RF-Eye-D: Probing Feasibility of CMOS Camera Watermarking with Radio-Frequency Injection

Hui Zhuang, Yan Long†, Kevin Fu
Northeastern University, Boston, USA
{zhuang.hu, y.long, k.fu}@northeastern.edu

***Abstract*—This work explores how to physically watermark images generated by CMOS cameras using deliberately injected radio-frequency signals. CMOS camera imaging is ubiquitous in embedded systems such as smartphones, AR/VR headsets, drones, and other IoT platforms to capture photos and videos. In restricted environments, a property owner may wish to prevent unauthorized camera recordings depending on spatio-temporal context. Indelible watermarks can deter unauthorized recording. A key research challenge is how to find a reasonably general mechanism to surreptitiously inject watermarks without access to the camera. Existing methods typically rely on software-based watermarking or metadata generation, assuming cooperation from camera owners. However, adversaries can trivially disable metadata or watermarking functions to evade forensic analysis. To address this gap, our work explores an unconventional approach of watermarking non-cooperative cameras by injecting radio-frequency interference in the environment to affect the analog sensing process and inject defender-controlled patterns in the image output. Our analysis explains how the rolling shutter and Bayer filter hardware convert radio-frequency signals into color stripes with variable widths. Building upon model-based simulation, our prototype design encodes and extracts imperceptible watermarks with a bandwidth of up to 50 bits per image. Proof-of-concept evaluations in lab environments show that the proposed technique could support watermarking images with diverse background scenes and reveal future challenges of improving watermark bandwidth and injection distance.**

I. Introduction

Our research probes the feasibility of externally injecting watermarking information into images taken by CMOS cameras sensors by generating intentional radio-frequency (RF) interference in the physical environment. Digital images produced by camera sensors have become one of the most common types of high-entropy data for sharing information. It is estimated more than 1.5 trillion images will be generated per year after 2022 by existing smartphones, IoT devices, etc., as well as emerging AR/VR headsets [1]. Given that many of the images generated by cameras could be taken in unauthorized locations or times, the ability to add geolocation and timestamp information to images is key to supporting image forensics and preventing malicious creation, manipulation, and distribution of camera images [2], [3].

While various image watermarking and metadata generation techniques exist, they rely on the key assumption that the camera software and hardware generating the photo are cooperative with the defender party who wants to embed watermarking information [4]–[7]. However, this assumption does not apply to many cases of unauthorized photography in the age of ubiquitous camera sensors in IoT and smart devices. For example, when a malicious party walks into a room and takes a photo using a smartphone secretly, they could have full control over their camera software and thus disable all metadata and watermarking functions on the camera device. This problem thus calls for a method to *externally watermark an image with defender-created information, even when the defender does not have control over the adversary's camera devices.*

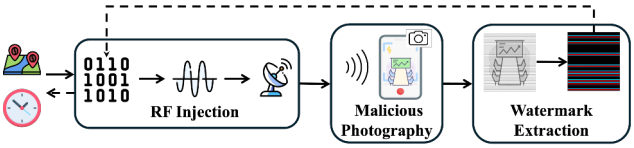


Fig. 1: To deter and trace malicious camera photography in prohibited locations and times, `RF-Eye-D` uses external RF signals to inject imperceptible stripe-like watermarks into CMOS camera images.

To bridge this gap, our work carries out the first investigation of how to watermark photos taken by non-cooperative CMOS cameras using RF injections. Our physical watermarking method builds upon our key observations that intentional RF interference in the ambient environments of camera hardware can induce voltage perturbations to the underlying analog signal readout circuits of CMOS camera sensors, thus creating defender-controlled information channels in the output images, as illustrated in Fig. 1. Furthermore, our analysis shows that this watermark injection process is bounded by the row-wise parallel readout architecture and the Bayer color filter hardware, creating stripe watermarking patterns whose width, number, and color can be predicted. Our extensive experimental analysis allows us to build a theoretical model for describing the injected patterns in the images.

However, turning these induced stripe patterns into useful watermarking primitives faces unique challenges. First, the injected RF signals cannot be synchronized with the internal timing of the CMOS sensing circuits because the adversary-controlled camera hardware is assumed not to be cooperative. As a result, the injected patterns could appear at arbitrary locations in the output images, and thus demand a synchronization-free watermark encoding/decoding scheme. Second, the actual

†Corresponding author

photos taken by cameras in real-world environments could have complex scene information, which is essentially noise to the defender who wants to recognize watermarks embedded in the images. While prior research has shown the feasibility of extracting similar stripe patterns induced by magnetic signals from dark images taken by blocked camera sensors [8], the requirement of simple dark-current images makes such techniques infeasible for watermarking purposes. The unique requirement of making watermarks imperceptible to human perception further adds to the challenge because imperceptibility means even lower signal-to-noise ratios that the watermark recognition system needs to operate on.

This research explores the solutions to these challenges through the design of `RF-Eye-D`, a model-based system for encoding and extraction of RF-induced watermarks in CMOS camera images. The system consists of a watermark enhancement and extraction frontend that utilizes color space transformations informed by the sensor's de-bayering process to amplify the difference between the watermark and the background scenes, and then employs a U-Net network well-suited for image segmentation tasks to separate the clean stripes from the background scenes. A watermark decoding backend uses bit region segmentation and bit value determination algorithms to convert cleaned stripes back to digital bits and then recursively searches for watermarks by comparing the Hamming distances of decoded bits to known preambles. Notably, the neural network-powered system could be trained exclusively on simulated image data generated by our RF watermarking model that describes how different frequencies of RF energy are transformed into imperceptible watermarks overlaid on ordinary images. Our proof-of-concept evaluations provide demonstrations of the RF-based watermarking technique's potential in various conditions in controlled lab environments. Tested factors include the textural complexity of camera scenes, lighting conditions, camera angles, the number of bits injected per image that can convey different typical types of geotagging and timestamping information, and the model of camera sensor hardware. Based on the observed limits, we further discuss possible future improvements that may take the proof-of-concept design to real-world deployment. In summary, the main contributions of this work include:

- The experimental methodology and theoretical models for using RF signals to physically inject watermarks into CMOS camera images. They lay the foundation of a novel paradigm of image watermarking that do not require work with non-cooperative camera devices.
- The pilot system design of RF watermark encoding, injection, and extraction. The system consists of hardware and software designs that can be reused in diverse physical camera watermarking scenarios.
- Proof-of-concept evaluations in lab environments that characterize the critical factors affecting the watermarking capabilities. The evaluations reveal key research questions that are worth further investigation.

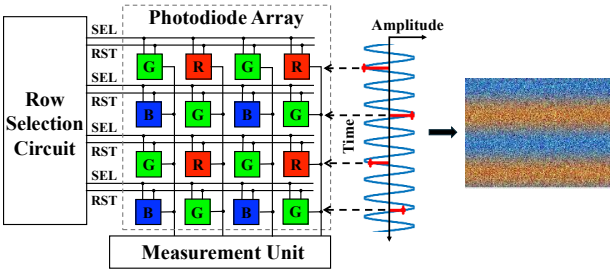


Fig. 2: The CMOS camera sensor hardware structure consisting of the Bayer color filter and the rolling shutter row-wise scanning control and measurement units. RF energy injected during the analog signal readout of pixel rows could induce colored, imperceptible stripe patterns for image watermarking.

II. Background

This section provides background information on CMOS camera sensing and RF and other electromagnetic interference's known impacts on sensor hardware to explain the motivation and preliminaries of `RF-Eye-D`.

A. CMOS Camera Sensor Hardware

Camera sensors are mainly categorized into Complementary Metal-Oxide-Semiconductor (CMOS) and Charge-Coupled Device (CCD) types. CCD sensors use a global shutter that transfers charges from all pixels to a centralized readout unit, but suffer from slower speeds and higher costs. CMOS sensors use a rolling shutter for row-wise exposure and readout, offering greater efficiency and lower cost, which makes them dominant in smartphones and other consumer-grade imaging systems [9]–[11]. A typical CMOS camera comprises a photodiode array and downstream scanning and measurement circuits.

Photodiode Array and Bayer Filter: Photodiodes are used to transduce incoming photons, converting them into electrical charge signals. The stronger the incoming light, the more signal charges are generated, resulting in a higher pixel value. Each photodiode captures the intensity of only one color channel (one of RGB), and the arrangement of these color channels follows the Bayer pattern [12], as shown in Fig. 2. After signal charges have accumulated for a certain period of time, i.e., the exposure time, the electrical signals are read out by the scanning unit.

Rolling Shutter Scanning: The scanning unit consists of row control logic and multiple shift registers. The control logic sequentially selects pixel rows in a predefined order, thereby enabling the row-by-row exposure process of the rolling shutter mechanism, as shown in Fig. 2. The analog voltage of each pixel in the activated row is then transmitted through the column bus to the corresponding measurement unit, where it is simultaneously sampled and digitized by column-parallel measurement units.

Measurement Unit: A measurement unit consists of an amplifier and an analog-to-digital converter (ADC). CMOS

camera sensors typically employ a column-parallel ADC architecture, in which each column has its own ADC. During image readout, the analog signals on all columns of the selected row are simultaneously transferred to the corresponding column ADC and then sampled. Since all column ADCs operate synchronously within the same row, injecting RF energy coupled into the analog circuitry and sampled by the ADCs during the sampling phase could only cause row-wise variation artifacts in the captured image, as will be shown in Section III.

Signal Sampling and Aliasing. When ADCs sample continuous signals with insufficient sampling rates, aliasing distortions could happen in the output digital sequences. A digital signal with a sampling rate of f_s has a bandwidth $f_s/2$, meaning that the continuous signals with frequencies lower than $f_s/2$ can be converted into digital signals without loss of information. Otherwise, aliasing could convert a signal into other unseen frequencies [13], causing the observed low-frequency stripes induced in images by high-frequency RF signals in Section III.

