiré areas, with window scanning. Then, we recover the embedded identity numbers via the mID Decoding module, in which we first transform the Moiré areas into the HSV (hue, saturation, value) color space, then perform saturation balance and enlargement for image pre-processing, and finally recover mID via k-means clustering with the assistance of check codes.

4.3 mID Generation

4.3.1 mID Framing

To label the information source via Moiré-pattern-based ID, we design an N-bit mID that consists of (1) a 2-bit front check code, (2) a payload, and (3) a 2-bit end check code. The payload represents the identity of the information source and appears as a sequence of binary digits (bits), each having either the value "0" or "1". We envision it can provide photo forensics from three levels.

- **Device level.** When devices and users are tightly bound, e.g., the devices can only be accessed by the owners, the payload can be generated based on the hardware information of the display device, e.g., the MAC (media access control) address.
- **Operating system (OS) level.** When multiple users share the same device but use their own OS accounts, the payload can be generated at the OS level based on the OS user account information.
- **Application level.** For sensitive applications, e.g., the internal mail system or the database of companies, the payload can be generated based on the account information associated with the application.

The front (end) check code is a two-digit segment "01" that appears before or after the payload of mID. As thus, a 14-bit mID with front and end check codes appears as 01XXXXX...01. We design such a check code to facilitate decoding with twofold benefits. First, the check code can help restore the exposure-imbalanced images and can thus improve the decoding accuracy. Second, it provides a baseline for the k-means clustering to determine which cluster maps to bit "0" or "1", as we will reveal in detail in Sec. 4.6.

4.3.2 Display Grating Generation

As mentioned in Sec. 2, the screen pixels (layer l_1') is projected onto the camera sensors to form layer l_1, and the CFA of the camera forms layer l_2, with their superposition generating mIDs. Among the three layers l_1', l_1, l_2, we can only manipulate the screen pixels (l_1') for mID grating generation, since the CFA layer (l_2) is determined by the physical structure of smartphone built-in cameras and the projected screen display (l_1) is affected by the cameras as well. Recall that a periodical grating layer can be modeled with a frequency and a phase term:

$$l(x,y) = p(\phi(x,y)) \tag{3}$$

where $l(x,y)$ represents the pixel value at the coordinate (x,y), $p(\cdot)$ is a periodic function that determines the frequency of the grating, and $\phi(x,y)$ is a phase function that determines its geometric layout, as shown in Fig. 2. We explain how to select appropriate periodic and phase functions for the layer l_1' to generate mIDs.

Periodic Function Selection. Due to the long photographing distance and the pinhole effect of cameras, layer l_1 has an increased frequency compared with that of layer l_1'. According to the Pinhole Camera Theory [35], the object size projected onto a camera sensor is inversely proportional to the distance between the object and the camera sensor:

$$S_{cam} = \frac{S_{obj} \times L_f}{D} \tag{4}$$

where S_{cam} and S_{obj} are the photographed and actual sizes of the object respectively, L_f is the focal length of the camera, and D is the distance between the camera and the object. Due to that the photographing distance D is usually much larger than the focal length L_f, the size of layer l_1' shrinks to $\frac{L_f}{D}$ per

unit area, which gives $f_1 = \frac{D}{L_f} \cdot f_1^{'}$. As a result, the frequency of the generated Moiré patterns can be given as $\frac{D}{L_f} \cdot f_1^{'} - f_2$. As the camera focal length L_f and the CFA frequency f_2 are fixed by the photographing device, for a specific device, the Moiré patterns are mainly determined by the photographing distance and the frequency of the generated grating.

Considering the goal of adversaries is to capture the contents on the screen completely and clearly, the photographing distance D used by adversaries shall be within a range. For a 24" LCD display commonly-seen on the market, the photographing distance D is usually larger than 60 *cm* for various smartphones, as calculated in Appendix 11.1. To improve the chances of the generated Moiré patterns to be captured by cameras, the frequency of the periodic function $p(\cdot)$ shall match the photographing distance, and should be as small as possible since thinner strips are more likely to appear as uniformly colored compared with wider stripes. Thus, we set the frequency of $p(\cdot)$ to be 2 pixels, which is shown to be effective in Sec. 6.

Phase Function Selection. While the periodic term affects the density of the grating, the phase function determines its geometric layout and thus the Moiré patterns. Due to that in LCD panels, unit structures of the same color are usually arranged in vertical, the natural grating formed by the LCD display is vertical stripes of red, green, and blue respectively, as shown in Fig. 3. To achieve vision insensitivity to the adversary, we design `mID` that imitates the Moiré patterns that are generated naturally by the screen-camera channel. Specifically, with the selected frequency, we generate a binary display grating for each bit of `mID` in the form of vertical stripes, as given below:

$$\begin{aligned} l_1^{'}(x,y) &= p_1(\phi_1(x,y)) \\ p_1(u) &= 0.5 + 0.5cos(\pi u) \\ \phi_1(x,y) &= y \ mod \ 2 \end{aligned} \tag{5}$$

As the generated `mID` display grating has a frequency of 2 pixels while the digital screen structure has a frequency of 1 pixel, the `mID`-related and the natural Moiré patterns appear at different distances and will not interfere with each other.

4.3.3 Intensity-based ID Encoding

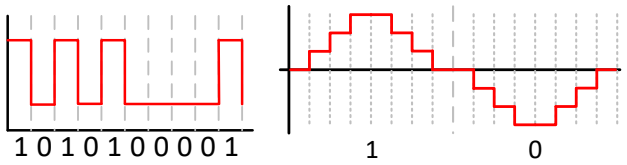
Existing work [17, 30, 43, 45] usually hides information in the Moiré patterns by manipulating the phase of one of the gratings, e.g., two secret images show no obvious patterns when observed separately, but reveal hidden information when overlapped. However, manipulating the phase is likely to bend the vertical stripes, and thus may result in visible changes to adversaries. In addition, phase-based methods usually induce significant patterns in the generated Moiré fringes, which is likely to alert the adversaries and is unacceptable.

Intensity of Moiré Pattern. To address it, we modify the intensity of Moiré patterns. Specifically, `mID`-related Moiré patterns are new frequency components generated by the superposition of the display grating and the camera CFA. Since the latter is determined by the camera, the intensity of the `mID`-related Moiré patterns depends on the intensity of the display grating. Thus, the intensity of the generated Moiré patterns can be changed by manipulating the pixel values of two adjacent grating stripes in the RGB color space, i.e., the contrast of two adjacent stripes. Such an observation is also validated by our experiments. As the generated grating has a spatial frequency of 2, we can denote the even column in the generated grating as $c_0 = (r_0, g_0, b_0)$, and the odd one as $c_1 = (r_1, g_1, b_1)$. With Equ. 6, $c_0 = (255, 255, 255)$ and $c_1 = (0, 0, 0)$ generate the most intensive Moiré patterns. Denote the color distance between two adjacent stripes, or in other words, a pair of color vectors $\{c_0, c_1\}$, as their l^2-norm in the RGB space:

$$C_d = ||\{c_0, c_1\}||_2 = \sqrt{(r_0 - r_1)^2 + (g_0 - g_1)^2 + (b_0 - b_1)^2} \tag{6}$$

A larger C_d represents a larger contrast between two adjacent stripes and thus represents a more significant stripe grating, which results in more intensive Moiré patterns. When C_d decreases to zero, i.e., the even and odd columns are identical, the generated grating loses its periodicity and thus no Moiré patterns will be observed.

Based on it, we propose to embed identity numbers into the generated Moiré pattern via its intensity levels. Intuitively, we can utilize the high intensity level to represent bit "1", and the low intensity level to represent bit "0", which is also known as the unipolar non-return-to-zero (NRZ) code [31], as shown in Fig, 5(a). However, such an implementation introduces discontinuity at the junction of bit "0" and bit "1", and thus may aggravate suspicious patterns to adversaries if they are adjacently encoded.



(a) Unipolar NRZ coding for binary sequence "101010001".
(b) Discretized bipolar NRZ coding for bi-nary sequence "10".

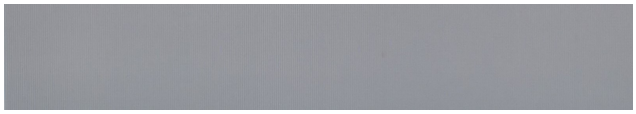
Figure 5: The improved discretized bipolar NRZ coding ensures a flat edge between bits "0" and "1".

Discretized Bipolar Non-return-to-zero Encoding. To alleviate the problem of discontinuity, we discretize both the high and low levels to make the possible junction smooth, which we call the discretized bipolar NRZ encoding. As shown in Fig. 5(b), we discretize the high (low) intensity level into k sub-levels with each sub-level consisting of n grating columns to approximate a cosine function, for the sake of being flat at the edge of a bit. Another benefit of such an implementation is that bipolar encoding increases C_d between bit "0" and bit "1" compared with the unipolar one, which may ease the difficulty of decoding.

Nonlinearity of Color Perception. In the discretized bipolar NRZ encoding, each intensity level is represented with

(a) Color vector pairs with identical distance.

(b) Color vector pairs with the same $\frac{c_0+c_1}{2}$ but increasing distances.



(c) Color vector pairs with luminance correction and increasing distances.