B. Image Processing Output under Interference

Since each pixel in the photodiode array captures only one color channel, the initial output is a raw Bayer-format image [14] that requires demosaicing to reconstruct a full RGB image. Demosaicing algorithms utilize the spatial arrangement of the Bayer matrix and apply appropriate interpolation methods, such as nearest-neighbor interpolation, to estimate the missing values of the other two color channels for each pixel based on the values of neighboring pixels.

The row-wise readout architecture of camera sensors stimulates the assumption that any external RF signals could only cause row-wise variations in the images because the injected RF energy could only cause all pixel values within the affected row to increase or decrease consistently, depending on the strength and polarity of the coupled signal. Given that each row of the Bayer color filter array contains only two types of color channels, we hypothesize that such disproportionate row-wise disturbances can disrupt inter-channel balance and lead to black-and-white or chromatic stripe artifacts in the demosaicing process.

C. Electromagnetic Interference in Sensor Hardware

Electromagnetic interference (EMI) refers to the phenomenon where electromagnetic waves, including RF and other ranges of frequencies, disrupt the normal operation of nearby electronic devices [15], [16]. Prior studies have demonstrated that electromagnetic signals can affect the reading of a wide range of sensors such as microphones [17], temperature sensors [18], [19], lidar [20], keyboards [21], touch sensors [22], etc. These sensors are vulnerable to EMI because they rely on electrical signals to convert physical inputs into digital data, and the electrical signals can be changed by external electromagnetic energy when the energy couples into the target systems through pervasive electrical traces such as metal interconnects on the sensor hardware.

Prior research has demonstrated the potential for EMI to change the image output of camera sensors. [11], [23] showed how to inject electromagnetic signals into CCD cameras to modify specific pixel values. However, due to the architectural differences between CCD and CMOS sensors, these findings cannot be directly applied to CMOS cameras. Notably, CMOS cameras employ a rolling shutter mechanism, which limits EMI from affecting image content on a row-by-row basis. Moreover, while these works primarily focused on inducing noticeable interference in images, our goal is to embed imperceptible watermarks and achieve reliable decoding, which is inherently challenging. On another front, [9] demonstrated a method for generating apparent row-wise purple stripes in CMOS camera images by causing bit losses of the digital camera data transmissions. However, this approach has two major limitations that prevent it from being used for watermarking. First, visible stripes only appear when an odd number of rows are lost. If the number of lost rows is even, no visual stripes are produced, significantly undermining its reliability for watermark purposes. Second, the intensity of the induced purple stripes, which essentially compromises the usability of the produced images and makes them easily detectable by malicious photography adversaries. These gaps require us to investigate RF injection methods that can create imperceptible and reliable watermarks.

III. RF-based Image Watermarking

A. Threat & System Model

The defensive watermarking technology of `RF-Eye-D` investigates how to associate a photo with a specific physical space to allow a defender such as the owner of the physical space to assert that the photo was captured in this space at a specific time. Instead of relying on software watermarks, we explore how to enable the environment to physically inject information into the images using RF signals.

Adversary. We consider an adversary that takes a photo in a location and/or at a time that the adversary does not want to reveal, such as in a photography-prohibited room. The adversary may subsequently distribute the photos over media platforms or other communication channels. The goal of the adversary is to prevent the defender party from asserting where and/or when the photo was taken. The adversary could attempt to achieve this by turning off the camera's location service and erasing the photo's metadata containing the software timestamp. We also assume the adversary, as the owner of the camera device, can disable any other software watermarking functionalities on their camera device.

Defender. The defender's goal is to inject imperceptible geotagging watermarks that can associate the photo with a specific physical environment. The location and time are two representative characteristics of the physical environment that the defender wants to identify through the injected watermarks. The defender's system consists of an RF emitter located in the physical space that is able to inject RF energy into the sensing circuits of the adversary's CMOS camera device and a suite

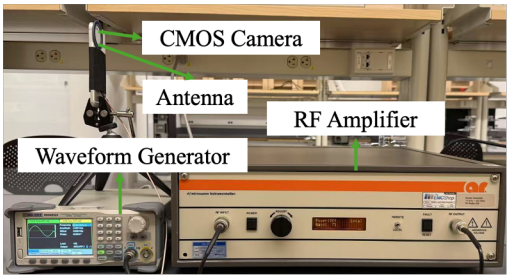
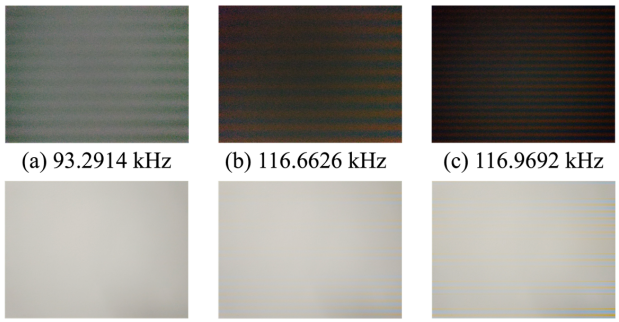


Fig. 3: Our experimental setup for feasibility tests.



(a) 93.2914 kHz (b) 116.6626 kHz (c) 116.9692 kHz

(d) No Amplification (e) With Amplification (Gain: 0%) (f) With Amplification (Gain: 50%)

Fig. 4: Examples of RF-induced stripe patterns under different RF frequencies and signal strengths.

of algorithms that can extract and identify the watermarks in photos distributed by the adversary.

Application Scenarios. The capability provided by this work enables an array of defensive applications, such as protecting against the threat of unauthorized photography in restricted areas and tracing the source of illegal images. In scenarios involving sensitive information, such as scientific laboratories, government agencies, or corporate meetings, photographing or recording videos could be prohibited to prevent the leakage of confidential information. However, malicious individuals may still secretly capture images and upload them to social media or share them via instant messaging applications, causing incidents of information leakage [24]–[26]. If specific physical environment information can be embedded during the imaging process of malicious photography, social media platforms could detect these embedded watermarks and automatically block sensitive content before image publication. Even if an adversary is equipped with an RF injector and attempts to interfere with the watermarking process, the injected signal primarily introduces additional stripe artifacts whose structural patterns remain detectable, allowing the system to identify the presence of a watermark and flag the image as suspicious. Such embedded watermarks can further be used in courtrooms to provide forensic evidence identifying the origin of the leaked content. Additionally, the traceability offered by such RF-induced watermarks enhances accountability in environments where traditional surveillance may be limited or privacy-constrained.

B. Feasibility Test

To verify our hypothesis that external RF signals are capable of interfering with the CMOS camera sensing process to inject watermark information, we conducted feasibility tests using the experimental setup in Fig. 3. The setup consists of a waveform generator (Siglent SDG6052X), an RF amplifier (Amplifier Research 25A250B), and a near-field probe antenna. The target camera sensor is a SONY IMX 378 CMOS sensor commonly used in smartphones and IoT camera devices. We operated the waveform generator in frequency sweep mode with a sine wave output to measure how the camera sensor could respond to different RF frequencies. The antenna was positioned in close proximity to the CMOS camera sensor to ensure strong controllable RF coupling.

We observed during the frequency sweep test that the captured images exhibited additional row-wise stripe patterns at certain frequencies. In addition, different RF frequencies could change both the color and number of induced stripes, as shown in Fig. 4. This frequency-dependent variation confirms that the stripe patterns result directly from RF signals. When increasing or decreasing the strength of the RF output, the intensity of the stripes becomes higher or lower accordingly. Further reducing the RF power makes the induced stripe patterns imperceptible to human eyes. These phenomena are fundamentally different from the induced patterns discovered in previous work [9], where EMI disturbing the digital data transmission of pixel data could only induce obvious purple stripes that have a constant, strong intensity. A key difference is the use of MHz and GHz electromagnetic frequencies in [9] that tend to affect digital transmission. In contrast, we utilize RF frequencies on the order of 100 kHz to adapt to the operation of the rolling shutter process. This further confirms that our new RF injection method affects the analog sensing process of CMOS camera sensors. Moreover, we observed that the variations in the number and color of the stripes exhibited periodic changes. Similar stripe distributions in terms of number and color consistently appeared at regular frequency intervals. This suggests that certain forms of sampling process during CMOS image acquisition have caused aliasing of the injected RF signal, which prompted us to further model the underlying causality of the injected patterns (Section III-C).

In addition to the impact of RF output, the injected patterns are also affected by the imaged scene content. The stripes were most prominent when the camera lens was physically covered, as the low light intensity caused the amplifier in the CMOS sensor's measurement unit to adaptively increase the gain, thereby amplifying image signals and making the stripes more visible. When the camera imaged a normal background scene, however, the visibility of the stripes could be significantly reduced and become imperceptible to human eyes. To verify whether the stripes still remain detectable in such imperceptible cases, we captured two images of the same scene, one with

(a) Clean (b) Image with Stripes (c) Difference

Fig. 5: Comparison between (a) a camera-captured image scene, and (b) the scene captured when there is RF injection. Although complex image scenes make RF-induced watermarks imperceptible to human eyes, the difference (c) shows it's computationally detectable.

RF signal injection enabled and the other without. We then computed the pixel-wise difference between the two images and amplified the result to reveal subtle variations. As shown in Fig. 5, the difference images clearly display the presence of stripe patterns. These observations provide evidence that RF interference can be used to embed imperceptible watermark information into camera-captured content, which aligns with the requirements of watermarking applications.

C. Causality Model

The feasibility tests have demonstrated how changing RF injection frequencies could potentially control the number and color of injected stripe patterns for watermarking. The defender injecting the stripes may thus utilize this characteristic to build a controlled information channel between the defender and the adversary's camera image outputs. Achieving precise control requires a deep understanding of how RF parameters are mapped to different stripes. This section thus seeks to explain the relevant observations and provide a theoretical model underpinning the defender-controlled watermark injection process.

Number of Injected Stripes. The change of stripe number is equivalent to changing the width of the injected stripes. Fig. 6 shows that the number of stripes exhibits a periodic variation pattern when RF frequency changes. Within each frequency cycle, the stripe count follows a repeating "decrease–increase" pattern. The minimum stripe count appears at a center RF frequency (116.3560 kHz), where the image displays alternating blue and orange bands covering the entire image. Moreover, across different frequency cycles, such as those centered at 116.3560 kHz and 209.4408 kHz, the stripe count and variation pattern remain consistent under identical frequency offsets from the center frequency.

This behavior stems from the fact that the ADCs in the measurement unit sample the rows of the pixel array at a certain sampling rate. As explained in II-A, aliasing occurs when the sampling rate is lower than twice the injected signal frequency, leading to periodic changes in the appearance and distribution of stripe patterns. The process is modeled below. Let f_{signal} and f_{sample} denote the frequency of the injected RF signal and the ADC sample frequency of the CMOS sensor

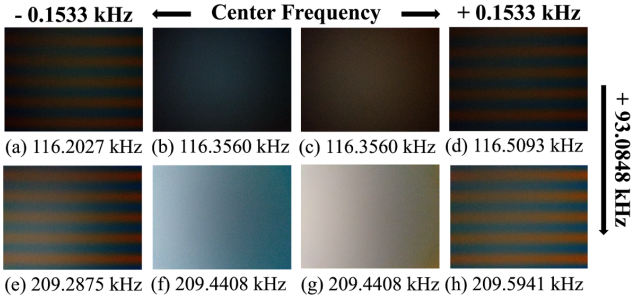


Fig. 6: Stripe number variations under different RF frequencies. The first row (a–d) is centered around 116.3560 kHz. At the center frequency (b and c), the induced stripe number reduced to less than 1. When the frequency slightly deviates from the center (a and d), the number of visible stripes increases symmetrically. The second row (e–h) follows the same patterns with another center frequency at 209.4408 kHz.

respectively. The resulting aliasing frequency observed during the row-by-row sampling phase can be expressed as:

$$f_{\text{alias}} = f_{\text{signal}} - N \times f_{\text{sample}}, \tag{1}$$

where $N \in \mathbb{N}$ and $-\frac{f_{\text{readout}}}{2} \leq (f_{\text{signal}} - N_1 \cdot f_{\text{s}}) \leq \frac{f_{\text{sample}}}{2}$. The stripe number is determined by the envelope of the aliased signal, which exhibits different behaviors across various frequency ranges:

$$f_{\text{env}} = \begin{cases} |f_{\text{alias}}|, & \text{if } |f_{\text{alias}}| \leq \frac{f_{\text{sample}}}{4} \\ \left| f_{\text{alias}} - \frac{f_{\text{s}}}{4} \right|, & \text{if } |f_{\text{alias}}| > \frac{f_{\text{sample}}}{4} \end{cases} \tag{2}$$

Accordingly, the number of stripes follows:

$$num\_stripes = \frac{W \times f_{\text{env}}}{f_{\text{sample}}} \tag{3}$$

where W denotes the total number of image rows. For the SONY IMX378 CMOS image sensor, the ADC sampling frequency is measured to be $f_{\text{sample}} = 93.0848$ kHz, and the number of rows is $W = 3036$ pixels. The number of stripes in Fig. 4 (b) and (c) is 10 and 20, respectively, showing how the actual number of injected stripes matches with this theoretical model's output.

Color of Injected Stripes. Varying frequencies of the injected RF signal also result in different stripe colors, as shown in Fig. 4. We observe that the possible outcomes can be categorized into black-and-white stripes (Fig. 4 (a)) and colored stripes (Fig. 4 (b)). The stripe color is jointly influenced by the f_{alias} and the structure of the Bayer filter matrix. In a typical Bayer matrix, adjacent rows consist of RG and GB channels. Since CMOS image sensors perform row-wise readout, a very small f_{alias} leads to sampled signals in adjacent pixel rows having highly similar amplitudes and phases. This causes the RGB channels in neighboring rows to increase or decrease simultaneously, resulting in black-and-white alternating stripes. However, the changes in channel values across adjacent rows become uneven when the amplitude

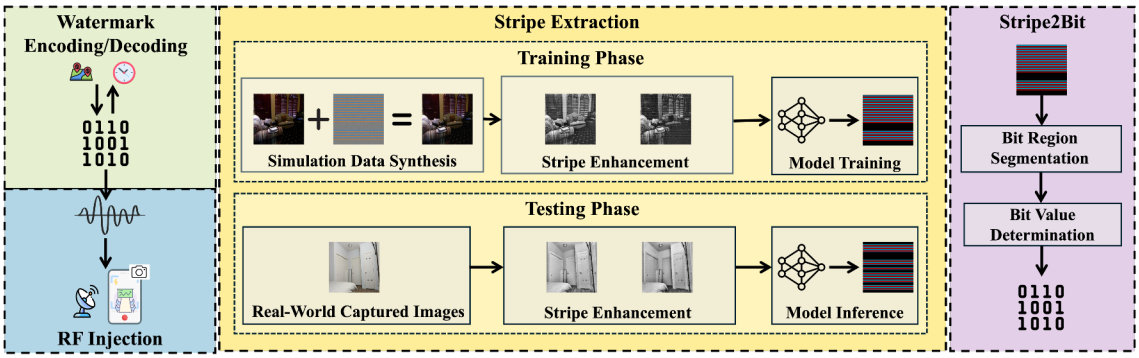


Fig. 7: Overview of the RF-Eye-D prototype system.

and phase differences between signals sampled by adjacent rows become slightly larger, resulting in colored stripes. The knowledge of stripe color enables us to perform color space transformations for stripe pattern enhancement, as will be shown in Section IV-C1.

IV. RF-Eye-D System Design

Fig. 7 provides an overview of our prototype system achieving CMOS camera image watermarking. It consists of four modules: Watermark Encoding/Decoding, RF Injection, Stripe Extraction, Stripe2Bit.

A. Watermark Encoding

Embedding watermark information into a CMOS camera through RF signal injection presents several challenges. The most critical issue lies in the lack of an effective synchronization mechanism between the injected RF signal and the image acquisition process, making it hard to determine the starting position of the embedded watermark in the decoding stage. Furthermore, unlike optical camera communication or electromagnetic signal injection-based camera communication methods [8] [27] that assume a cooperative camera can thus leverage multiple video frames to progressively accumulate information, non-cooperative camera watermarking requires the entire watermark to be embedded within a single image frame. This imposes stricter constraints on RF watermark design.

To resolve these challenges, our proposed encoding scheme first converts the watermark information into a binary bit sequence and place a specific preamble at the beginning of the sequence to enable synchronization during decoding. Using on–off keying, we map each bit to a signal pattern: a bit value of "1" triggers the emission of an RF signal at a designated frequency, whereas a bit value of "0" corresponds to no signal transmission. Subsequently, a time-domain signal is synthesized based on the bit sequence, with a total duration equal to the image capture time t_S (defined in Equation 5). The duration of each bit, denoted as t_B, is defined by:

$$t_B = \frac{t_S}{num\_bits} \tag{4}$$

where num_bits represents the total number of bits, including the preamble. This modulated signal is then transmitted in a continuous loop. Given that the exact temporal alignment between the RF signal and the beginning of image acquisition cannot be guaranteed, we can, without loss of generalizability, assume that the RF signal starts to affect the image from row n. Thus, the portion of the image from row n to row W captures $\left(\frac{W-n+1}{W}\right) \times num\_bits$ bits approximately, starting from the first bit in the current signal cycle; the rows from 1 to $n-1$ correspond to the tail end of the previous cycle and cover the bit indices ranging from $\left(\frac{W-n+2}{W}\right) \times num\_bits$ to num_bits.

Once the preamble is successfully located in the reconstructed bit sequence during the decoding process, the system can achieve proper alignment and then recover the complete watermark sequence. In cases where the detected bit sequence does not contain an exact match to the predefined preamble due to the presence of bit transmission and detection errors, the system computes the Hamming distance between the known preamble and all candidate subsequences within the detected sequence and select the one with the minimum distance as the best match.

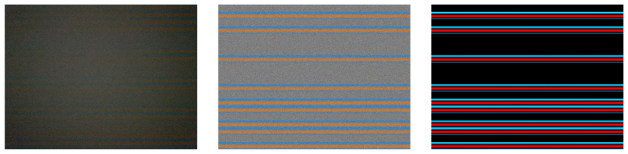
B. Stripe Injection

RF-Eye-D carefully designs the RF waveforms to ensure that a specific number of bits can be embedded into a single image. To this end, the duration of the emitted signal should match to the image acquisition duration. To compute the required acquisition time for a single image, we must first determine the total number of image rows W. During actual image acquisition, the CMOS camera captures images at its maximum resolution and subsequently resizes the captured image according to the user-specified target resolution. Therefore, we only need to verify the maximum resolution of the camera model used to determine the parameter W. Then the image acquisition time t_S can be calculated based on the total number of image rows W and the sampling frequency f_{sample} of the CMOS camera:

$$t_S = \frac{W}{f_{\text{sample}}} \tag{5}$$

(a) Original Image (b) Orange Channel (c) Blue Channel

Fig. 8: Example of how watermark stripe enhancement using color space transformation and model-based knowledge of stripe colors could amplify the watermarks.