Figure 6: An illustration of ten intensity levels for encoding using three methods, and the ones created by the proposed luminance correction scheme (c) show almost no visual difference and can embed `mID` without being noticed by adversaries.

one pair of color vectors $\{c_0, c_1\}$ in the RGB space. Since bit "0" and bit "1" share the same baseline, the encoding requires $(2k-1)$ intensity levels in total, i.e., $(2k-1)$ pairs of color vectors with increased color distances.

We employ the visual average effect of human visual system (HVS) [51] to generate the required color vectors, which suggests that human eyes take the average of contiguous objects as their perception and many image scaling methods are built upon it [18]. As a result, we attempt to generate various color vectors for different intensity levels while keep their average RGB vector $\frac{c_0+c_1}{2}$ identical, which we assume may have the potential to ensure the evenness of the generated grating image.

Take the mid-gray $(128, 128, 128)$ as the background color for an instance, we can generate k pairs of vectors with increased color distances such as:

$$\begin{aligned} Level_i : \{c_0, c_1\}_i = \{&(128+5i, 128+5i, 128+5i), \\ &(128-5i, 128-5i, 128-5i)\}, i \in \{1, 2, ..., k\} \end{aligned} \tag{7}$$

Compared with the naive color vector pairs with an identical distance shown in Fig.6(a), the proposed ones, i.e., color vectors with the same $\frac{c_0+c_1}{2}$ but increasing distances, exhibit much fewer visual differences as shown in Fig. 6(b). Yet, it is insufficient to generate an even grating image. After careful analysis, we find it is because that adjusting the distance of $\{c_0, c_1\}$ changes its luminance perceived by human eyes, as a result of the Gamma Correction [25] adopted by modern display devices.

As humans perceive light and color non-linearly, with greater sensitivity to relative differences between darker tones than between lighter ones, gamma encoding is applied in images to optimize the usage of bits when encoding an image, or bandwidth used to transmit an image [32]. Correspondingly, modern display devices conduct gamma correction to reveal the true colors. Both gamma encoding and gamma correction follow a pow-law expression [47]:

$$V_{out} = A V_{in}^{\gamma} \tag{8}$$

where the input value V_{in} is multiplied by the constant A and powered by the gamma value γ to get the output value V_{out}, with $\gamma < 1$ for encoding and $\gamma > 1$ for correction (decoding). As a result, the generated RGB vector is expanded before display and the luminance perceived by human eyes is not the arithmetic mean $\frac{c_0+c_1}{2}$ as supposed.

Luminance Correction. To further make the encoding unnoticeable, we propose a luminance correction algorithm based on gamma correction and the non-uniformity color perception of HVS. Specifically, we model the average luminance Y of an RGB vector pair $\{c_0, c_1\}$ by removing the gamma compression, which transforms the image to a linear RGB color space as follows:

$$Y\{c_0, c_1\} = w_r(r_0^{\gamma} + r_1^{\gamma}) + w_g(g_0^{\gamma} + g_1^{\gamma}) + w_b(b_0^{\gamma} + b_1^{\gamma}) \tag{9}$$

where $\gamma = 2.2$ for most modern display devices [47]. w_r, w_g and w_b are the weights of the RGB channels respectively, which represent the intensity (luminance) perception of typical humans to lights of primary colors. Given that human vision is most sensitive to green and least sensitive to blue, w_g has the largest value of 0.7152 and w_b has the smallest value of 0.0722, with $w_r = 0.2126$ [48].

With luminance correction, we can generate RGB vector pairs with even luminance by optimizing the following equations:

$$\begin{aligned} & E = |Y\{c_0, c_1\} - Y_{bg}| \\ & Y_{bg} = w_r r_{bg}^{\gamma} + w_g g_{bg}^{\gamma} + w_b b_{bg}^{\gamma} \\ & \textbf{\textit{max}} \quad C_d = ||\{c_0, c_1\}||_2 \\ & \quad s.t. \quad E < \varepsilon \\ & \quad s.t. \quad r_i, g_i, b_i \in \mathbb{Z} \cap [0, 255], i = 0, 1 \end{aligned} \tag{10}$$

We utilize the global search algorithm to solve the above optimization problem. However, as we can see, the solution to the formula is not unique and the number of searched vector pairs is determined by the error threshold ε. A larger ε contributes to more RGB vector pairs at the cost of less evenness of the generated grating image. Thus, ε can be determined upon the requirement of k, or in other words, the number of RGB vector pairs needed to implement the discretized bipolar NRZ encoding. After luminance correction, the generated grating is almost invisible even with increasing color distances, as shown in Fig. 6(c).

In summary, we utilize the discretized bipolar NRZ encoding to embed identity numbers into the generated Moiré pattern via its intensity, and employ luminance correction to ensure the evenness of the generated gratings.

4.4 `mID` Embedding

To embed the generated gratings and maximize their possibility of being captured in photos, we automatically analyze the

current page of the screen.

Region of Interest. Given the company's goal is to prevent cyber-theft of trade secrets, some regions of the current page that contain confidential information such as texts or images, are of more interests to the company, i.e., regions of interest (ROIs). To search for suitable regions for `mID` embedding, we first locate the possible ROI of the current page with computer vision (CV) techniques [14, 24, 54], which mainly extract the locations of texts and images, as shown in Fig. 7. The number of ROI extracted is determined by the screen content, and we calculate the centroid of these regions as the center of ROI for the current page. Alternatively, the defenders can manually mark the ROIs according to their demands.

Region of Embedding. To maximize the possibility that `mID` is captured in the screen photos, we embed the generated gratings in the vicinity of the ROI center, i.e., regions of embedding (ROEs). In general, we assume that flat regions close to the ROI center are more suitable for embedding since (1) `mID` is more likely to be captured in the screen photos, and (2) fewer details of the current page will be lost and less vision disparity will be caused to users. In addition, we design to embed one bit of `mID` in each ROE. It is because that embedding the whole `mID` in one ROE may require a large flat region. Separating the `mID` into several ROEs helps to reduce the size requirement of ROEs.

Therefore, we search for N rectangular regions close to the ROI center, where N is the number of bits of `mID`. Each embedding region has a size of $p \times q$, where p and q represent the height and width of a 1-bit grating, respectively. The width q can be further calculated as $q = 2k \times n$. The height p can be any value theoretically but a minimum one is required to ensure the distinguishability of Moiré patterns in the screen photos. In practice, we suggest that $p > 50$. Note that the embedding region can be any shape. We employ rectangle here for the ease of encoding and decoding.

We utilize a sliding window with a size of $p \times q$ and a step of w_m to scan through the current page for ROE searching. For each image window $B(x,y)$ with (x,y) as the centroid coordinate, we evaluate its fitness $F(x,y)$ in consideration of both evenness and location:

$$\begin{aligned} D(x,y) &= \frac{1}{\sum\limits_{ch=\{r,g,b\}} \sigma(ch[x-\frac{p}{2}:x+\frac{p}{2}, y-\frac{q}{2}:y+\frac{q}{2}])} \\ L(x,y) &= \frac{1}{abs(\frac{x}{h_B}-C_x)+abs(\frac{y}{w_B}-C_y)} \\ F(x,y) &= w_D \cdot D(x,y) + w_L \cdot L(x,y) \end{aligned} \tag{11}$$

where $\sigma(ch)$ refers to the standard deviation of channel $ch = \{r,g,b\}$ of the current page. h_B and w_B are the height and width of the current page, (C_x, C_y) is the centroid coordinate of ROI, and w_D and w_L are the weights of the deviation $D(x,y)$ and location $L(x,y)$ functions, respectively. In our implementation, we set $w_D = w_L = 0.5$.

We employ the first N image windows in the descending order of fitness ranking as our ROE, and rearrange them according to the horizontal coordinates (in an ascending order)*. As thus, we obtain N image windows in the horizontal direction. With the obtained regions, we embed the corresponding `mID` bits by replacing the pixels of the original page with that of the generated gratings. As thus, we embed the generated `mID` gratings into the current page of the screen, under the premise of non-obvious visual impact to users.

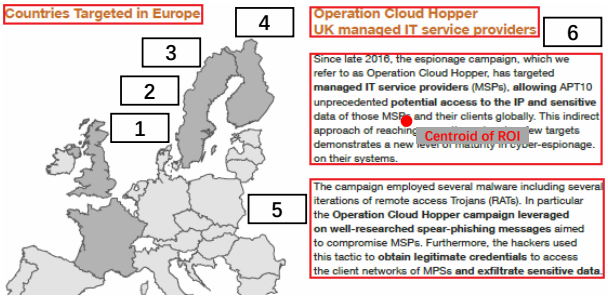


Figure 7: Illustration of ROI (red box) and ROE (black box) of the current page [26]. The red dot is the center of ROI.

4.5 `mID` Extraction

The image captured by smartphones contains Moiré patterns as well as other elements. To obtain the embedded `mID`, we first locate the regions of Moiré patterns in the smartphone-captured image, which we call the Moiré areas.