(a) Real Image (b) Gaussian Noise Background (c) Pure Black Background

Fig. 9: Comparison between real (a) and simulated images. The simulated watermarks with Gaussian noise background (b) is blended with the clean image datasets to synthesize training input of U-Net, and the simulated stripes with pure black background (c) serves as the training output.

Once t_S is calculated, the duration of the signal corresponding to each bit t_B can be obtained accordingly.

The required signal frequency f_{signal} that generates exactly num_bits stripes over the duration t_S can be computed according to Equation 1, 2 and 3:

$$f_{signal} = \frac{num\_bits \times f_{sample}}{W} + N \times \frac{f_{sample}}{4} \qquad (6)$$

Finally, we generate a sinusoidal waveform with frequency f_{signal} and duration t_S, and divide this waveform evenly into num_bits segments, each corresponding to one bit with a duration of t_B. If the bit value is "1", the corresponding waveform segment remains unchanged, resulting in a stripe in the image. If the bit value is "0", the waveform segment is set to zero, so that the region where a stripe would otherwise appear remains the same as the original image without RF injections. Subsequently, we utilize an arbitrary waveform generator combined with an antenna to transmit the constructed waveform in a continuous loop.

C. Stripe Extraction

The next critical challenge is accurate extraction of the embedded imperceptible stripes from images with complex backgrounds. Note that in real-world scenarios, the defender cannot simply take the difference between two images as demonstrated in Fig. 5 because the ground-truth image without RF injection is unknown to the defender. To address this, we propose a stripe extraction approach involving a stripe enhancement algorithm and a background removal model to enable accurate retrieval of invisible stripes.

1) Stripe Enhancement: Fig. 8 (a) illustrates an image with a complex background embedded with alternating orange and blue stripes, which are barely perceptible to the human eye. To enhance the visibility of the embedded stripes, we transform the image from its original RGB color space into a specialized color space defined explicitly by the known stripe colors, such as orange and blue vectors, and their orthogonal vector. Fig. 8 (b) and (c) visualize the resultant orange and blue channel images with the enhanced stripes. For example, bright regions in the orange channel image indicate orange stripes. We further apply Contrast Limited Adaptive Histogram Equalization (CLAHE) to the transformed orange and blue channel images to enhance contrast between alternating stripes.

2) Background Removal Model: The background removal model proposed aims to accurately extract clean embedded stripe patterns from complex backgrounds. While stripe enhancement helps make previously invisible patterns visible, the presence of complex backgrounds still poses a significant challenge to stripe recognition. Specifically, relying solely on inter-row pixel intensity variations for detection becomes infeasible, as background clutter introduces substantial noise that can obscure the embedded stripe signals. To address this issue, we adopt the image segmentation network U-Net [28] to separate stripes from the complex background. U-Net features end-to-end spatial alignment between input and output, allowing the model to learn pixel-level mappings between images with complex background interference and their corresponding clean stripe representations. Additionally, the skip connection mechanism allows shallow features such as spatial position information to be directly passed to the decoder stage, preserving the precise spatial localization of stripes. This precise position retention is essential for identifying regularly distributed stripe patterns and preventing the loss of elongated stripe structures during downsampling, thereby enhancing stripe reconstruction performance under complex background interference.

Training the background removal model requires a paired input-output dataset, where each input is a background image with injected stripes, and the output is the corresponding clean stripe pattern. Specifically, the actual input to the model is composed by concatenating the orange and blue channels extracted from the stripe-injected background image after applying a stripe enhancement process, which highlights the embedded stripe signals for improved visibility. However, constructing such a dataset consisting using physically injected stripes is highly challenging and impractical. Since the locations of the physically injected stripes are uncontrollable during image acquisition, it is impossible to guarantee that their spatial distribution under complex background conditions matches that captured in ideal output of background-free settings (e.g., when the CMOS camera is physically occluded). *This poses a significant challenge for applying any data-driven background removal techniques.*

To resolve this problem for training the U-Net model, we

simulate stripe images over Gaussian noise backgrounds based on the modeled mechanism of RF injection into CMOS sensors (Section III-C). As shown in Fig. 9 (b), the simulated image closely resembles the real captured image in Fig.9 (a). To further obtain stripe-injected background images as inputs to the model, we blend the simulated stripe images generated on Gaussian noise backgrounds with real background images using a fixed opacity. The specific blending formula is as follows:

$$I_{stripe} = I_{bg} + opacity \times I_{sim} \tag{7}$$

where I_{stripe} is the synthesized background image with injected stripes, I_{bg} is the clean background image without stripes, I_{sim} is the simulated stripe image generated on Gaussian noise, and we set $opacity = 0.04$.

However, additional challenges arise due to the camera random imaging noise and the image demosaicing process, which could create extra visual artifacts in image regions corresponding to a bit value of 0. Specifically, these regions can exhibit patterns that appear slightly lighter than the surrounding orange areas, thereby introducing misleading visual cues that may interfere with the accurate recognition of stripe patterns. To mitigate the interference caused by this phenomenon, we generated an additional set of simulated stripe images that have a pure black background, where all initial pixel values are set to zero, in parallel with the stripe images simulated on Gaussian noise backgrounds. During the simulation of stripe injection on the pure black background, we deliberately increased the strengths of the injected signal to enhance the visibility of the stripe patterns. As shown in Fig. 9 (c), the color of the stripes changed from the original orange to a more vivid red. This visual shift is consistent with the effect observed in real captured images when the injection signal strength is increased using a RF signal amplifier, thereby validating the simulation's fidelity to real-world conditions.

The stripe-injected background images synthesized using Equation 7 are used as inputs to the model. At the same time, the stripe images simulated on pure black backgrounds serve as the output for the U-Net model. This design enables the model to produce a pure black background when the bit value is 0 and a clearly colored stripe when the bit value is 1, thereby significantly reducing the complexity of subsequent stripe to bit transformation task. It is worth noting that the training set only includes synthesized images from the simulation so that we can generate training images in a scalable manner. In contrast, we directly evaluate our system on real-world RF-injected images during testing. This will enable us to further verify the correctness and generalizability of the stripe injection modeling.

D. Stripe2Bit

After constructing the training dataset and training the U-Net model, we use real-world captured images with injected stripe watermarks as the test input. The model then generates clean stripe images, thereby effectively eliminating the interference of complex backgrounds. Based on clean stripe images from the U-Net, we develop the watermark decoding module consisting of two main components: Bit Region Segmentation and Bit Value Determination.

In bit region segmentation component, since the position of the injected stripes within the image is not controllable during capture, it is not feasible to simply divide the image from the first row. Therefore, we first locate the region where red and blue stripes are most prominent and define it as the starting region. The image is then evenly divided along the row axis into a number of bit regions equal to the total number of bits.

In the bit value determination component, we evaluate each row by computing the sum of its RGB values within each segment. If this sum exceeds a predefined threshold, the row is categorized as red or blue based on its color composition; otherwise, it is considered black. We then count the number of rows within the segment that are classified as either red or blue. If this count exceeds one-fourth of the total number of rows in the segment, the corresponding bit is assigned a value of 1; otherwise, it is set to 0. After identifying the bit value in each region, we obtain a complete bit sequence. However, the region with the most prominent red and blue stripes may not correspond to the actual starting position. Therefore, after feeding the inferred bit sequence into the watermark decoding module, we perform a circular shift based on a predefined preamble to ensure that the bit sequence begins in the correct order, thereby enabling accurate decoding of the embedded watermark information.

We evaluated the bit region segmentation and bit value determination algorithms on 9,000 synthetically generated stripe-only images. The results show that both algorithms can reliably identify stripe regions and accurately determine the corresponding bit values, indicating that the proposed Stripe2Bit module is well-suited for our task.

V. Evaluation

Our evaluation characterizes the feasibility and factors of the RF-based watermark injection technique in proof-of-concept lab settings. The main objective of the evaluation is to identify the factors posing challenges for `RF-Eye-D` watermarking and reveal possible resolutions.

A. Experimental Setup

The experiments use a setup similar to Fig. 3. By default, the system is evaluated on SONY IMX 378 CMOS camera on a Google Pixel smartphone. Other camera sensors will be further evaluated in Section V-C.

1) Datasets: For the simulated training set, we randomly select 10,000 RGB images from the Home Office and Living Room categories of the NYU Depth Dataset V2 [29] as background images. The simulated images are generated in MATLAB following the methodology in Section IV-C.

For the test set, we use real images captured by the camera sensors with injected stripes and evaluate them on the U-Net trained solely on synthetic data. We collected four different categories of test images that represent typical application scenarios of using such watermarking techniques

to defend against malicious photography, including indoor scenes, printed documents, computer screens, and white wall backgrounds. For each test case, the camera sensor captures 20 images of different scenes for each category of scenarios. Each image is embedded with 20 bits of information. To further assess the robustness of the model, we additionally collect 20 images per condition under varying illumination levels, different imperceptibility levels, and different camera devices. In addition, we also collected images embedded with varying numbers of bits for testing to investigate the upper bound of the encoding capacity of the RF watermarking system. In total, 300 watermarked images were physically collected for evaluation for these test cases.

2) Metrics: Since image watermarking requires not only correct transmission of watermark bits but also imperceptibility with various background scenes, we employ a diverse set of three evaluation metrics: peak signal-to-noise ratio (PSNR) for assessing the imperceptibility of the watermark, entropy for assessing the complexity of image background, and bit error rate (BER) for assessing the accuracy of the encoding and decoding process.