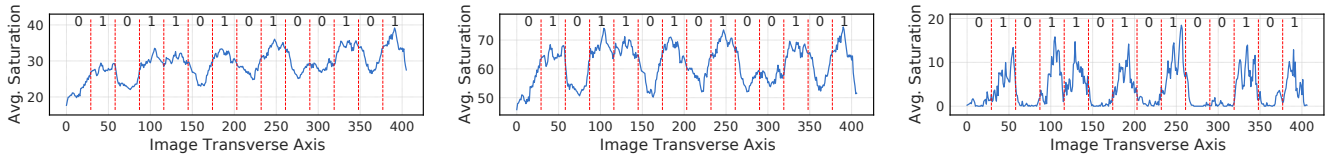
Image Rectification. Smartphone-captured images usually suffer from geometric distortion due to the unparalleled camera and screen planes, i.e., an angle exists between them. As a result, the captured screen is no longer a regular rectangle but a distorted quadrilateral. To address it, we first rectify the distorted image with the commonly-used projection transformation under the homogeneous coordinates [8, 53], and then extract the rectified rectangle that contains the screen for further Moiré area extraction.

Moiré Area Extraction. One intuitive method to extract the Moiré areas is to search for the red-green fringes. However, as Moiré patterns may appear as various colors on different backgrounds and blur due to the noise introduced by the screen-camera channel, simply searching for fringes of a specific color may not suffice. Therefore, we turn to the transverse coding style we employ for `mID` encoding, because of which the Moiré area is likely to have larger color variations in the horizontal direction compared to the vertical one.

To extract the Moiré areas with robustness, we use a 2-dimension (2D) window W_m with a size of $h_m \times w_m$ and a step of t_m to scan through the rectified rectangle image. Specifically, we calculate the average color variation Var_h and Var_v in both the horizontal and vertical directions, and determine whether the current window belongs to the Moiré area with the following in-equation:

$$Var_v > r \cdot Var_h \tag{12}$$

*Arrange by vertical coordinates If equal horizontal coordinates.



(a) Saturation of exposure-unbalanced JMA. (b) Saturation of exposure-balanced JMA. (c) Saturation of difference-enlarged JMA.

Figure 8: After pre-processing, the saturation of JMA is balanced and the difference between bits "0" and "1" is enlarged.

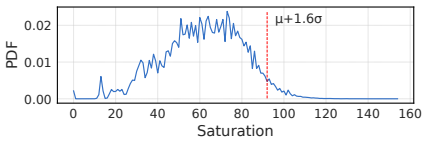
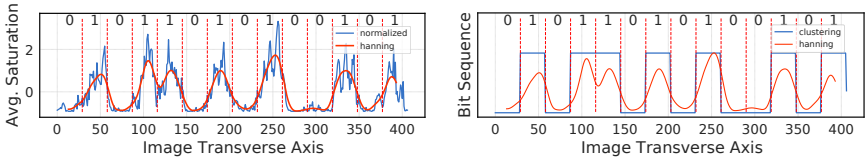


Figure 9: The JMA saturation distribution is roughly a Gaussian distribution.



(a) Saturation curve of the pre-processed Moiré area. (b) Bit sequence recovered by bit clustering.

Figure 10: An illustration of mID recovered by k-means clustering with check codes.

where r is the ratio threshold and the window with significantly larger horizontal variation will be regarded as a part of the Moiré area. To achieve high extraction precision, the window size and step are usually supposed to be in fine granularity. In practice, we set $h_m = w_m = s_m = 10$ *pixels*, and $r = 1.5$. After scanning, we obtain a number of Moiré windows in several clusters with possibly a few outliers. The number of clusters, i.e., the number of Moiré areas contained in the photo, is usually less than or equal to the number of mID bits N since two adjacent embedding regions appear as one Moiré area in the photos. To locate the Moiré areas, we first cluster those Moiré windows with mean shift clustering [6], which obtains the center of each Moiré area roughly. Then, we utilize Random Sample Consensus (RANSAC) [11] to discriminate outliers and search for the minimum rectangle that contains the rest of clustered Moiré windows for each Moiré area. We gradually iterate their boundaries until convergent, with which we extract the Moiré areas for further mID decoding.

4.6 mID Decoding

After extracting the Moiré areas, we perform mID decoding to recover the embedded mID. To ease burden of decoding, we arrange and connect the obtained Moiré areas together according to their horizontal coordinates. In this way, we obtain a joint Moiré area (JMA) for decoding.

4.6.1 Image Pre-processing

The first set of decoding procedures is image pre-processing that includes (1) Color Space Transformation that makes the decoding algorithm robust across different colors, (2) Saturation Balance that reduces the impact of focus position and the ambient light, and (3) Saturation Difference Enlargement that enlarges the saturation difference between bit "0" and bit "1" to help decoding.

Color Space Transformation. The colors of the mID-related Moiré patterns depend on the screen backgrounds. For instance, a white background will produce Moiré patterns with red and green stripes. To make the decoding algorithm robust across different RGB colors, we transform the joint Moiré area into the HSV (hue, saturation, value) color space [49]. Specifically, as we utilize the Moiré pattern intensity to encode bits and high intensity results in high color saturation, we perform mID decoding in the saturation dimension.

Saturation Balance. When taking a picture towards a screen, people tend to focus on the center of the screen to capture the whole screen. As a result, the Moiré areas close to the focus may be better exposed compared to the remote ones, as shown in Fig. 8(a). To reduce the impact of focus position as well as the ambient light, we balance the saturation of the joint Moiré area with the help of check codes.

Specifically, we focus on the horizontal saturation balance since we encode mID in a transverse way and thus horizontal saturation unbalance has a larger impact on decoding compared to that in the vertical direction. To address it, we divide the joint Moiré area into N splits by width, where the 1^{st} and 2^{nd} splits correspond to the bit "0" and bit "1" of the front check code, and the $(N-1)^{th}$ and N^{th} splits correspond to that of the end check code. We compare the average saturation of the 2^{nd} and N^{th} splits and enhance the side with lower saturation. The image enhancement algorithm we employ manipulates every pixel of the image and balances the saturation in a horizontal and linear way, as shown in Algorithm 1 in Appendix 11.2. For exposure-imbalanced images, saturation balance is able to restore the actual Moiré patterns as shown in Fig. 8(b), and thus can improve the decoding accuracy.

Saturation Difference Enlargement. After saturation balance, we enlarge the saturation difference between bit "0" and bit "1" to improve the decoding efficiency. In general, the Moiré area of bit "1" is likely to have more pixels with large saturation values compared with that of bit "0". However, the noise (e.g., the Gaussian and salt-and-pepper noise [4], which are common in photos) introduced during the process of photographing may blur the image and thus increase the difficulty of decoding. To ease the burden, we perform saturation difference enlargement. Specifically, we assume that the saturation values of pixels in the joint Moiré area are in concordance with the Gaussian distribution based on the Cen-

tral Limit Theorem (CLT) [10], i.e., $S \sim \mathcal{N}(\mu, \sigma^2)$, as shown in Fig. 9. Based on it, we enhance the discrepancy between bit "0" and bit "1" as follows:

$$s(x,y) = \begin{cases} 0 & s(x,y) < \mu + \alpha \cdot \sigma \\ s(x,y) & s(x,y) \geq \mu + \alpha \cdot \sigma \end{cases} \quad (13)$$

where α is the amplification factor. An appropriate α is able to reduce the saturation intensity of bit "0" while maintain that of bit "1", and thus can help enlarge their differences, as shown in Fig. 8(c). In practice, we set α to be 1.6.

4.6.2 ID Recovery

After image pre-processing, we recover IDs via k-means clustering with the assistance of check codes.

Saturation Curving. With the enhanced joint Moiré area, we first calculate the histogram of each column and obtain a $1 \times W$ saturation matrix, where W is the length of the joint Moiré area. It is based on the transverse encoding we employ, which means that pixels of the same column are supposed to be identical. We then perform normalization on the matrix and utilize a Hanning window to reduce the noise introduced during photographing and improve the SNR (signal-to-noise ratio). As thus, we obtain a horizontal saturation curve for further decoding, as shown in Fig. 10(a).

Bit Clustering. For an N-bit `mID`, we further divide its saturation curve into N splits and calculate the saturation sum of each split as the value of the corresponding bit, denoted as $\{P_0, P_1, ..., P_{N-1}\}$. Since the processed saturation sequence may have outliers (abnormally large values in our case) that are likely to affect the clustering threshold, we reduce their impacts by suppressing data points with large values. Specifically, for an N-bit `mID`, we decrease the largest K data points as follows:

$$P^{'} = \frac{P}{\sigma'^{\beta}} \quad (14)$$

where σ' is the standard deviation of the saturation sequence $\{P_0, P_1, ..., P_{N-1}\}$, and β is the decreasing factor. A larger β suppresses outliers more heavily. In practice, we set $\beta = 0.1$.

After that, we employ k-means clustering [50] to group the same bit into the same class and utilize the check codes to identify each class, i.e., bit "0" or bit "1", as shown in Fig. 10(b). In this way, we recover `mIDs` from screen photos and the whole decoding process is shown in Algorithm 2 in Appendix 11.2.

5 Implementation

We implement `mID` scheme at both the OS and application levels in Windows, where `mID` runs as a background application or a script after a user logs in, receptively. For the OS level, `mID` employs the entire screen as the display window and creates a rendering context using the Windows API functions `GetDC()` and `wglCreateContext`. For the application level, `mID` employs the application window as the display window and uses its own rendering context. Then, `mID` captures the current page of the screen or application in real-time using the function `glReadPixels()` under the OpenGL (Open Graphics Library) framework [19]. After that, it searches for the ROI and ROE with the methods proposed in the `mID Embedding` module. With the obtained ROE, `mID` replaces the pixels of ROE with the gratings generated by the `mID Generation` module, passes the new `mID`-embedded screen (application) frame to the function `glBufferData()`, and finally renders it on the display.