Peak Signal-to-Noise Ratio (PSNR). PSNR is a commonly used metric for measuring the similarity between images, which allows us to evaluate how imperceptible the injected watermarks are. The PSNR value is greater than zero, and a higher PSNR denotes more similar images before and after watermark injection and thus better imperceptibility. PSNR is calculated as:

$$PSNR = 10 \cdot \log_{10} \left(\frac{MAX_I^2}{MSE} \right) \tag{8}$$

where MSE refers to Mean Squared Error:

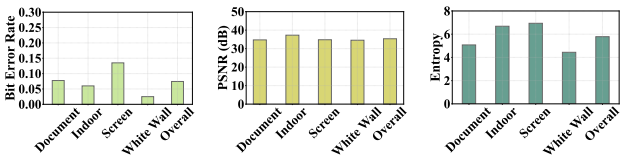
$$MSE = \frac{1}{mn} \sum_{i=1}^{m} \sum_{j=1}^{n} \left(I_{stripe}(i,j) - I_{bg}(i,j) \right)^2 \tag{9}$$

MAX_I denotes the maximum possible pixel value of the image; H and W represent the height and width of the image; I_{stripe} denotes the image with injected stripe signals; I_{bg} represents the background image captured from the same viewpoint without stripe injection. It is worth noting that I_{bg} is introduced solely for the purpose of evaluating watermark imperceptibility during the experimental phase, and is not required by `RF-Eye-D` in real-world applications. Examples of images with different levels of PSNR are provided in Appendix A.

Entropy. Image entropy reflects the degree of disorder in the pixel value distributions [30], [31]. A higher entropy value indicates greater complexity in the image scenes. The entropy of an image is defined as:

$$Entropy = -\sum_{i=0}^{255} p_i \log_2 p_i \tag{10}$$

where p_i denotes the probability of pixels with gray-scale value i, and the range of i depends on the bit depth of the grayscale image converted from the original image. For an 8-bit grayscale image, $i \in [0, 255]$ and the maximum possible entropy is 8. Examples of images with different ranges of entropy are provided in Appendix B.



(a) BER (b) PSNR (c) Entropy

Fig. 10: Overall performance of `RF-Eye-D` system.

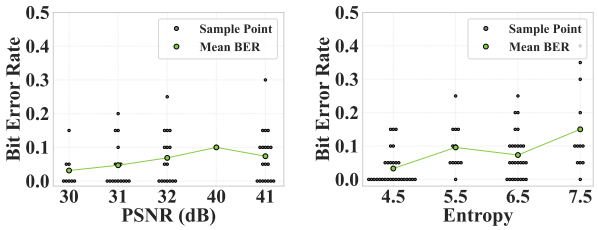
Bit Error Rate. BER is an important metric for evaluating the reliability of watermark information transmission. BER ranges from 0 to 1, where a lower BER value indicates higher decoding accuracy of the watermark. It is defined as the ratio between the number of erroneous bits and the total number of bits in the sequence:

$$BER = \frac{error\_bits}{num\_bits} \tag{11}$$

B. Overall Watermarking Performance

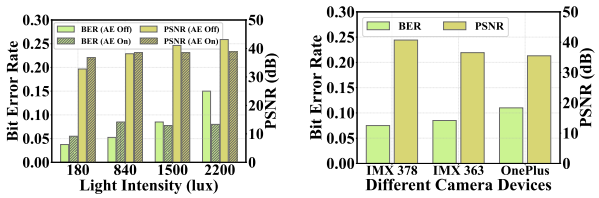
Fig. 10 shows the BER, PSNR, and Entropy across the four test scenarios. We observe that the BER of the RF watermarking system remains below 0.1 in the document, indoor, and white wall scenarios, while it exceeds 0.1 in the screen scenario. Interestingly, although the PSNR in the computer screen scenario is lower than that in the indoor scenario, which suggests that the stripe patterns are more visually apparent, the decoding performance was actually worse. This phenomenon may be attributed to two factors: First, the average visual complexity of images in the screen scenario is relatively higher. Second, when capturing screen content using a camera, moiré patterns may appear in the images [32], which can interfere with the stripe extraction algorithm and lead to a higher BER value. We further investigate the impact of moiré patterns on the system in the Appendix C. In the white wall scenario, although the PSNR is comparable to other scenarios, the significantly lower background complexity leads to higher decoding accuracy. In contrast, the BER is lower in the indoor scenario despite its higher background complexity and PSNR compared to the dataset average. This suggests that even under conditions of more imperceptible stripes and greater background complexity, `RF-Eye-D` has better decoding performance in the indoor scenario. One possible explanation is that the stripe extraction model was primarily trained on synthetic data generated from indoor scenes, which may enhance its extraction ability in such environments, thereby reducing the BER value.

The results above demonstrate the potential of using RF injection to watermark images, but also reveal several factors that may affect its performance.



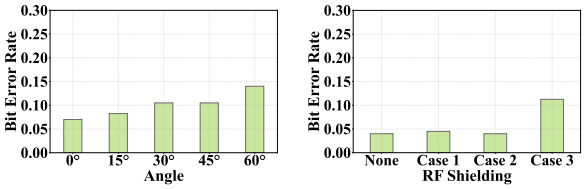
(a) BER under different PSNR (b) BER under different Entropy

Fig. 11: Impact of different watermark imperceptibility and background complexity.



(a) BER and PSNR under Different Lighting Intensities (Auto Exposure Camera Devices disabled vs. enabled) (b) BER and PSNR under Different

Fig. 12: Impact of lighting and camera device variations.



(a) BER under Different Angles (b) BER under Different RF Shielding Cases

Fig. 13: Impact of angles and RF shielding.

C. Factors

We further characterize the robustness of RF-Eye-D to the variations of key factors to quantify the impact of possible interference of the physical environment, device hardware, etc.

1) Imperceptibility Level: We first evaluate how RF-Eye-D performs under varying levels of watermark imperceptibility by collecting a total of 60 images with varying output strengths of the RF signals. All images were captured against the same background to control variables and eliminate the impact of background complexity.

Fig. 11 (a) illustrates the distribution of PSNR and the corresponding BER values across the collected 60 watermarked images. It is observed that under the consistent background scene, the average BER tends to increase as PSNR increases, verifying that better imperceptibility leads to lower watermark decoding accuracy. A small rise-and-fall glitch appears around a PSNR of 40 and is most likely caused by a single sample point, and can thus be considered an outlier rather than a general trend. Notably, even when the PSNR reaches 41 dB, a level at which the stripe becomes highly imperceptible (as Fig. 14 in Appendix A shows that stripes are already barely visible at 35 dB), the average BER remains below 0.1, and most image samples also maintain a BER at or below 0.1. This relatively robust performance is attributed to the effectiveness of the stripe enhancement algorithm, which amplifies the colored signals of subtle stripes in the transformed color space, thus ensuring accurate extraction of the embedded bit information even under highly imperceptible conditions.

2) Image Background Complexity: As observed in our main experiments, the BER of watermark extraction is not only related to imperceptibility but is also affected by the complexity of the image background. To quantify background complexity, we calculate the entropy of each image collected in the main experiments (Section V-B) and group them into discrete intervals with a step size of 1 to analyze the corresponding BER distribution. Fig. 11 (b) shows that BER generally increases with increasing background complexity, though slight fluctuations are observed. The results indicate that our watermarking system maintains an average BER below 0.1 for images with simple or moderately complex backgrounds. However, performance declines on images with extremely high background complexity, particularly those with entropy values in the range of 7 to 8. This suggests that the performance of our RF watermarking system still needs further improvement when dealing with images that feature highly complex textures and rich colors, such as natural landscapes in outdoor scenes. Section VI will further discuss the possible ways of future improvements.

3) Ambient Lighting: To evaluate the robustness of the RF-Eye-D system under varying lighting conditions, we captured images of the same background scene while adjusting the illumination intensity. Since modern cameras typically apply auto-exposure to compensate for lighting changes, we conducted experiments under two separate configurations: with auto-exposure disabled and with it enabled. When auto-exposure was turned off, the image brightness directly reflected the actual ambient illumination. When enabled, the internal image processing algorithm of camera will adjust the overall brightness to compensate for lighting variations. For each exposure setting, we collected 80 images under four illumination levels: 180 lux, 840 lux, 1500 lux, and 2200 lux.

As shown in Fig. 12 (a), when auto-exposure was disabled, PSNR increases as lighting intensity becomes stronger. This is because the embedded stripes become more visually prominent in darker environments, reducing their imperceptibility and thus causing lower PSNR. Meanwhile, we observe that BER increases as PSNR increases, indicating that more imperceptible stripes result in lower watermark decoding accuracy. Overall, the RF watermarking system maintains an average BER lower than 0.1 under lighting conditions up to 1500 lux. Although system performance may degrade under extremely bright lighting conditions (2200 lux), images captured in such environments are often over-exposed, which obscures visual

TABLE I: Parameters of Tested Camera Devices

Phone Model	CMOS Sensor	Resolution	Typical RF Freq.
Google Pixel	IMX 378	4048 × 3036	116.9692 kHz
Google Pixel 3	IMX 363	4032 × 3024	117.9676 kHz
OnePlus DE2117	Unknown	4160 × 3120	143.4675 kHz

content and consequently reduces the feasibility of malicious photography attacks.

With auto-exposure enabled, the system exhibited slightly lower performance in low-light conditions, as brightness compensation made the embedded stripes more imperceptible, leading to higher PSNR and reduced decoding accuracy. In high-light conditions, auto-exposure reduced image brightness, making the stripes less imperceptible, which resulted in lower PSNR and improved decoding accuracy. Overall, the system exhibited more stable performance under different lighting conditions when auto-exposure was enabled.

4) Angle: To investigate the impact of camera angle on the performance of our RF watermarking system, we adjusted the angle between the camera and the magnetic probe from 0° to 90° in 15° increments and observed the corresponding changes in model performance. Fig. 13 presents the results for the 0°–60° range, illustrating a gradual increase in BER as the angle increases. Further analysis of the experimental data reveals that at an angle of 75°, the watermark could not be detected in nearly half of the samples, and at 90°, it could not be recognized in any of the samples. This is because, increasing the angle reduces the RF signal component perpendicular to the CMOS sensor surface, which lowers the effective injected energy and consequently degrades watermark embedding and extraction performance.