6 Evaluation

In this section, we evaluate the performance of the `mID` scheme. We conduct experiments under various settings and collect over 5000 photos with 5 display devices and 6 smartphones over 3 months. In particular, we evaluate the impact of (1) IDs, (2) display devices, (3) capturing devices, (4) ambient lights, (5) shooting distances, and (6) shooting angles with the metrics of bit error rate (BER) and identity number error rate (NER). In addition, we evaluate the performance of `mID` against several photo processing attacks. The performance of the `mID` scheme is summarized below:

- `mID` achieves an average BER of 0.6% and an average NER of 4.0%, which demonstrates promises towards screen photo forensics.
- `mID` performs well with little influence from the type of display devices, cameras, IDs, and ambient lights.
- `mID` performs well at a shooting distance of $(60cm, 80cm)$ and a shooting angle of $(-20°, 20°)$, which are within the possible attack distances and angles adopted by adversaries as suggested by the theoretical calculation (in Appendix 11.1).

6.1 Experiment Setup

We evaluate `mID` scheme in a laboratory setting with various display and capturing devices. The detailed settings are as follows.

Display Device. We use a BenQ EW Series LCD screen as the default display device. To evaluate the impact of display devices, we use 2 other LCD displays and 2 laptops of different brands and models. Throughout the experiments, the display devices remain in the default settings with normal color mode and 50% screen brightness. The detailed information of each display device is shown in Tab. 1.

Capturing Device. We use an LG Nexus 5X smartphone as the default capturing device. In addition, we use 5 other smartphones of various brands and models to evaluate the impact of capturing devices. Throughout the experiments, the capturing device is clamped on a tripod with a height of 30 *cm* from the desk and alighted with the center point of the display screen, as shown in Fig. 11. The shooting distance and angle are set to 70 *cm* and 0° respectively. We use the main camera of each device in the default settings, with Auto-focusing

Table 1: Summary of display devices.

No.	Manuf.	Model	Display Size	Aspect Ratio	Viewing Area	Native Resolution	Panel Type	Backlight
1	BenQ	EW2440ZC	24"	16:9	53.1 cm × 29.9 cm	1920 × 1080	MVA	LED
2	HP	24w	23.8"	16:9	52.7 cm × 29.6 cm	1920 × 1080	IPS	LED
3	AOC	LV243XIP	23.8"	16:9	52.7 cm × 29.6 cm	1920 × 1080	IPS	LED
4	Lenovo	IdeaPad Y700	15.6"	16:9	34.5 cm × 19.4 cm	1920 × 1080	IPS	LED
5	ASUS	FX50J	15.6"	16:9	34.5 cm × 19.4 cm	1920 × 1080	IPS	LED

MVA: Multi-domain Vertical Alignment. IPS: In-Plane Switching

Table 2: Summary of main camera specifications of the capturing devices.

No.	Manuf.	Model	Camera	Resolution	Aperture	Focal Length†	Pixel Size	Image Size	AF‡	HDR§
1	LG	Nexus 5X	Single	12.3 MP	f/2.0	5 mm, 26 mm (wide)	1.55 μm	4032 × 3024	√	√
2	HUAWEI	Mate 10	Dual	12 MP 20 MP B/W	f/1.6 f/1.6	4 mm, 27 mm (wide) 4 mm, 27 mm (wide)	1.25 μm 1.25 μm	3968 × 2976	√	√
3	HUAWEI	P9	Dual	12 MP 12 MP B/W	f/2.2 f/2.2	4.5 mm, 27 mm (wide) 4.5 mm, 27 mm (wide)	1.25 μm 1.25 μm	3968 × 2976	√	√
4	Apple	iPhone X	Dual	12 MP 12 MP	f/1.8 f/2.4	4 mm, 28 mm (wide) 6 mm, 52 mm (telephoto)	1.22 μm 1.0 μm	4032 × 3024	√	√
5	Motorola	G4 Plus	Single	16 MP	f/2.0	5 mm, 27 mm (wide)	-	4608 × 2592	√	√
6	Vivo	Xplay3S	Single	13 MP	f/1.8	4 mm, 28 mm (wide)	-	4128 × 3096	√	√

† Physical (former) and equivalent (latter) focal lengths for smartphone's built-in cameras. ‡ AF: Auto-focusing
§ HDR: High Dynamic Range Imaging

(AF) and High Dynamic Range Imaging (HDR) activated. No other image processing techniques, e.g., filters, are used during the experiments since (1) not all smartphones provide these techniques, and (2) they are not activated by default. The detailed parameters of the cameras are shown in Tab. 2. Note that at the time of writing, we find no Moiré pattern filter functions available on smartphones on the current market.

Ambient Light. We conduct most experiments under the artificial lights produced by LEDs ($\sim 200\ lm$), as it is the most likely attack environment in practice. In addition, we conduct experiments under the case of (1) natural lights ($\sim 20\ lm$), and (2) no additional lights except for those from the display screen ($< 5\ lm$), to evaluate the impact of ambient lights.

Application Scenario. Without loss of generality, we study the PDF document as an illustration of confidential files and use Adobe Reader as the default document browser under the standard reading mode in this paper. The PDF document used in the experiments contains texts only. In addition, we conduct experiments with (1) Microsoft Word, (2) JetBrains PyCharm 2017, and (3) Google Gmail Web Client with 4 various background colors. Due to the space limitations and the similar performance across these applications, we demonstrate the results of Adobe Reader only. Note that mID scheme is applicable to both text-only and image-contained files. For instance, the Google Gmail Web Client has several images and logos in the background, and the mID scheme is able to cooperate with it as well.

Encoding Parameter. We choose 14-bit mID as an illustration in this paper, i.e., $N = 14$. As such, each generated mID is consisted of a 2-bit front check code, a 10-bit information code, and a 2-bit end check code, i.e., in a form of 01XXXXX...01. For the discretized bipolar NRZ encoding, we employ 4 sub-levels with each sub-level consisting of 4 grating columns, i.e., $k = 4$ and $n = 4$. Note that all the aforementioned parameters are not mandatory and can be adjusted based on user requirements.

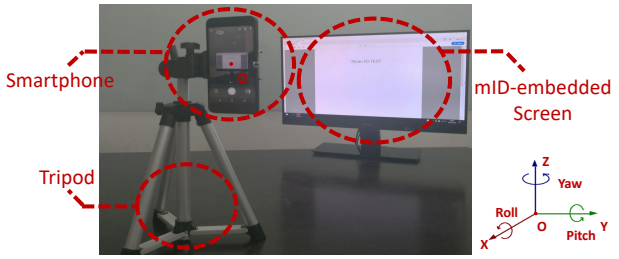


Figure 11: The current page of the display is embedded with mID, which can be captured by the built-in cameras of smartphones.

6.2 Performance Metrics

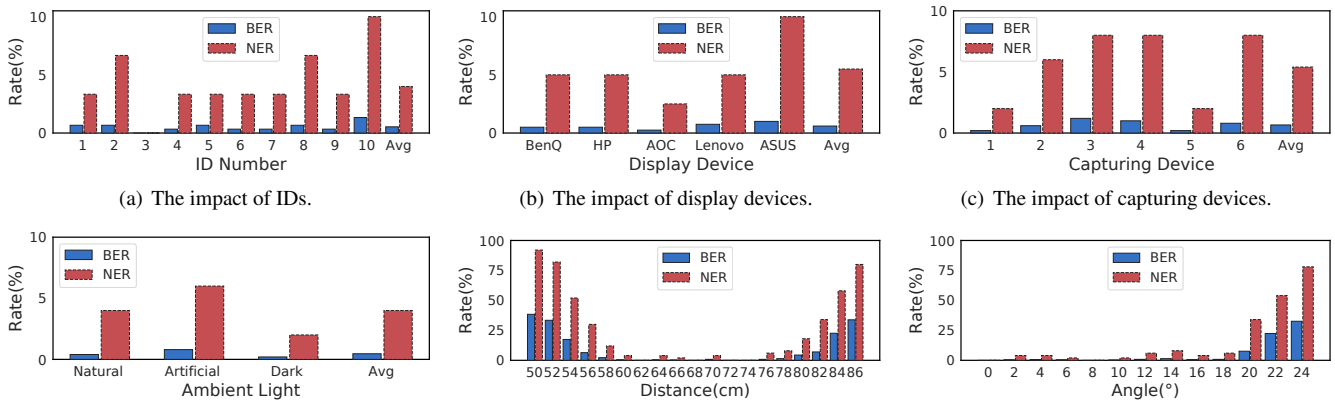
We use BER (bit error rate) and NER (identity number error rate) to evaluate mID from two different perspectives.

BER. BER refers to the number of bit errors divided by the total number of mID bits (excluding check codes), which evaluates the performance of mID decoding in a fine granularity.

NER. NER refers to the number of the IDs that were not correctly decoded (IDs with at least one bit error) divided by the total number of mIDs. Thus, NER is a stricter criterion compared with BER and demonstrates the effectiveness of the proposed mID method.

6.3 Overall Performance

In this section, we first evaluate the overall performance of mID decoding with various IDs, and then evaluate the impact of the aforementioned factors including display devices, capturing devices, ambient light, etc.



(a) The impact of IDs. (b) The impact of display devices. (c) The impact of capturing devices.

(d) The impact of ambient lights. (e) The impact of shooting distances. (f) The impact of shooting angles.

Figure 12: Performance of mID decoding under various settings.

6.3.1 Impact of IDs

In the first set of experiments, we evaluate the overall performance of mID with various IDs. We randomly generate 10 mIDs, embed them into the PDF files with mID generation and mID embedding, and then display the modified files on the default LCD monitor, respectively. We then capture 30 photos with the Nexus 5X smartphone for each mID. During the photographing, we use the default camera settings.

We perform mID extraction and mID decoding on the captured photos for each mID. The results in Fig. 12(a) reveal that mID scheme achieves an average BER and NER of 0.6% and 4.0%, respectively. Specifically, ID 3 achieves the best performance of 0 BER and NER while ID 10 achieves the worst with a BER of 1.3% and an NER of 10.0%. Although the NER of each ID varies due to the limited samples as well as the randomness introduced during photographing, the BER remains relatively low and stable, demonstrating the effectiveness of the mID decoding algorithm.

6.3.2 Impact of Display Devices

In real-world deployment, display devices may have various types and models. However, as mID does not use any explicit device attribute during design, mID should be compatible with any display devices. To investigate it, we utilize two other LCD monitors and two laptops with different screen sizes and panel types to display the modified files. The details of each display device are summarized in Tab. 1. With the default settings, we utilize the default capturing device and collect 40 photos for each display device.

The decoding results in Fig. 12(b) reveal that laptop screens show relatively higher NER and BER compared with LCD monitors. ASUS shows the worst performance with a BER of 1.0% and an NER of 10.0% while AOC shows the best with a BER of 0.3% and an NER of 2.5%. We believe it is because laptops have smaller screen sizes compared with LCD monitors. As a result, the Moiré area occupies less area in the photos displayed on laptops, and thus is more likely to suffer from noise and more difficult to distinguish. Nevertheless, mID can still achieve an average BNR of 0.6% and an average NER of 5.5% among various display devices.

6.3.3 Impact of Capturing Devices

In practice, adversaries may use any smartphone to take pictures. To investigate whether mID works well under various smartphones, we conduct experiments with 5 other smartphones from different brands and models, in addition to the default capturing device Nexus 5X. The details of each capturing device are summarized in Tab. 2. We utilize the main camera (single or dual) of each smartphone to collect 50 photos respectively, with the auto-focusing setting.

From the results shown in Fig. 12(c), we can observe that single-camera smartphones achieve better performance compared with dual-camera ones. For instance, Nexus 5X and Motorola G4 Plus perform best in the experiments with a BER of 0.2% and an NER of 2.0%, which are both single-camera phones. By contrast, dual-camera devices suffer from relatively higher NERs, e.g., HUAWEI P9 performs the worst with a BER of 1.0% and an NER of 8.0%. We believe it is because that dual-camera devices utilize images from both cameras to composite the final photo, which may have impact on the Moiré patterns and thus the decoding results. Overall, mID can achieve an average BER of 0.7% and an average NER of 5.4% with capturing devices various in resolution, aperture, and focal length.

6.3.4 Impact of Ambient Lights

During photographing, the ambient lights are likely to affect imaging and mID decoding. To investigate the impact of ambient lights, in addition to the artificial lights produced by LEDs ($\sim 200\ lm$), we conduct experiments under two other light conditions, i.e., (1) natural lights ($\sim 20\ lm$), and (2) no additional light except for the one from the display screen (i.e., dark environment ($< 5\ lm$)). For each light condition, we collect 50 photos and perform mID decoding.

The results in Fig. 12(d) demonstrate that the dark environment helps to improve the decoding performance while

artificial lights have negative effects. Specifically, the dark environment achieves the best BER of 0.2% and NER of 2.0%, followed by the natural environment with a BER of 0.4% and an NER of 4.0%. The artificial environment performs the worst with a BER of 0.8% and an NER of 6.0%. We believe it is because that the LEDs in the experimental room are multiple and decentralized. As a result, the light source is heterogeneous during photographing, which may cause the unevenness of exposure and thus decrease the decoding accuracy. Nevertheless, mID can still achieve an average BER of 0.5% and an average NER of 4.0% with various light conditions.

6.3.5 Impact of Photograph Distances

Theoretically, adversaries may take a photo from any distance. However, since the goal of the adversary is to record the information on the screen, the picture is likely to be taken at a reasonable distance and angle. To investigate the impact of photograph distance, we first survey the common shooting distance adopted by normal volunteers, which turns out to be in the range of 50 *cm* - 100 *cm* for the sake of capturing the screen well. We then conduct experiments during this range with a step of 2 *cm*. For each distance, we collect 50 photos and perform mID decoding.

From the results in Fig. 12(e), we can observe that mID achieves the best performance with a shooting distance around $58-80$ *cm*. With a photograph distance either >84 *cm* or <56 *cm*, mID decoding accuracy drops due to that the generated Moiré patterns become invisible to both human eyes and camera sensors. Overall, mID decoding can achieve an average BER of 0.4% and an average NER of 2.7% under the distance range of $(60\,cm, 80\,cm)$. In addition, according to the calculation shown in Appendix 11.1, to capture a 24" display screen completely, the photograph distance D is usually larger than 60 *cm* for various smartphones. Therefore, we believe that mID is basically sufficient to cover the possible attack distances adopted by adversaries.

6.3.6 Impact of Photograph Angles

In addition to the photograph distance, we investigate the impact of shooting angles from three degrees-of-freedom, i.e., roll, pitch, and yaw, as shown in Fig. 11. The first degree-of-freedom roll rotates the image captured by the camera in the x-y plane, and we can reduce its impact by image rotation. The second and third degrees-of-freedom pitch and yaw mainly cause vertical and horizontal deformation in the captured image respectively. In real attacks, the former may have little impact since we employ the transverse encoding, which means pixels of the same column are supposed to be identical and thus vertical deformation may not affect the information representation. Besides, both the vertical and horizontal deformation can be addressed with rectification techniques [8, 53]. Therefore, we mainly evaluate the impact of the last degree-of-freedom, i.e., yaw, in this paper since it is most relevant to mID scheme.

During the experiments, we take the symmetry axis of the smartphone screen as the center axis and rotate the yaw angle with a step of 2°. We set the shooting distance to default throughout the experiments and the smartphone is tangent to the arc consisted by its motion locus. Without loss of generality, we start from the central point where the smartphone is paralleled with the display screen, i.e., 0°, and increase the angle of inclination in both clockwise (+) and anticlockwise (-) directions. For each angle, we collect 50 photos and perform mID decoding.

From the results shown in Fig. 12(f), we can observe that mID achieves a relatively low BER and NER with a photograph angle less than 20°. When the inclination angle is further increased, the distortion of Moiré stripes becomes non-negligible and difficult to be corrected, and thus may affect the performance of mID extraction and decoding. However, we argue that with an inclination angle larger than 20°, the image content is heavily distorted as well, which may also deviate from the goal of the adversaries. Overall, mID is able to achieve an average BER of 0.5% and an average NER of 3.6% with a photograph angle within $(-20°, 20°)$.

7 Preliminary User Study

We measure whether users will notice the presence of mID and how users cope with mID-related Moiré patterns in the screen photos by conducting a user study among 34 volunteers. Most of them are graduate students aged 20-30 years old. We followed the local regulations to protect the rights of human participants despite the absence of Institutional Review Board (IRB).

To study whether users can recognize the presence of mID, we conduct the following test: on an LCD monitor in an office room, we display a PDF document using Abode Reader, which is an IELTS essay about news and we embed mID in both sides of the document body. The participants are required to sit in front of the monitor and provided 5 minutes to read the essay. After reading, we conduct a questionnaire survey for each participant, in which we ask three choice questions and three 7-point scale questions, and the detailed descriptions of each question are summarized in Tab. 4 in Appendix 11.3. For comparison, we conduct another contrast test using a PDF document without mID. From Tab. 4, we can see that the first 3 choice questions are essay-content-related, which are the superficial tasks of the test. The real aim is to learn whether the participants feel or notice any visual abnormality during the process of reading (Question 6) and we cover it up with two transitional questions (Question 4 and 5). The results shown in Fig. 13 demonstrate that the participants hardly perceive the existence of mID in the course of normal use (with an average of 1.147, a standard deviation of 0.429, a 95% confidence interval of [1.003,1.291] on a 7-point scale) compared with the mID-free situation (with an average of 1.118, a standard

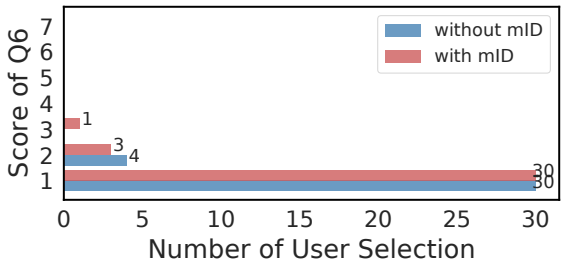


Figure 13: Scores of Question 6 when the screen is embedded with/without mID.

deviation of 0.322, a 95% confidence interval of [1.009,1.226] on a 7-point scale). Thus, mID should be able to satisfy the requirement of no visual impact on users.