These findings suggest that under conditions where a near field magnetic probe is used without a high-power amplifier, the RF-Eye-D system achieves a maximum effective angular coverage of approximately -60° to +60°, totaling about 120°. To further expand the coverage area of our RF watermarking system, future work can incorporate a high-power amplifier to enhance signal strength, or adopt multi-directional antennas to enable stable signal injection across wider angular ranges.

5) Camera Device: While the experiments above only use a Google Pixel smartphone equipped with a SONY IMX378 camera sensor, `RF-Eye-D` is applicable to various types of CMOS camera sensors. Experiments on two additional smartphones demonstrate this, including a Google Pixel 3 and a OnePlus Nord N200 phone. Table I shows the information on the three tested devices, where "Resolution" denotes the maximum image resolution of the camera, and "Typical RF Freq." refers to the RF frequencies used to inject 20-bit watermarks.

As shown in Fig. 12 (b), the performance of `RF-Eye-D` is not strictly correlated with PSNR values across different devices. While the system achieves lower BER on devices equipped with SONY IMX sensors compared to the OnePlus DE2117, its average BER on all three devices remains below 0.1. It is worth noting that SONY IMX-series CMOS sensors are widely used in smartphones and IoT devices [33]. This observed robustness and generalizability of RF-based watermarks can be attributed to the fact that our stripe extraction model is trained entirely on simulated data and does not rely on hardware-specific characteristics, thereby enabling effective generalization across various camera platforms.

Since different camera device models operate at distinct sampling frequencies, it is necessary to select appropriate RF injection frequencies accordingly. To facilitate this, defenders can identify the device model by analyzing the device's appearance captured in surveillance footage. Alternatively, since different devices emit distinctive electromagnetic signal patterns during operation, defenders may infer the model by detecting these electromagnetic emissions [34]. Furthermore, defenders can also transmit a set of RF frequencies known to be compatible with a variety of commonly used camera sensors, thereby expanding the system's applicability to a wider range of devices.

6) Number of Injected Bits: As a feasibility characterization, this paper primarily evaluated the case of embedding 20 bits per image. However, increasing the number of embedded bits per image not only increases the potential bandwidth but also allows for the use of longer error correction codes to mitigate bit errors. To better understand the potential impact, we further explore the feasibility of embedding more bits per image.

Table II presents the RF frequency parameters and the corresponding BER performance for embedding 20, 50, and 100 bits. Note that more frequencies may be used according to Section III-C while we only aimed to demonstrate a feasible setup here. We found that when the number of embedded bits does not exceed 50, the system could maintain a BER below 0.1. However, the average BER increases to 0.15 when the number increases to 100, indicating that approximately 15 bits per image may be erroneous. This performance degradation is caused by the increased watermark density, which results in narrower stripes that are more difficult to extract accurately.

D. Geotagging and Timestamping

With the understanding of the potential channel capacities informed by the experiments above, it is possible to further analyze the possible types of information that could be injected as watermarks. A comprehensive approach geotagging and timestamping approach involves converting the latitude, longitude, and timestamp at the moment of capture into bits. Achieving a meter-level localization accuracy requires latitude/longitude precision of 0.00001 degrees, corresponding to at least 51 effective bits. Additionally, representing a timestamp with second-level resolution over the range from 1970 to 2038 (the Year 2038 problem [35]) requires 32 more bits. These requirements do not include the overhead for preambles or error correction, thus placing stringent demands on the bit embedding capacity per image.

However, only certain locations are truly sensitive in actual use cases. Embedding full geographic coordinates and timestamps in every image may result in considerable bit

TABLE II: Injecting Different Number of Bits

Number of Bits	RF Frequency	BER
20	116.9692 kHz	0.075
50	117.8890 kHz	0.051
100	119.4220 kHz	0.15

redundancy. We envision a more agile, need-based encoding scheme: assigning a unique identifier to each sensitive region and embedding the identifier instead of the full coordinate data. Similarly, temporal information can be simplified by defining a common starting date and assigning a unique number to each day thereafter.

Need-based Unique ID Watermarking. When the number of stripes is set to 20, the average BER in our experiments ranges from 0.05 to 0.1, indicating that approximately 2 bits may be erroneous per image. To ensure reliable decoding, we employ BCH codes for error correction. Under a configuration of 20 total bits and the ability to correct up to 2 errors, Bose–Chaudhuri–Hocquenghem (BCH) [36] coding requires 10 redundant bits, leaving 10 bits for effective information encoding. When the number of stripes is increased to 50, the observed BER is 0.051, corresponding to roughly 3 erroneous bits per image. Correcting up to 3 bit errors requires 18 redundant bits, leaving 32 bits for effective information encoding. This setup can theoretically support up to 2^{32} unique identifiers. Even if 12 of these bits are reserved to encode time information covering a span of about 10 years, 2^{20} distinct sensitive locations can still be encoded. We believe this capacity is sufficient to support watermarking applications in sensitive areas.

VI. Discussion

This section discusses the observed limitations and possible venues for future works that aim to deploy the proposed RF watermarking techniques.

Alternative Defensive Strategy. While our current system focuses on imperceptible watermark embedding, an alternative direction worth exploring is proactive visual disruption, where the RF power is deliberately increased to induce strong visible distortion in captured images, thereby making unauthorized photos unusable and helping prevent malicious photography in sensitive environments.

RF Injection Distance. As a proof-of-concept to validate the feasibility of RF-injected watermarks, experiments in this work only tested near-field probes, with a maximum injection distance on the order of 10cm. Increasing the RF injection distances can be implemented by employing dedicated far-field RF antennas and higher-power amplifiers, which have been demonstrated extensively in prior research [17], [21], [37]–[40] We also provide a theoretical analysis of long-range injection in Appendix E. Base on the analysis, an RF amplifier capable of outputting 55 W and an antenna with a 10 dBi gain are sufficient to perform such RF-based watermarking injection at a distance of $\mathbf{1\,m}$.

TABLE III: BER under Watermark Removal Methods

Type	Algorithm	Document	Indoor	Screen	WhiteWall	Overall
Baseline	No Removal	0.0775	0.0600	0.1350	0.0250	0.0744
General	Downsampling then Interpolating	0.0775	0.0550	0.1550	0.0450	0.0831
	Format Conversion	0.0725	0.0575	0.1425	0.0225	0.0737
	Compression	0.0750	0.0600	0.1225	0.0300	0.0719
	Image Cut	0.3575	0.3500	0.3937	0.3975	0.3747
	Downscaling	0.0775	0.0500	0.1725	0.0325	0.0831
	Duplication	0.0775	0.0600	0.1350	0.0250	0.0744
	Gaussian Filtering	0.0675	0.0600	0.1500	0.0375	0.0787
	Upscaling	0.0750	0.0600	0.1425	0.0475	0.0812
Targeted	Color Filtering	0.0750	0.0600	0.1600	0.0250	0.0800

Watermark Removal Resistance. To systematically evaluate the vulnerability of our system to such attacks, we investigated the impact of general watermark removal algorithms, targeted watermark removal algorithms, and perceptual watermark removal algorithms.

General Watermark Removal: We adopted eight widely used methods from [41]: image duplication, lossless compression, image upscaling, image downscaling, format conversion (JPG to PNG), image cropping, downsampling with interpolation, and image filtering. We applied these methods to our RF watermarking system, and the results are presented in Table III. According to Table III, all methods except image cropping have minimal impact on our RF watermarking system. Image cropping leads to significant performance degradation, primarily because the watermark in our design is distributed across the entire image. When the image is cropped, portions of the embedded watermark may be physically removed, resulting in decoding failure. However, it is important to note that cropping also destroys meaningful content in the image itself, reducing its overall integrity and usability. Future work can further optimize the watermark design, such as by reducing its spatial footprint or embedding it in central regions that are less likely to be cropped, thereby enhancing robustness and retention.

Targeted Watermark Removal: Since the RF-injected stripes are nearly imperceptible when imaging normal backgrounds but become visible when the camera is covered and the captured frame is entirely dark, we assume that the adversary can obtain a stripe-only image by masking the camera and extract the RGB values of the stripe regions. The attacker then attempts to remove the watermark by filtering out pixels whose RGB values fall within a ±30 range of the extracted values in each channel from an image captured under normal background conditions. The resulting decoding performance is reported in the last row of Table III and shows only a slight increase in BER, indicating minimal degradation. This limited impact can be attributed to the fact that, although the adversary may capture the RGB characteristics of the stripes under dark backgrounds, their color distribution changes when overlaid with real-world image content, thereby undermining the effectiveness of color-based filtering strategies.

Perceptual Watermark Removal: Although the system is designed to make the watermark highly imperceptible to the human eye, it may still become visually detectable under certain extreme conditions, such as when the attacker is very close to the RF transmitter and the transmission power is high. Under such circumstances, the watermark could potentially

be identified and removed using image editing tools such as Photoshop. To address this potential threat, future work can incorporate a human motion tracking mechanism to improve the adaptability and security of the system. For example, the system can utilize surveillance video to monitor the position of the attacker in real time and dynamically adjust the RF signal transmission power based on the distance between the attacker and the transmitter. This mechanism helps ensure successful watermark injection while further reducing its perceptibility, thereby effectively preventing visual detection and removal of the watermark.

RF Device Deployment. To effectively prevent the dissemination of privacy-sensitive images, the RF-Eye-D system currently employs a design based on continuous RF signal emission. While this approach has demonstrated effective defensive capabilities, it raises concerns regarding power consumption and potential electromagnetic interference with surrounding electronic equipment. To mitigate such drawbacks, future work can incorporate human presence detection to enable RF transmission only upon detecting human activity, thereby preserving system effectiveness while minimizing power and environmental impact.