To study how well the decoding technique works for the realistic photos taken by attackers, we conduct a real-world experiment by asking each volunteer to take 5 photos towards the mID-embedded screen after finishing the questionnaire survey, with the imagination of leaking important information to competitors and the need of capturing the information on the screen completely and clearly. The results illustrate that for the 170 photos taken by the volunteers, an average decoding accuracy of around 95% can be achieved. In addition, the results demonstrate that more than 91% (31/34) volunteers take photos with Moiré patterns as they are used to them. The other 3 users carefully adjust the shooting angle and distance to avoid Moiré patterns. However, the adjustment is not adopted by most users since it may twist the photo content. It indicates that the attackers are likely to include Moiré patterns in the screen photos, although studying people's willingness will need to be further studied in the future.

8 Discussion and Limitations

In this section, we discuss several issues of mID as well as its limitations.

In-camera Image Processing. Modern smartphones utilize in-camera image processing techniques such as auto-focusing, optical anti-vibration, HDR, and multi-camera system to form a better picture. Among those, HDR combines a sequence of photos to achieve a greater dynamic range of luminosity, and we use it by default in our evaluation. To illustrate the impact of HDR, we conduct a contrast experiment with the HDR-deactivated Nexus 5 smartphone. The results demonstrate that HDR can reduce the BER from 0.4% to 0.2% for the Nexus 5 smartphone. We assume it is because that HDR can increase the luminosity difference between the bit "0" and "1", and is beneficial to mID decoding. The multi-camera system combines photos from each individual camera to achieve a better depth of field. In our evaluation, we use 3 single-camera and 3 dual-camera smartphones, and the results shown in Sec. 6.3.2 demonstrate that dual-camera phones show a slightly higher NER but can still achieve a good performance ($\sim$ 93%).

Table 3: Impact of photo processing.

No.	Photo Processing Technique	Defense
1	Image Duplication (copy and paste)	√
2	Image Compression (lossless)	√
3	Image Upscaling	√
4	Image Downscaling	Partial
5	Format Conversion (PNG to JPG)	√
6	Image Cut	Partial

Post-camera Image Processing. We adopt the same design assumption as the ones of watermarking or steganographic techniques, i.e., adversaries are unaware of the technique, and we hide information by embedding IDs in Moiré patterns that appear natural. Nevertheless, in a rare case, the adversary may process the captured screen photos to reduce the risk of being traced, as mentioned in Sec. 3. The possible post-camera processing techniques include two types: (1) commonly-used image editing operations such as photo duplication, photo compression, image up/downscaling, format conversion, and image cut, and (2) specially-designed evasion algorithms targeted at removing Moiré stripes.

For the former, we randomly choose 10 photos from the screen photos collected under the default settings and conduct experiments to investigate whether mID can resist these attacks. From the results shown in Tab. 3, we can see that mID can successfully resist the attacks of photo duplication (copy and paste), compression (lossless), upscaling (any upscaling ratio), and format conversion (PNG to JPG). The reason is that those attacks do not or hardly cause information loss of the screen photos (upscaling even increases the amount of information contained in the photos), thus have no obvious impact on mID decoding. The rest of the attacks, on the contrary, may affect the content of the photo and thus the decoding. For image downscaling, we evaluate its impact by setting the downscaling ratio to 0.9, 0.8, ..., 0.1, and the information loses uniformly. The results show that mID can achieve a good performance (>90%) with a downscaling ratio larger than 0.6. With a smaller ratio, e.g., 0.5, many details of the photos, including the Moiré patterns, are lost, resulting a performance decrease. However, in this case, the content in the photo is blurred as well, which may affect the reading. For image cut, as we only embed mID in the vicinity of ROI, removing other photo areas do not impact the decoding of mID. If the adversary must remove the Moiré areas, which is possible but may be difficult since they are usually surrounded by ROI, we may not obtain enough information to recover the embedded mID. Thus, in general, mID is able to resist the attacks of photo duplication, photo compression, image upscaling, and format conversion, and partial attacks of image downscaling and image cut.

For the latter, existing Moiré pattern evasion approaches mainly have three categories: (1) adding an optical low-pass filter (OLPE) over the camera lens, (2) using an enhanced color interpolation algorithm, and (3) employing post image processing techniques. The first two categories are both pre-

ventive measures and implemented within the cameras, thus are not capable of removing existing Moiré stripes contained in the screen photos. For the last category, however, automatically removing Moiré patterns from a single photo is still challenging at present even with the help of deep learning [52]. In most cases, it is still done manually with professional image processing software. We admit that such evasion is possible but at the cost of rendering the photo blurred and thus may greatly increase the difficulty of reading. Thus, we believe that in most cases, in the interest of leaking as much information as possible, adversaries will not bother to remove Moiré patterns.

Display Device. In the aforementioned evaluation, we evaluate the performance of mID with display devices of various manufacturers, models, sizes and panel types. In addition to these factors, the resolution and image rendering mode of display devices may also have impacts on the performance of mID. The dominated resolution of digital screens on the current market is 2 K, and is likely to be increased to 4 K in the future. For mID, resolution enhancement is favorable since it can help smoothen the gratings as a result of decreased distance between two adjacent stripes. For image rendering mode, most users do not change the default settings (with a standard gamma value $\gamma = 2.2$). If by any chance, the users select other rendering modes, e.g., Low Blue Light, Cinema, or Game modes that are available on some mainstream monitors and laptops, the gamma value is likely to be different. However, it will not affect mID because we can obtain the current gamma value of the screen through relevant APIs, and make corresponding adjustments in the luminance correction process.

Capturing Device. Considering convenience and concealment, we assume that smartphones are the most likely capturing devices. However, mID utilizes the interaction between display devices and the CFA of digital cameras. In practice, other digital photographic equipment, e.g., DSLR (Digital Single Lens Reflex) cameras, can also capture the Moiré patterns and thus can cooperate with mID. In addition, compared with smartphones' built-in cameras, they employ less image processing algorithms during photo forming, and thus the Moiré patterns captured are closer to the theoretical superposition results, which may contribute to higher decoding accuracy.

Transmission over Instant-messaging Tools. The adversary may exfiltrate the captured screen photo via instant-messaging tools, e.g., WhatsApp, Skype, and QQ. Image transmission via instant-messaging tools has two forms: (1) The image is transmitted as a file, and (2) The image is transmitted as a photo. The first form is usually (1) lossless (neither the format or size is changed), or (2) format converted (e.g., PNG to JPG). The second form is usually (3) downscaled (compressed). To exfiltrate the confidential information clearly, the adversary is more likely to share the screen photo as a file. In this case, experimental results demonstrate that mID shows no performance difference in decoding 30 screen photos before/after shared as files since mID is robust to format conversion attacks. In a few cases, the adversary may choose to share the screen photo directly as a photo. In this case, the screen photo is downscaled and the EXIF (exchangeable image file) information is lost. Since we encode mID in the horizontal direction and do not rely on any EXIF information, the horizontal downscaling ratio (in the form of pixel numbers) is the main factor that may affect the decoding accuracy. Based on our experiments, the horizontal downscaling ratio depends on the photo contents and the used instant-messaging tools (different tools may use different compression algorithms), and usually ranges from 0.3 to 0.8. With the current encoding parameters shown in Sec. 6.1, mID can still decode screen photos with a horizontal downscaling ratio above 0.6 (i.e., a pixel loss of up to 64%) after shared. For screen photos with smaller horizontal downscaling ratios, the decoding accuracy drops, e.g., by 63.5% for a ratio of 0.5. This can be addressed by adding more redundant pixels, i.e., increasing the value of k, for encoding. Experimental results show that with a larger $k = 8$, the decoding scheme can cope with a horizontal downscaling ratio as low as 0.3. Thus, we assume mID has the potential to survive from the transmission over instant-messaging tools.

Encoding Space. The encoding space of mID mainly depends on the resolution of the display device and the composition of its current page. Specifically, a higher display resolution or a simpler page composition lead to a larger encoding space. An N-bit mID takes $q = 2k \times n \times N$ pixels in width with the capability of identifying 2^{N-4} devices. With a minimal grating height of $p = 50$, for a display device with a resolution of 1920×1080 pixels, the encoding space limit is $2^{[\frac{1920}{2k \times n} - 4] \times [\frac{1080}{p}]} = 2^{1176}$ with our default implementation. We acknowledge that the encoding space cannot reach the limit in practice since only portions of the screen can be used to embed mID. However, we believe that the encoding space of mID is still relatively large and sufficient for screen photo forensics, especially for highly-confidential scenarios.

Shooting Focus, Distance and Angle. mID works well with photos focused on the center of the screen and taken within a distance range of $(60\ cm, 80\ cm)$ and an angle range of $(-20^\circ, 20^\circ)$. We choose these parameters to reflect the goal of adversaries who wish to capture the contents on the screen completely and clearly. Thus, we set the For the shooting focus, we set it on the center of the screen during the experiments considering adversaries' wishes to capture the confidential content completely and clearly. It is okay if the camera is not centrally focused as long as the mID-related Moiré patterns are captured in the photos. For the shooting distance and angle, we agree that beyond the aforementioned ranges may render the generated Moiré patterns out of the visible frequency range, leading to partial or even no Moiré patterns in the captured screen photos. However, we argue that the distances and angles that mID supports can cover most of the possible shooting positions, given the goal of capturing

the contents on the screen completely and clearly.