In more complex electromagnetically dense environments, external RF interference could potentially pose challenges to system robustness. However, we note that the operational frequency range of RF-Eye-D is approximately below 40 MHz, whereas prevalent RF sources such as Wi-Fi and 5G signals operate in the GHz band, and FM radio typically exceeds 80 MHz [14]. Therefore, these signals are unlikely to interfere with our watermarking system. Even if interference does occur within the operating band of RF-Eye-D, it would still manifest as stripe artifacts in the captured images, which can be treated as suspicious and flagged for further inspection.

RF Shielding. To evaluate the possibility of bypassing the watermark through RF shielding, we selected three RF shielding bags designed for smartphones, labeled as Case 1, Case 2, and Case 3. The experimental setups for these three cases are illustrated in the Fig. 17 in Appendix D. Case 1 and Case 2 are opaque shielding bags from different brands. To capture images using these two bags, the camera must be exposed to ensure proper imaging. In contrast, Case 3 was intentionally chosen for its transparent material, allowing image capture even when the camera is fully enclosed.

As illustrated in Fig. 13 (b), Case 1 and Case 2 exhibited similar performance to the baseline scenario without shielding. This is because the camera must remain partially exposed for image acquisition, allowing RF watermark signals to be injected as usual. In contrast, Case 3 caused a moderate degradation in system performance, but the BER still remained around 0.1. This suggests that although Case 3 introduces some level of interference, it cannot completely block our RF watermarking system. Moreover, prior work [17] has demonstrated that the signal attenuation induced by RF shielding can be compensated by increasing the injection power. Therefore, future work can boost the transmission strength to ensure reliable watermark embedding and decoding when facing shielding-based countermeasures.

VII. Related Work

This section provides further background information on the two categories of most related previous works.

Camera-based Communications. Prior studies have shown that external physical signals can interfere with the image formation process by exploiting the rolling shutter mechanism of CMOS cameras, enabling high-bandwidth camera-based communication. These methods typically rely on either optical signals [27], [42], [43] or magnetic signals [8]. In optical communication, modulated LED lights are used to embed information into images. While [42], [43] require clean backgrounds for reliable decoding, [27] supports more complex scenes but depends on producing visibly prominent stripes during capture, which compromises imperceptibility and limits its suitability for watermarking applications. With magnetic signals, Magcode [8] used modified NFC devices to cause black-and-white stripes embedded in the imaging stage. However, Magcode requires the camera to be blocked during operation, as its stripe extraction relies on simple binarization. This leads to significant performance degradation or failure in the presence of background content. Our work builds upon these two lines of work and provides dedicated modeling and system design to demonstrate methodologies for RF-based imperceptible watermarking with complex image scenes.

Physical method-based Camera Watermarking. Prior studies have explored embedding watermarks into camera-captured images via physical-layer signal injection. mID [32] proposed a Moiré-pattern-based method that subtly modifies screen content to prevent photography of sensitive information. However, it is limited to screen-capturing applications and cannot generalize to diverse scenarios. Xinyu et al. [44] introduced an LED-based method to inject visible stripes into CMOS cameras for obstructing unauthorized photography. While effective for disruption, this approach destructively alters image content and is thus unsuitable for covert watermarking.

VIII. Conclusion

This paper presents the `RF-Eye-D` system, which for the first time demonstrates the feasibility of embedding imperceptible watermarks into images captured by non-cooperative CMOS cameras using RF signals. We uncover the underlying mechanism and patterns of RF-induced stripe injection in CMOS sensors, and develop a method to extract these stripes from complex image backgrounds and decode them into watermark information. Furthermore, we evaluate the impact of critical factors on watermarking performance in lab environments and outline key directions for future research.

IX. Acknowledgment

We thank the reviewers and SPQR Lab members for their feedback on early drafts, and Qinhong Jiang and Nina Shamsi for their constructive suggestions. This research was supported in part by the National Science Foundation (NSF) Industry-University Cooperative Research Centers Program, CHEST, under grant IUCRC-1916762.

REFERENCES

[1] S. Enfield. (2022) How many photos will be taken in 2022? Mylio News. Accessed: 2025-04-16. [Online]. Available: https://news.mylio.com/how-many-photos-taken-in-2022/

[2] H. T. Sencar and N. Memon, *Digital image forensics*. Springer, 2013.

[3] A. Piva, "An overview on image forensics," *International Scholarly Research Notices*, vol. 2013, no. 1, p. 496701, 2013.

[4] R. Chandramouli and N. Memon, "Analysis of lsb based image steganography techniques," in *Proceedings 2001 international conference on image processing (Cat. No. 01CH37205)*, vol. 3. IEEE, 2001, pp. 1019–1022.

[5] B. L. Gunjal and R. Manthalkar, "An overview of transform domain robust digital image watermarking algorithms," *Journal of Emerging Trends in Computing and Information Sciences*, vol. 2, no. 1, 2010.

[6] C.-T. Hsu and J.-L. Wu, "Hidden digital watermarks in images," *IEEE Transactions on image processing*, vol. 8, no. 1, pp. 58–68, 1999.

[7] M. Barni, F. Bartolini, and A. Piva, "Improved wavelet-based watermarking through pixel-wise masking," *IEEE transactions on image processing*, vol. 10, no. 5, pp. 783–791, 2001.

[8] D. Dai, Z. An, Q. Pan, and L. Yang, "Magcode: Nfc-enabled barcodes for nfc-disabled smartphones," in *Proceedings of the 29th Annual International Conference on Mobile Computing and Networking*, ser. ACM MobiCom '23. New York, NY, USA: Association for Computing Machinery, 2023. [Online]. Available: https://doi.org/10.1145/3570361.3592528

[9] Q. Jiang, X. Ji, C. Yan, Z. Xie, H. Lou, and W. Xu, "{GlitchHiker}: Uncovering vulnerabilities of image signal transmission with {IEMI}," in *32nd USENIX Security Symposium (USENIX Security 23)*, 2023, pp. 7249–7266.

[10] Y. Long, Q. Jiang, C. Yan, T. Alam, X. Ji, W. Xu, and K. Fu, "Em eye: Characterizing electromagnetic side-channel eavesdropping on embedded cameras," in *Network and Distributed Systems Security (NDSS) Symposium*, 2024.

[11] Y. Ren, Q. Jiang, X. Ji, C. Yan, and W. Xu, "Ghostshot: Manipulating the image of ccd cameras with electromagnetic interference," in *Network and Distributed System Security Symposium (NDSS)*, January 2025.

[12] B. Bayer, "Color imaging array," *United States Patent, no. 3971065*, 1976.

[13] R. A. Roberts and C. T. Mullis, *Digital signal processing*. Addison-Wesley Longman Publishing Co., Inc., 1987.

[14] Wikipedia contributors, "Raw image format — wikipedia, the free encyclopedia," April 2025, accessed: 2025-04-24. [Online]. Available: https://en.wikipedia.org/wiki/Raw_image_format

[15] P. Mathur and S. Raman, "Electromagnetic interference (emi): measurement and reduction techniques," *Journal of Electronic Materials*, vol. 49, pp. 2975–2998, 2020.

[16] R. E. Taylor, "Radio frequency interference handbook," Tech. Rep., 1971.

[17] D. F. Kune, J. Backes, S. S. Clark, D. Kramer, M. Reynolds, K. Fu, Y. Kim, and W. Xu, "Ghost talk: Mitigating emi signal injection attacks against analog sensors," in *2013 IEEE Symposium on Security and Privacy*, 2013, pp. 145–159.

[18] Y. Long, S. Rampazzi, T. Sugawara, and K. Fu, "Protecting covid-19 vaccine transportation and storage from analog cybersecurity threats," *Biomedical Instrumentation & Technology*, vol. 55, no. 3, pp. 112–117, 2021.

[19] Y. Tu, S. Rampazzi, B. Hao, A. Rodriguez, K. Fu, and X. Hei, "Trick or heat? manipulating critical temperature-based control systems using rectification attacks," in *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, 2019, pp. 2301–2315.

[20] Z. Jin, Q. Jiang, X. Lu, C. Yan, X. Ji, and W. Xu, "Phantomlidar: Cross-modality signal injection attacks against lidar," in *2025 Network and Distributed System Security (NDSS) Symposium*. Internet Society, 2025.

[21] Q. Jiang, Y. Ren, Y. Long, C. Yan, Y. Sun, X. Ji, K. Fu, and W. Xu, "Ghosttype: The limits of using contactless electromagnetic interference to inject phantom keys into analog circuits of keyboards," in *Network and Distributed Systems Security (NDSS) Symposium*, 2024.

[22] X. Zhang, Y. Tu, Y. Long, L. Shan, M. A. Elsaadani, K. Fu, Z. Lin, and X. Hei, "From virtual touch to tesla command: Unlocking unauthenticated control chains from smart glasses for vehicle takeover," in *2024 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2024, pp. 2366–2384.

[23] S. Köhler, R. Baker, and I. Martinovic, "Signal injection attacks against ccd image sensors," in *Proceedings of the 2022 ACM on Asia Conference on Computer and Communications Security*, ser. ASIA CCS '22. New York, NY, USA: Association for Computing Machinery, 2022, p. 294–308. [Online]. Available: https://doi.org/10.1145/3488932.3497771

[24] S. A. Carlton, "Industrial espionage: reality of the information age," *Research-Technology Management*, vol. 35, no. 6, pp. 18–24, 1992.

[25] B. Wimmer, CPP, *Business Espionage: Risks, Threats, and Countermeasures*. Butterworth-Heinemann, 2015.

[26] B. G. Mujtaba, "Cybercrimes and safety policies to protect data and organizations," *Journal of Crime and Criminal Behavior*, vol. 4, no. 1, pp. 91–112, 2024.

[27] R. Xiao, L. Zhao, F. Qian, L. Yang, and J. Han, "Practical optical camera communication behind unseen and complex backgrounds," in *Proceedings of the 22nd Annual International Conference on Mobile Systems, Applications and Services*, ser. MOBISYS '24. New York, NY, USA: Association for Computing Machinery, 2024, p. 113–126. [Online]. Available: https://doi.org/10.1145/3643832.3661866

[28] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," 2015. [Online]. Available: https://arxiv.org/abs/1505.04597

[29] P. K. Nathan Silberman, Derek Hoiem and R. Fergus, "Indoor segmentation and support inference from rgbd images," in *ECCV*, 2012.

[30] C. E. Shannon, "A mathematical theory of communication," *The Bell system technical journal*, vol. 27, no. 3, pp. 379–423, 1948.

[31] N. R. Pal and S. K. Pal, "Entropy: A new definition and its applications," *IEEE transactions on systems, man, and cybernetics*, vol. 21, no. 5, pp. 1260–1270, 1991.

[32] W. Xu, Y. Cheng, X. Ji, and Y.-C. Chen, "On tracing screen photos – a moiré pattern-based approach," *IEEE Transactions on Dependable and Secure Computing*, vol. 21, no. 4, pp. 2068–2084, 2024.

[33] Sinoseen. (2024, May) Unveiling the sony imx sensor list: A comprehensive guide to sony's imaging technology. [Online]. Available: https://sinoseen.com/sony-imx-sensor-features-applications--future-trends

[34] T. Burton and K. Rasmussen, "Smartphone model fingerprinting using wifi radiation patterns," *Computer Science and Information Technology*, vol. 11, no. 18, 2021.

[35] G. J. Holzmann, "Out of bounds," *IEEE Software*, vol. 32, no. 6, pp. 24–26, 2015.

[36] R. Bose and D. Ray-Chaudhuri, "On a class of error correcting binary group codes," *Information and Control*, vol. 3, no. 1, pp. 68–79, 1960. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0019995860902874

[37] S.-G. Kim, E. Lee, I.-P. Hong, and J.-G. Yook, "Review of intentional electromagnetic interference on uav sensor modules and experimental study," *Sensors*, vol. 22, no. 6, p. 2384, 2022.

[38] B. B. Yilmaz, E. M. Ugurlu, M. Prvulovic, and A. Zajic, "Detecting cellphone camera status at distance by exploiting electromagnetic emanations," in *MILCOM 2019-2019 IEEE Military Communications Conference (MILCOM)*. IEEE, 2019, pp. 1–6.

[39] M. I. Hossen, Y. Tu, and X. Hei, "A first look at the security of eeg-based systems and intelligent algorithms under physical signal injections," in *Proceedings of the 2023 Secure and Trustworthy Deep Learning Systems Workshop*, 2023, pp. 1–8.

[40] J.-H. Jang, M. Cho, J. Kim, D. Kim, and Y. Kim, "Paralyzing drones via emi signal injection on sensory communication channels." in *NDSS*, 2023.

[41] W. Xu, Y. Cheng, X. Ji, and Y.-C. Chen, "On tracing screen photos – a moiré pattern-based approach," *IEEE Trans. Dependable Secur. Comput.*, vol. 21, no. 4, p. 2068–2084, Jul. 2024. [Online]. Available: https://doi.org/10.1109/TDSC.2023.3299983

[42] J. Hao, Y. Yang, and J. Luo, "Ceilingcast: Energy efficient and location-bound broadcast through led-camera communication," in *IEEE INFOCOM 2016 - The 35th Annual IEEE International Conference on Computer Communications*, 2016, pp. 1–9.

[43] Y.-S. Kuo, P. Pannuto, K.-J. Hsiao, and P. Dutta, "Luxapose: indoor positioning with mobile phones and visible light," in *Proceedings of the 20th Annual International Conference on Mobile Computing and Networking*, ser. MobiCom '14. New York, NY, USA: Association for Computing Machinery, 2014, p. 447–458. [Online]. Available: https://doi.org/10.1145/2639108.2639109

[44] S. Zhu, C. Zhang, and X. Zhang, "Automating visual privacy protection using a smart led," in *Proceedings of the 23rd Annual International Conference on Mobile Computing and Networking*, ser. MobiCom '17.

New York, NY, USA: Association for Computing Machinery, 2017, p. 329–342. [Online]. Available: https://doi.org/10.1145/3117811.3117820

APPENDIX

A. Examples of PSNR levels

Fig. 14 presents examples of images watermarked by the RF-induced stripe patterns under typical PSNR levels. As shown by Fig. 14 (c), we found that it becomes challenging for humans to perceive the injected stripes when the PSNR is 35 dB or higher. Watermarked images with lower PSNR levels may cause some notice but could still be accepted by adversaries.

(a) 24 dB (b) 29 dB (c) 35 dB

Fig. 14: Examples of watermarked images where watermark imperceptibility is measured by different levels of PSNR.

B. Examples of Entropy Ranges

(a) 4~5 (b) 5~6 (c) 6~7 (c) 7~8

Fig. 15: Different ranges of entropy.

Fig. 15 presents examples of images that have different ranges of entropy. As the entropy increases, the background complexity of the images significantly increases. Entropy values in the range of 4 to 5 typically correspond to relatively simple backgrounds; values between 5 and 7 are associated with common daily scenes; and values between 7 and 8 indicate particularly complex backgrounds.

C. The impact of moiré patterns on the RF watermarking system

First, to verify whether moiré patterns could cause false positives in the RF watermarking system, we first collected 20 image samples containing only moiré patterns, without any RF signals. Among them, 4 samples were falsely identified as containing RF watermarks, indicating that moiré patterns can indeed cause false positives. Future work can incorporate structural priors to distinguish between irregular moiré textures and the row-wise stripe patterns characteristic of RF watermarks, thereby avoiding misidentification.

To further evaluate the impact of moiré on watermark extraction accuracy, we conducted experiments under three conditions: without moiré interference, with moiré interference, and with moiré interference while increasing the RF output power. As shown in Fig. 16, the presence of moiré patterns indeed reduces the accuracy of watermark decoding when the RF output power remains the same. However, when we moderately increase the RF signal power, the watermark stripes in the image no longer appear overly weak compared to the moiré patterns, which effectively mitigates the interference and improves watermark decoding accuracy.

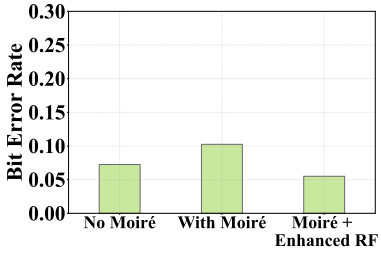


Fig. 16: BER under Moire Pattern Interference

(a) Case 1 (b) Case 2 (c) Case 3

Fig. 17: BER under Moire Pattern Interference

D. RF Shielding Cases

Fig. 17 illustrates the three different RF shielding cases we used during the RF shielding experiments.

E. Theoretical Modeling of Long-range Injection

To simulate long-range injection, we first estimate the minimum magnetic field strength required to induce stripe interference in the image sensor by conducting near-field experiments. We use an SDG6052X signal generator to produce the signal, which is transmitted through a magnetic probe with a radius of 2.5 cm, positioned 1 cm from the image sensor. Stripe interference is observed when the output amplitude reaches 3 V_{pp}.

Based on this experimental observation, we further apply the Biot–Savart law and adopt the expression for the magnetic flux density generated by a circular current loop along its axis to estimate the minimum magnetic field strength corresponding to the interference threshold:

$$B(z) = \frac{\mu_0 I r^2}{2(r^2 + z^2)^{3/2}} \tag{12}$$

Here, $\mu_0 = 4\pi \times 10^{-7}\,\mathrm{H/m}$ is the vacuum permeability, $r = 0.025\,\mathrm{m}$ is the radius of the magnetic probe, $z = 0.01\,\mathrm{m}$ is the axial distance between the probe and the image sensor, and I

represents the equivalent peak current. The input impedance R of magnetic probe is $50\,\Omega$, and the peak-to-peak voltage V_{pp} of the signal generator output is 3 V, we calculate I as:

$$I = \frac{V_{\mathrm{pp}}}{2R} = \frac{3}{2 \times 50} = 0.03 \text{ A} \tag{13}$$

By substituting all values into Equation 12, we calculate that the minimum magnetic field strength B_{min} required to induce stripe interference in the image sensor is approximately $604\,\mathrm{nT}$.

To simulate far-field injection, we employ the Friis transmission formula to estimate the required transmit power for electromagnetic interference at various distances. Rearranging the Friis formula to solve for transmit power yields:

$$P_t = \frac{S \cdot 4\pi d^2}{G_t} \tag{14}$$

where P_t is the transmitted power in watts, S is the power density at the receiver location in $\mathrm{W/m^2}$, d is the distance between transmitter and receiver in meters, and G_t is the transmit antenna gain.

Since our near-field experiments determine the threshold in terms of magnetic field strength B_{min}, we convert this to power density for far-field analysis. For a electromagnetic wave, the power density can be computed from the magnetic field strength as:

$$S = \frac{(B_{min} \cdot c)^2}{2Z_0} \tag{15}$$

where $B_{\mathrm{min}} \approx 604\,\mathrm{nT}$, $c = 3 \times 10^8\,\mathrm{m/s}$ is the speed of light, and $Z_0 = 377\,\Omega$ is the impedance of free space. We then obtain that S is approximately $43.5\,\mathrm{W/m^2}$. By substituting S into Equation 14, we can calculate the required transmit power for different scenarios.