Comparison with Other Invisible Digital Watermark Techniques. Due to the noises introduced by the electronic screen and the camera sensors, traditional invisible digital watermarks may no longer be recognizable after photographed. Thus, we propose to utilize Moiré patterns for photo forensics since they are optical phenomena generated during the process of photographing screens. We compare our methods with 8 commonly-used invisible digital watermarks including 3 popular commercial tools: (1) SignMyImage [42], (2) Icemark [39], and (3) OpenStego [40], and 5 open source techniques from GitHub [21]: (1) Wavelet Transform, (2) Discrete Wavelet Transform, (3) Discrete Cosine Transform, (4) Least Significant Bit, and (5) Discrete Wavelet Transform and Singular Value Decomposition. The results show that none of the digital watermarks provided by the aforementioned methods work in the screen-photo-based leakage attacks while `mID` can successfully trace to the source of a screen photo with an average accuracy of 96%. Thus, we believe `mID` is suitable for screen photo forensics.

9 Related Work

In this section, we present studies relevant to ours. Specifically, we discuss the aspects related to image watermarking, Moiré pattern, and optical cryptography.

Image watermarking to enable digital media protection. Digital media requires protection when transferring through internet or other mediums. Image watermarking techniques have been developed to fulfill this requirement [38]. Most existing image watermarking approaches are performed in the spatial [1, 7, 29] or DWT (discrete wavelet transform) [12, 20, 23] domains and use frame synchronization methods to resist to geometric distortions. Beyond that, Riad et al. [36, 37] proposed a robust watermarking method based on Discrete Fourier Transform (DFT) for printed and scanned identity images. Gourrame et al. [13] proposed a Fourier based watermarking method to resist print-cam attacks for real captured images and revealed that FFT domain resists better to the perspective distortions compared to the DWT domain. Thongkor and Amornraksa [41] proposed a watermarking method for posters that is robust against distortions due to printing and camera capturing processes. Different from these methods, `mID` is an optical watermark based on Moiré patterns and can be used for screen photo forensics.

Leveraging Moiré patterns to hide invisible messages. Moiré patterns are explored in various studies to hide messages. Lebanon et al. [22] explored ways to superimpose various patterns of gratings to create Moiré patterns of face images. Hersch et al. [15] created moving Moiré components running up and down at different speeds and orientations with the help of a revealing layer. Desmedt et al. [9] created secret sharing schemes based on Moiré patterns with shares being realistically looking images. Tsai et al. [43] enabled the creation of Moiré art and allowed visual decoding by superimposing grating images printed on separate transparencies. Walger and Hersch [45] proposed a method to embed information corresponding to up to seven level-line Moirés within a single base layer, and the information can be recovered later with a revealer printed on a transparency or an array of cylindrical lenses. These studies mainly use two semi-transparent layers and overlap one on the other to reveal hidden images or information. By contrast, `mID` exploits the nonlinear optical interaction of the screen-camera channel to embed identity information.

Optical and visual cryptography to enable secure information exchange. Existing techniques [3, 16, 34] of visual cryptography (VC) usually encode a secret image into several shares with camouflaged visual patterns, and stack a sufficient number of shares to reveal the original secret image. For instance, Huang and Wu [17] proposed an optical watermarking method in which a hidden binary image can be decoded by superposing a transparent key image onto a printed image. These studies [5, 27] applied VC to Quick Response (QR) codes to check the identity accessing to the QR codes or control the permission to the protected data. Inspired by the aforementioned work, `mID` utilizes the inherent attributes of the screen-camera channel and proposes a Moiré-pattern-based optical watermarking scheme to enable screen photo forensics.

10 Conclusion

In this paper, we propose `mID`, a digital forensics mechanism to identify the source of the file leakages via photos utilizing Moiré patterns. We show that Moiré patterns are ideal for photo forensics because they are optical phenomena naturally generated during the process of photographing screens and are observed regularly in photos of digital screens. Leveraging it, we design an effective screen photo forensics scheme, and evaluate it with 5 display devices and 6 smartphones of various manufacturers and models. The evaluation results demonstrate that `mID` can achieve an average BER of 0.6% and an average NER of 4.0%. In addition, the performance is barely affected by the type of display devices, cameras, IDs, and ambient lights. We believe that `mID` is a promising technique and can work complementarily to several existing methods to cope with illegal information leakage. Future directions that worth studying include exploring a wider attack range and further improving the decoding accuracy.

Acknowledgments

We thank our shepherd Apu Kapadia and the anonymous reviewers for their valuable comments. This work is supported by China NSFC Grant 61925109, 61941120, 62071428, and ZJNSF Grant LGG19F020020.

References

[1] Jobin Abraham and Varghese Paul. An imperceptible spatial domain color image watermarking scheme. *Journal of King Saud University-Computer and Information Sciences*, 2016.

[2] Isaac Amidror. *The Theory of the Moiré Phenomenon: Volume I: Periodic Layers*, volume 38. Springer Science & Business Media, 2009.

[3] Carlo Blundo, Alfredo De Santis, and Moni Naor. Visual cryptography for grey level images. *Information Processing Letters*, 75(6):255–259, 2000.

[4] Ajay Kumar Boyat and Brijendra Kumar Joshi. A review paper: Noise models in digital image processing. *arXiv preprint arXiv:1505.03489*, 2015.

[5] Xiaohe Cao, Liuping Feng, Peng Cao, and Jianhua Hu. Secure qr code scheme based on visual cryptography. In *AIIE'16*. Atlantis Press, 2016.

[6] Yizong Cheng. Mean shift, mode seeking, and clustering. *IEEE transactions on pattern analysis and machine intelligence*, 17(8):790–799, 1995.

[7] WN Cheung. Digital image watermarking in spatial and transform domains. In *TENCON 2000 Proceedings: Intelligent Systems and Technologies for the New Millennium*, volume 3, pages 374–378. IEEE, 2000.

[8] Harold Scott Macdonald Coxeter, Harold Scott Macdonald Coxeter, Harold Scott Macdonald Coxeter, and Harold Scott Macdonald Coxeter. *Introduction to geometry*, volume 136. Wiley New York, 1969.

[9] Yvo Desmedt and Tri Van Le. Moiré cryptography. In *CCS'00*, pages 116–124. ACM, 2000.

[10] Richard M Dudley and Richard M Dudley. *Uniform central limit theorems*. Number 63. Cambridge university press, 1999.

[11] Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.

[12] Emir Ganic and Ahmet M Eskicioglu. Robust dwt-svd domain image watermarking: embedding data in all frequencies. In *MM&Sec'04*, pages 166–174. ACM, 2004.

[13] Khadija Gourrame, Hassan Douzi, Rachid Harba, Frederic Ros, Mohamed El Hajji, Rabia Riad, and Meina Amar. Robust print-cam image watermarking in fourier domain. In *ICISP'16*, pages 356–365. Springer, 2016.

[14] Ankush Gupta, Andrea Vedaldi, and Andrew Zisserman. Synthetic data for text localisation in natural images. In *CVPR'16*, pages 2315–2324, 2016.

[15] Roger David Hersch and Sylvain Chosson. Band moiré images. *ACM Transactions on Graphics (TOG)*, 23(3):239–247, 2004.

[16] Young-Chang Hou. Visual cryptography for color images. *Pattern recognition*, 36(7):1619–1629, 2003.

[17] Sheng Huang and Jian Kang Wu. Optical watermarking for printed document authentication. *IEEE Transactions on Information Forensics and Security*, 2(2):164–173, 2007.

[18] Cambridge in Colour. *Digital Image Interpolation*, 2019. https://tinyurl.com/twxaxjk.

[19] The Khronos Group Inc. *OpenGL - The Industry's Foundation for High Performance Graphics*, 2019. https://www.opengl.org/.

[20] Chih-Chin Lai and Cheng-Chih Tsai. Digital image watermarking using discrete wavelet transform and singular value decomposition. *IEEE Transactions on instrumentation and measurement*, 59(11):3060–3063, 2010.

[21] lakshitadodeja. *image_watermarking*, 2017. https://github.com/lakshitadodeja/image_watermarking.

[22] Guy Lebanon and Alfred M Bruckstein. Variational approach to moiré pattern synthesis. *Journal of the Optical Society of America A*, 18(6):1371–1382, 2001.

[23] Qiang Li, Chun Yuan, and Yu-Zhuo Zhong. Adaptive dwt-svd domain image watermarking using human visual model. In *ICACT'07*, volume 3, pages 1947–1951. IEEE, 2007.

[24] Minghui Liao, Baoguang Shi, Xiang Bai, Xinggang Wang, and Wenyu Liu. Textboxes: A fast text detector with a single deep neural network. In *AAAI'17*, 2017.

[25] Ming-Jiun Liaw, Ho-Hsin Yang, and Yuh-Ren Shen. Automatic gamma correction system for displays, 2003. US Patent 6,593,934.

[26] PricewaterhouseCoopers LLP. *Study on the Scale and Impact of Industrial Espionage and Theft of Trade Secrets through Cyber*, 2019. https://tinyurl.com/ro5qu6o.

[27] Jianfeng Lu, Zaorang Yang, Lina Li, Wenqiang Yuan, Li Li, and Chin-Chen Chang. Multiple schemes for mobile payment authentication using qr code and visual cryptography. *Mobile Information Systems*, 2017, 2017.

[28] Jack Morse. *Leaking anonymously is hard. Here's how to do it right, and not get caught*, 2017. https://tinyurl.com/wplyk38.

[29] Nikos Nikolaidis and Ioannis Pitas. Robust image watermarking in the spatial domain. *Signal processing*, 66(3):385–403, 1998.

[30] Hao Pan, Yi-Chao Chen, Guangtao Xue, Chuang-Wen Bing You, and Xiaoyu Ji. Secure qr code scheme using nonlinearity of spatial frequency. In *UbiComp'18*, pages 207–210. ACM, 2018.

[31] Tutorials Point. *Digital Communication - Line Codes*, 2019. https://tinyurl.com/yx6gxblv.

[32] C. Poynton. *Digital Video and HD: Algorithms and Interfaces*. Electronics & Electrical. Elsevier Science, 2003.

[33] Charles A Poynton. Smpte tutorial:"gamma" and its disguises: The nonlinear mappings of intensity in perception, crts, film, and video. *SMPTE journal*, 102(12):1099–1108, 1993.

[34] P Punithavathi and S Geetha. Visual cryptography: A brief survey. *Information Security Journal: A Global Perspective*, 26(6):305–317, 2017.

[35] Abigail Raney. *Pinhole Camera Theory Summary*, 2017. https://tinyurl.com/s5bpf9h.

[36] Rabia Riad, Rachid Harba, Hassan Douzi, Mohamed El-hajji, and Frédéric Ros. Print-and-scan counterattacks for plastic card supports fourier watermarking. In *ISIE'14*, pages 1036–1041. IEEE, 2014.

[37] Rabia Riad, Frédéric Ros, Rachid Harba, Hassan Douzi, and Mohamed El Hajji. Pre-processing the cover image before embedding improves the watermark detection rate. In *WCCS'14*, pages 705–709. IEEE, 2014.

[38] Lalit Kumar Saini and Vishal Shrivastava. A survey of digital watermarking techniques and its applications. *arXiv preprint arXiv:1407.4735*, 2014.

[39] Phibit Software. *Icemark*, 2016. http://www.phibit.com/icemark/.

[40] syvaidya. *OpenStego*, 2015. https://sourceforge.net/projects/openstego/.

[41] Kharittha Thongkor and Thumrongrat Amornraksa. Robust image watermarking for camera-captured image using image registration technique. In *ISCIT'14*, pages 479–483. IEEE, 2014.

[42] Advanced Photo Tools. *SignMyImage*, 2013. http://www.adptools.com/signmyimage/.

[43] Pei-Hen Tsai and Yung-Yu Chuang. Target-driven moire pattern synthesis by phase modulation. In *ICCV'03*, pages 1912–1919, 2013.

[44] Version. *2018 Data Breach Investigations Report*, 2018. https://tinyurl.com/qm3dmm2.

[45] Thomas Walger and Roger David Hersch. Hiding information in multiple level-line moirés. In *DocEng'15*, pages 21–24. ACM, 2015.

[46] Wikipedia. *Color filter array*, 2019. https://en.wikipedia.org/wiki/Color_filter_array.

[47] Wikipedia. *Gamma correction*, 2019. https://en.wikipedia.org/wiki/Gamma_correction.

[48] Wikipedia. *Grayscale*, 2019. https://en.wikipedia.org/wiki/Grayscale.

[49] Wikipedia. *HSL and HSV*, 2019. https://en.wikipedia.org/wiki/HSL_and_HSV.

[50] Wikipedia. *k-means clustering*, 2019. https://en.wikipedia.org/wiki/K-means_clustering.

[51] Wikipedia. *Visual system*, 2019. https://en.wikipedia.org/wiki/Visual_system.

[52] Shanxin Yuan, Radu Timofte, Ales Leonardis, Gregory Slabaugh, Xiaotong Luo, Jiangtao Zhang, Yanyun Qu, Ming Hong, Yuan Xie, Cuihua Li, et al. Ntire 2020 challenge on image demoireing: Methods and results. *arXiv preprint arXiv:2005.03155*, 2020.

[53] Kai Zhang, Chenshu Wu, Chaofan Yang, Yi Zhao, Kehong Huang, Chunyi Peng, Yunhao Liu, and Zheng Yang. Chromacode: A fully imperceptible screen-camera communication system. In *MobiCom'18*, pages 575–590. ACM, 2018.

[54] zlyBear. *BearOCR*, 2019. https://github.com/zlyBear/BearOCR.

11 Appendix

11.1 Minimal Photograph Distance

Considering the goal of recording the confidential information displayed on the screen with smartphones, we assume the adversary is likely to capture the screen in complete and hold the smartphone vertically to avoid the signs of secret filming. In this case, the photograph distance D adopted by the adversary shall be larger than a minimal value D_{min} to contain the entire screen in photos.

According to Equ. 4, the photograph distance D can be calculated as $D = \frac{S_{obj} \times L_f}{S_{cam}}$. For the minimal distance D_{min}, S_{obj} is the physical width of the display screen, L_f is the physical focal length of the camera, and S_{cam} refers to the image width of the camera. S_{cam} can be further calculated as $S_{cam} = S_p \times N_p$, where S_p is the size of a single pixel and N_p is the number of pixels in width of the camera. As a result, D_{min} can be given as follows:

$$D_{min} = \frac{S_{obj} \times L_f}{S_{cam}} = \frac{S_{obj} \times L_f}{S_p \times N_p} \tag{15}$$

With the device specifications in Tab. 1 and Tab. 2, we can calculate the minimal photograph distance D_{min} for various screen-camera settings. For instance, for our default setting, i.e., the BenQ EW2440ZC monitor for image display and the LG Nexus 5X smartphone for image capture, $D_{min} = \frac{S_{obj} \times L_f}{S_p \times N_p} = \frac{53.1cm \times 5mm}{1.55\mu m \times 3024} = 56.6cm$. For HUAWEI Mate 10, HUAWEI P9, and Apple iPhone X, it will be $57.1cm$, $64.2cm$, and $57.6cm$, respectively. Therefore, to capture a 24" LCD display that is most commonly seen on the market with smartphones, the photograph distance shall usually be larger than 60 cm.

11.2 `mID` Algorithms

Algorithm 1: Saturation Balance

Input:
- $M = \{H,S,V\}$: extracted joint Moiré area
- N: number of bits of `mID`.
- w: width of the joint Moiré area

Output: S': saturation of the balanced joint Moiré area

$sp_2, sp_N \leftarrow$ SPLIT_MOIRÉ _AREA(M,N) $s_l, s_r \leftarrow$ AVERAGE_SATURATION(sp_2, sp_N)
$p_l = 0.5 + 0.5 * abs(min(0, \frac{s_l - s_r}{s_r}))$
$p_r = 0.5 + 0.5 * abs(min(0, \frac{s_r - s_l}{s_l}))$
for $M(x,y) \in M$ **do**
  if $y \leq w/2$: **then**
    $a = \frac{1}{max(S(x,y), 1-p_l)}$
  else
    $a = \frac{1}{max(S(x,y), 1-p_r)}$
  $S'(x,y) = a \cdot S(x,y)$

Algorithm 2: `mID` Decoding

Input:
- $M = \{H,S,V\}$: extracted joint Moiré area
- N: number of bits of `mID`.

Output: B: decoded bit sequence

$W \leftarrow$ WIDTH(M) // get the width of the joint Moiré area
$S' \leftarrow$ SATURATION_BALANCE(M,N,W) // get the saturation of the balanced joint Moiré area
$\mu \leftarrow$ AVERAGE(S')
$\sigma \leftarrow$ STD(S')
$\alpha = 1.6$ // amplification factor
for $S'(x,y) \in S'$ **do**
  if $S'(x,y) \leq \mu + \alpha \cdot \sigma$: **then**
    $S'(x,y) = 0$
$S'(y) = normalization(\sum_x S'(x,y))$ // get the normalized saturation matrix
$S'(y) = hanning(S'(y))$ // noise suppress
for $i \in [0, N-1]$ **do**
  $P_i = \sum S'(\frac{w}{N} \cdot i : \frac{w}{N} \cdot (i+1))$
$P_k \leftarrow$ K_LARGEST(P,k) // get the k^{th} largest value
$\sigma' \leftarrow$ STD(P)
$\beta = 0.1$ // the decreasing factor
for $i \in [0, N-1]$ **do**
  if $P_i \geq P_k$: **then**
    $P_i = \frac{P_i}{\sigma'^{\beta}}$
$B \leftarrow$ K-MEANS(P) // k-means clustering
$f \leftarrow$ CHECK_CODE_MATCHING(B)
if $f ==$ *True:* **then**
  $B = \sim B$

11.3 Summary of Questionnaire Survey

The first 3 questions of the questionnaire survey are essay-content-related choice questions, which are the superficial tasks of the test. The real aim is to learn whether the participants feel or notice any visual abnormality during the process of reading (Question 6). Two transitional questions (Question 4 and 5) are used to cover it up.

Table 4: Summary of questionnaire survey.

No.	Question
1-3	Essay-content-related choice questions
4	Is this test difficult? (7-point scale, where 7 indicates the most difficult)
5	Did the display device work well? (7-point scale, where 7 indicates the best functionality)
6	Do you feel abnormal or uncomfortable during reading, e.g., display glitch/flicker or visual abnormity? (7-point scale, where 7 indicates the most abnormal)